

Using Dreamweaver MX

Trademarks

Afterburner, AppletAce, Attain, Attain Enterprise Learning System, Attain Essentials, Attain Objects for Dreamweaver, Authorware, Authorware Attain, Authorware Interactive Studio, Authorware Star, Authorware Synergy, Backstage, Backstage Designer, Backstage Desktop Studio, Backstage Enterprise Studio, Backstage Internet Studio, Design in Motion, Director, Director Multimedia Studio, Doc Around the Clock, Dreamweaver, Dreamweaver Attain, Drumbeat, Drumbeat 2000, Extreme 3D, Fireworks, Flash, Fontographer, FreeHand, FreeHand Graphics Studio, Generator, Generator Developer's Studio, Generator Dynamic Graphics Server, Knowledge Objects, Knowledge Stream, Knowledge Track, Lingo, Live Effects, Macromedia, Macromedia M Logo & Design, Macromedia Flash, Macromedia Xres, Macromind, Macromind Action, MAGIC, Mediamaker, Object Authoring, Power Applets, Priority Access, Roundtrip HTML, Scriptlets, SoundEdit, ShockRave, Shockmachine, Shockwave, Shockwave Remote, Shockwave Internet Studio, Showcase, Tools to Power Your Ideas, Universal Media, Virtuoso, Web Design 101, Whirlwind and Xtra are trademarks of Macromedia, Inc. and may be registered in the United States or in other jurisdictions including internationally. Other product names, logos, designs, titles, words or phrases mentioned within this publication may be trademarks, servicemarks, or tradenames of Macromedia, Inc. or other entities and may be registered in certain jurisdictions including internationally.

This guide contains links to third-party Web sites that are not under the control of Macromedia, and Macromedia is not responsible for the content on any linked site. If you access a third-party Web site mentioned in this guide, then you do so at your own risk. Macromedia provides these links only as a convenience, and the inclusion of the link does not imply that Macromedia endorses or accepts any responsibility for the content on those third-party sites.

Apple Disclaimer

APPLE COMPUTER, INC. MAKES NO WARRANTIES, EITHER EXPRESS OR IMPLIED, REGARDING THE ENCLOSED COMPUTER SOFTWARE PACKAGE, ITS MERCHANTABILITY OR ITS FITNESS FOR ANY PARTICULAR PURPOSE. THE EXCLUSION OF IMPLIED WARRANTIES IS NOT PERMITTED BY SOME STATES. THE ABOVE EXCLUSION MAY NOT APPLY TO YOU. THIS WARRANTY PROVIDES YOU WITH SPECIFIC LEGAL RIGHTS. THERE MAY BE OTHER RIGHTS THAT YOU MAY HAVE WHICH VARY FROM STATE TO STATE.

Copyright © 1997 - 2002 Macromedia, Inc. All rights reserved. This manual may not be copied, photocopied, reproduced, translated, or converted to any electronic or machine-readable form in whole or in part without prior written approval of Macromedia, Inc. Third Party Software Notices and/or Additional Terms and Conditions can be found at <http://www.macromedia.com/go/thirdparty/>. Part Number ZDW60M300

Acknowledgments

Project Management: Sheila McGinn

Writing: Chris Bedford, Kim Diezel, Jed Hartman, Charles Nadeau, Jennifer Rowe

Editing: Mary Ferguson, Mary Kraemer, Lisa Stanziano

Production Management: Patrice O'Neill

Multimedia Design and Production: Aaron Begley, Benjamin Salles, Noah Zilberberg

Print and Help Design and Production: Caroline Branch, John Francis

Illustrations: Chris Basmajian

Web Editing and Production: George Brown, Rebecca Godbois, Jeff Harmon, Jon Varese

Special thanks to Luciano Arruda, Jake Cockrell, Kristin Conradi, George Comninos, David Deming, Tonya Estes, Stephanie Goss, David Halbakken, Nick Halbakken, Wanda Huang, Narciso (nj) Jaramillo, Craig Jennings, Ken Karleskint, Sho Kuwamoto, David Leno, Jay London, Bonnie Loo, Sam Mathews, Larry McLister, Susan Morrow, Masayo Noda, Dan Radigan, Mike Sundermeyer, Heidi Bauer Williams, Jorge Taylor, Lawrence Teschmacher, Venu Venugopal, and the entire Dreamweaver engineering and QA teams.

First Edition: June 2002

Macromedia, Inc.

600 Townsend St.

San Francisco, CA 94103

CONTENTS

Part I ***Dreamweaver Basics***

CHAPTER 1

Welcome to Dreamweaver	17
Learning Dreamweaver	18
Website creation workflow	19
Where to start	23
Typographical conventions	26
What's new in Dreamweaver MX	26
HTML and web technologies resources	29

CHAPTER 2

Exploring the Workspace	31
Using the Dreamweaver workspace	31
Using windows and panels in Dreamweaver	37
Using Dreamweaver with other applications	49
Customizing Dreamweaver: Basics	50

CHAPTER 3

Planning and Setting Up Your Site	55
About site planning and design	55
Setting up a Dreamweaver site	59
Using the Site Definition Wizard	60
Setting up a local folder	61
Setting up a remote folder	63
Editing a Dreamweaver site	65
Editing existing websites in Dreamweaver	66

CHAPTER 4

Managing Your Site	69
About the Site panel	70
About the site map	80
Importing and exporting sites	86
Removing a site from your list of sites	87
Using Check In/Check Out	87

Getting and putting files	90
Synchronizing the files on your local and remote sites	93
Cloaking folders and files in your site	94
Using Design Notes.	97
Using reports to improve workflow	101
About the Sitespring panel in Dreamweaver	103
Using the Sitespring panel.	104

CHAPTER 5

Setting Up a Document.	107
Creating Dreamweaver documents	108
Working with the New Document dialog box	108
Opening existing documents	111
Setting document properties	112
Working with colors	114
Selecting elements in the Document window	116
Using visual guides in the design process	117
Viewing and editing head content	119
About automating tasks.	120

Part II

Preparing to Build Dynamic Sites

CHAPTER 6

Setting Up a Web Application.	127
A note for Dreamweaver UltraDev 4 users.	127
What you need to build web applications	128
Setting up a web server	129
Setting up an application server	129
Creating a root folder for the application.	132
Defining a Dreamweaver site.	132
Connecting to a database.	135
Troubleshooting application server errors	136

CHAPTER 7

Database Connections for ColdFusion Developers	139
Connecting to a database.	139
Editing or deleting a database connection	140
Connecting using UltraDev 4 connectivity	141

CHAPTER 8

Database Connections for ASP.NET Developers	145
Connecting to a database.	145
Editing or deleting a database connection	148

CHAPTER 9	
Database Connections for ASP Developers	151
Understanding ASP database connections	151
Creating a DSN connection	152
Creating a DSN-less connection	155
Connecting to a database on an ISP	158
Editing or deleting a database connection	160
Reference.	162

CHAPTER 10	
Database Connections for JSP Developers	165
Understanding JSP connections	165
Connecting to a database.	165
Connecting through an ODBC driver.	168
Editing or deleting a database connection	170

CHAPTER 11	
Database Connections for PHP Developers.	171
Connecting to a database.	171
Editing or deleting a database connection	172

Part III

Working with Page Code

CHAPTER 12	
Setting Up Your Coding Environment.	175
Viewing your code.	175
Setting viewing preferences	176
Setting coding preferences	177
Customizing keyboard shortcuts	180
Opening files in Code view by default	180
Setting Validator preferences	181
Managing tag libraries	181
Importing custom tags into Dreamweaver.	184
Using an external HTML editor with Dreamweaver	187

CHAPTER 13	
Coding in Dreamweaver	189
About the Dreamweaver coding environment	189
Writing and editing code.	191
Searching and replacing tags and attributes	196
Accessing language references	200

CHAPTER 14	
Optimizing and Debugging Your Code.	203
Cleaning up your code	203
Verifying that tags and braces are balanced	204
Validating your tags.	205

Making pages XHTML compliant	205
Using the JavaScript debugger	209
Using the ColdFusion debugger	215

CHAPTER 15

Editing Code in Design View	217
Editing code with the Property inspector	217
Editing code with a tag editor	218
Editing code with the Quick Tag Editor	218
Editing code with the tag selector	221
Editing scripts	221
Working with server-side includes	223

Part IV
Designing the Page Layout

CHAPTER 16

Presenting Content with Tables	227
Inserting a table	228
Adding content to a table cell	228
Importing tabular data	229
Selecting table elements	229
Formatting tables and cells	231
Resizing tables	233
Changing column widths and row heights	234
Adding and removing rows and columns	234
Nesting tables	237
Cutting, copying, and pasting cells	237
Sorting tables	239
Exporting table data	239

CHAPTER 17

Laying Out Pages in Layout View	241
About layout cells and tables	242
Switching into and out of Layout view	242
Drawing layout cells and tables	243
Adding content to a layout cell	246
Moving and resizing layout cells and tables	248
Formatting layout cells and tables	250
Setting column width	250
Setting Layout view preferences	254

CHAPTER 18

Using Frames	255
About frames and framesets	256
Deciding whether to use frames	257
About creating frame-based web pages in Dreamweaver	258
Creating frames and framesets	259

Selecting frames and framesets	261
Saving frame and frameset files	263
Viewing and setting frame properties.	264
Viewing and setting frameset properties	264
Controlling frame content with links.	265
Handling browsers that can't display frames.	266
Using JavaScript behaviors with frames	267

Part V

Adding Content

CHAPTER 19

Inserting and Formatting Text	271
Inserting and formatting HTML text	271
Formatting text	273
Using HTML styles to format text	279
About Cascading Style Sheets	285
Converting CSS styles to HTML tags	293
Checking spelling	294
Searching and replacing text	294

CHAPTER 20

Inserting Images	297
About images	297
Inserting an image.	298
Resizing an image	301
Creating a rollover image.	301
Using an external image editor	302
Applying behaviors to images	303

CHAPTER 21

Dreamweaver Integration with Other Applications	305
About Fireworks and Flash integration	306
Working with Dreamweaver and Fireworks	306
Creating a Web photo album	313
Working with Dreamweaver and Flash	314

CHAPTER 22

Inserting Media	319
Inserting and playing media objects.	320
Launching an external editor for media files	320
Using Design Notes with media objects.	321
About Flash content	322
Inserting a Flash button object	322
Inserting a Flash text object	324
Inserting Flash movies	326
Inserting Shockwave movies	326
Adding video	327

Adding sound to a page	327
Linking to an audio file	328
Embedding a sound file	328
Inserting Netscape Navigator plug-in content	329
Inserting an ActiveX control	330
Inserting a Java applet	330
Using behaviors to control media	331

CHAPTER 23

Dreamweaver and Accessibility	333
Using Dreamweaver accessibility features	334
Authoring for accessibility	337
Testing your website for accessibility	343

Part VI
Working with Behaviors and Animations

CHAPTER 24

Using JavaScript Behaviors	349
Using the Behaviors panel	350
About events	350
Applying a behavior	351
About behaviors and text	352
Attaching a behavior to a timeline	352
Changing a behavior	353
Updating a behavior	353
Creating new actions	353
Downloading and installing third-party behaviors	354
Using the behavior actions that come with Dreamweaver	354

CHAPTER 25

Animating Layers	377
About HTML code for layers	378
Creating layers on your page	378
Nesting layers	380
Manipulating layers	381
Adding content to layers	383
Viewing and setting layer properties	384
Using tables and layers for layout	386
Animating your layers	388
Animating layers using behavior actions	395

Part VII

Working with Multiple Pages

CHAPTER 26

Linking and Navigation	399
About document locations and paths.	399
Creating links	402
Managing links	409
Creating jump menus	412
Creating navigation bars	414
Creating image maps.	416
Attaching behaviors to links	418

CHAPTER 27

Managing Site Assets, Libraries, and Templates	419
Using the Assets panel	420
Managing the Assets panel.	428
Working with Library items	430
Creating, managing, and editing library items	431
About Dreamweaver templates	435
Creating a Dreamweaver template	440
Creating editable regions.	442
Creating repeating regions.	443
Defining editable tag attributes	446
About optional regions	447
Creating a template-based document.	449
Editing content in a template-based page	450
About nested templates	454
Creating a nested template	456
Applying a template to an existing document	456
Editing and updating templates.	457
About XML.	458

CHAPTER 28

Testing a Site	461
Checking for browser compatibility.	462
Using Behaviors to detect browsers and plug-ins	463
Previewing pages in browsers.	463
Checking links in a page or site.	465
Fixing broken links	466
Opening linked documents in Dreamweaver.	467
Setting download time and size	467
Using Reports to test a site	468

Part VIII

Making Pages Dynamic

CHAPTER 29

Optimizing the Workspace for Visual Development	473
Displaying useful panels	473
Viewing your database structure within Dreamweaver	475
Viewing live data in Design view	476
Working in Design view without live data	480
Previewing dynamic pages in a browser	480
Restricting database information displayed in Dreamweaver	481

CHAPTER 30

The Dreamweaver Workflow for Dynamic Page Design	483
Designing the page	483
Creating a source of dynamic content	484
Adding dynamic content to a web page	486
Enhancing the functionality of a dynamic page	486
Testing and debugging the page	488

CHAPTER 31

Storing and Retrieving Data for Your Page	491
Using a database to store content	491
Collecting data submitted by users	492
Accessing data stored in session variables	495

CHAPTER 32

Defining Sources of Dynamic Content	501
Understanding dynamic content sources	502
Defining a recordset	502
Defining URL parameters	507
Defining form parameters	508
Defining session variables	509
Defining application variables	510
Defining server variables	511
Caching content sources	516
Changing or deleting content sources	516
Copying a recordset from one page to another page	517

CHAPTER 33

Adding Dynamic Content to Web Pages	519
About Adding Dynamic Content	520
Making text dynamic	520
Making images dynamic	521
Making HTML attributes dynamic	523
Making ActiveX, Flash, and other object parameters dynamic	525

Changing dynamic content	526
Deleting dynamic content	526
CHAPTER 34	
Displaying Database Records.	527
Applying typographic and page layout elements to dynamic data	528
Applying formats to data.	528
Customizing existing data formats	528
Creating recordset navigation links	529
Showing and hiding regions based on recordset results	532
Displaying multiple behaviors	533
Creating a table with a Repeat Region server behavior.	534
Creating a record counter	535
Creating an ASP.NET DataGrid or DataList web control	538
CHAPTER 35	
Using ColdFusion Components	543
Understanding ColdFusion components	543
Visually creating a component in Dreamweaver	544
Viewing ColdFusion components in Dreamweaver	544
Editing ColdFusion components in Dreamweaver.	545
Building web pages that use ColdFusion components	546
CHAPTER 36	
Using Web Services.	547
Understanding web services.	548
Installing and configuring proxy generators	550
Adding a web service proxy using the WSDL description	552
Adding a web service to a page	554
Editing the UDDI web service site list.	556
CHAPTER 37	
Adding Custom Server Behaviors	557
Installing additional server behaviors	557
Creating server behaviors	558
Writing code blocks.	561
Making code blocks conditional	563
Editing and modifying server behavior code	571
CHAPTER 38	
Creating Interactive Forms	573
About form objects	574
Creating a form.	575
Understanding form objects	577
Inserting checkboxes and radio buttons.	584
Adding lists and menus	586
Adding form buttons.	589
About form design.	590

Using a client-side JavaScript function to process a form	590
Using behaviors with forms	591
Creating dynamic form objects	592

Part IX

Developing Applications Rapidly

CHAPTER 39

Building Master/Detail Pages	599
About master/detail pages	599
Building master/detail pages rapidly	600
Building master/detail pages block by block	602
Modifying master/details pages	605

CHAPTER 40

Building Pages that Search Databases	607
About search/results pages	607
Creating the search page	608
Building the results page	609
Creating a detail page for a results page	613
Working with related pages (ASP and JSP users only)	617

CHAPTER 41

Building Pages that Modify Databases.	619
Building a page to insert records	619
Building a page to update records	622
Building a page to delete a record	626
Using stored procedures to modify databases	629
Using ASP commands to modify a database	634
Using JSP prepared statements to modify a database	636

CHAPTER 42

Building Pages that Restrict Access to Your Site	639
Building a registration page	639
Building a login page.	643
Building a page only authorized users can access	645

Part X

Appendixes

APPENDIX A

Beginner's Guide to Databases	651
About databases.	651
Database design basics.	652
Understanding database connections	657

APPENDIX B	
SQL Primer.....	663
Syntax basics.....	663
Defining the columns of a recordset.....	665
Limiting the records in a recordset.....	665
Sorting the records in a recordset.....	668
Joining tables.....	669
APPENDIX C	
Setting Up a DSN in Windows.....	671
Understanding DSNs.....	671
Creating a DSN.....	671
APPENDIX D	
Quick Reference: Macromedia ASP.NET Tags.....	675
INDEX	683

Part I

Dreamweaver Basics

Learn how to use Dreamweaver documentation and other resources, and set up the Dreamweaver workspace to fit your preferred working style. Then plan and set up a site, and begin to create pages.

This part contains the following chapters:

- Chapter 1, “Welcome to Dreamweaver”
- Chapter 2, “Exploring the Workspace”
- Chapter 3, “Planning and Setting Up Your Site”
- Chapter 4, “Managing Your Site”
- Chapter 5, “Setting Up a Document”

CHAPTER 1

Welcome to Dreamweaver

Macromedia Dreamweaver MX is a professional HTML editor for designing, coding, and developing websites, web pages, and web applications. Whether you enjoy the control of hand-coding HTML or prefer to work in a visual editing environment, Dreamweaver provides you with helpful tools to enhance your web creation experience.

The visual editing features in Dreamweaver let you quickly create pages without writing a line of code. You can view all your site elements or assets and drag them from an easy-to-use panel directly into a document. You can streamline your development workflow by creating and editing images in Macromedia Fireworks, then importing them directly into Dreamweaver, or by adding Macromedia Flash objects you create directly in Dreamweaver.

Dreamweaver also includes many coding-related tools and features, including code editing tools in the Code view (such as code coloring and tag completion); reference material on HTML, CSS, JavaScript, CFML, ASP, and JSP; and a JavaScript Debugger. Macromedia Roundtrip HTML technology imports your hand-coded HTML documents without reformatting the code; you can then choose to reformat code with your preferred formatting style.

And Dreamweaver now incorporates and expands on all of the capabilities from Macromedia UltraDev, helping you to build dynamic database-backed web applications using server languages such as ASP, ASP.NET, ColdFusion Markup Language (CFML), JSP, and PHP.

Dreamweaver is fully customizable. You can create your own objects and commands, modify keyboard shortcuts, and even write JavaScript code to extend Dreamweaver capabilities with new behaviors, Property inspectors, and site reports.

This chapter contains the following sections:

“Learning Dreamweaver” on page 18

“Website creation workflow” on page 19

“Where to start” on page 23

“Typographical conventions” on page 26

“What’s new in Dreamweaver MX” on page 26

“HTML and web technologies resources” on page 29

Learning Dreamweaver

Dreamweaver includes a variety of resources to help you learn the program quickly and become proficient in creating your own websites and pages. These resources include a printed *Getting Started* guide, an online help system, and tutorials. In addition, you can find regularly updated tips, TechNotes, examples, and information at the Dreamweaver Support Center on the Macromedia website.

Getting Started guide

The *Getting Started* guide is designed to guide you through the process of creating a simple but functional website using Dreamweaver. It's aimed at users who have created web pages but are unfamiliar with particular aspects of Dreamweaver. It includes chapters on creating and editing simple static pages, editing the code by hand, and creating dynamic web applications. The guide is provided as a printed manual, as part of the help system, and as a PDF file.

Tutorials

The tutorials provide further lessons on particular topics, giving more information in specific areas than the rest of the *Getting Started* guide. By working through the tutorials, you'll learn how to edit a sample website with some of the most useful and powerful design features of Dreamweaver. The tutorials are available in the printed *Getting Started* guide, in Dreamweaver Help (in the Help menu in Dreamweaver, choose Tutorials), and as part of the *Getting Started* PDF file.

If you're already familiar with Dreamweaver's design features but want to know more about building web applications, start with the dynamic-page tutorials, which teach you how to create interactive pages that connect to databases.

Dreamweaver Help

Dreamweaver Help provides comprehensive information about all Dreamweaver features, optimized for online reading. Dreamweaver Help is displayed in the help viewer provided by your operating system: Microsoft HTML Help (Windows) or Apple Help (Macintosh). On both platforms, you can find the information you need in any of four ways:

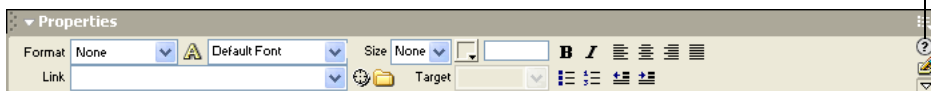
The table of contents enables you to see all of the information organized by subject. Click top-level entries to view subtopics.

The index, like a traditional printed index, allows you to look up specific terms or concepts.

Search allows you to find any character string, anywhere in the text of the help system.

Context-sensitive help provides a way to open a relevant help topic from each dialog box, panel, and inspector. To view context-sensitive help, click a Help button in a dialog box, or choose Help from the Options menu in a panel group's title bar, or click the question-mark icon in an inspector or other kind of window.

Click here to open help



Each help topic provides buttons you can click to move from topic to topic. The left and right arrow buttons move to the previous or next topic in a section (in the order in which the topics are listed in the table of contents).



Extending Dreamweaver

The *Extending Dreamweaver* help system provides information on the Dreamweaver Document Object Model and the APIs (application programming interfaces) that allow JavaScript and C developers to create extensions for Dreamweaver.

Dreamweaver Support Center

To help you get the most out of Dreamweaver, you can consult a web-based support center. The Dreamweaver Support Center website at <http://www.macromedia.com/support/dreamweaver/> is updated regularly with the latest information on Dreamweaver, plus advice from expert users, examples, tips, updates, and information on advanced topics. Check the website often for the latest news on Dreamweaver and how to get the most out of the program.

Design & Developer Center

The Macromedia Design & Developer Center at <http://www.macromedia.com/desdev/> provides tools, tutorials, and more for all Macromedia products.

Dreamweaver online forums

Discuss technical issues and share helpful hints with other Dreamweaver users by visiting the Dreamweaver online forums. You'll find information on accessing the forums on the Macromedia website at http://www.macromedia.com/go/dreamweaver_newsgroup.

Keyboard shortcuts

Charts showing all the keyboard shortcuts in the default Dreamweaver configuration are available on the Macromedia website at http://www.macromedia.com/go/dreamweaver_mx_shortcuts.

Website creation workflow

There are many possible approaches to creating a website. In the workflow we present in this documentation, you start by defining a site's strategy or goals. If you're developing web applications, you have to set up servers and databases as needed. Then you design the look and feel of the site. When the design is complete, you build the site and code the pages, adding content and interactivity; then you link pages together, and test the site for functionality and to see if it meets its defined objectives. You can include dynamic pages in your site as well. At the end of the cycle, you publish the site on a server. Many developers also schedule periodic maintenance to ensure that the site remains current and functional.

To make it easier for you to find the information you need as you develop websites, the *Using Dreamweaver* documentation is divided into broad sections that model this approach to web development—site planning, design, development, testing, and publishing and maintenance.

About site planning

Planning and organizing your site carefully from the start can save you time later on. Organizing your site includes not only determining where the files will go, but also examining site requirements, audience profiles, and site goals. Additionally, you should consider technical requirements such as user access, as well as browser, plug-in, and download restrictions.

Once you've organized your information and determined a structure, you can begin creating your site.

- Determine what strategy to employ and what issues to consider as you plan your site. See Chapter 3, “Planning and Setting Up Your Site,” on page 55.
- Use the Dreamweaver Site panel to set up your site's organizational structure. In the Site panel you can easily add, delete, and rename files and folders to change the organization as needed. See Chapter 3, “Planning and Setting Up Your Site,” on page 55 and Chapter 4, “Managing Your Site,” on page 69.
- You can begin to create simple pages which you'll later turn into more complex designs. Create new blank pages or pages based on predesigned page designs in the New Document dialog box. See Chapter 5, “Setting Up a Document,” on page 107.

If you work on a web-development team, you may also be interested in these topics:

- Set up a system to prevent team members from overwriting files; see “Using Check In/Check Out” on page 87.
- Use Design Notes to communicate with web team members; see “Using Design Notes” on page 97.

About setting up a web application

More and more websites contain dynamic pages that allow visitors to view information stored in databases, and usually allow some visitors to add new information and edit information in the databases. To create such pages, you must first complete several preparatory steps.

- Set up a web server and application server, then create or modify a Dreamweaver site. See Chapter 6, “Setting Up a Web Application,” on page 127.
- Connect to a database. Depending on the server technology your application server uses, see Chapter 7, “Database Connections for ColdFusion Developers,” on page 139, Chapter 8, “Database Connections for ASP.NET Developers,” on page 145, Chapter 9, “Database Connections for ASP Developers,” on page 151, Chapter 10, “Database Connections for JSP Developers,” on page 165, or Chapter 11, “Database Connections for PHP Developers,” on page 171.

About hand-coding

Coding web pages by hand isn't so much a step in the website creation workflow as an approach to creating pages. Dreamweaver provides easy-to-use visual editing tools, but it also provides sophisticated coding tools; you can use either approach, or both, to create and edit your pages.

- You can work in a coding environment without visual tools; coding tools help you create and edit code, format code, make sure that your code adheres to standards, and debug JavaScript code. See Chapter 12, “Setting Up Your Coding Environment,” on page 175 and Chapter 13, “Coding in Dreamweaver,” on page 189, and Chapter 14, “Optimizing and Debugging Your Code,” on page 203.
- You can also use some Dreamweaver coding tools in Design view, the visual design environment. See Chapter 15, “Editing Code in Design View,” on page 217.

About laying out web pages

Many web design projects start with storyboards or flowcharts that are turned into sample pages. Use Dreamweaver to create mock-up sample pages as you work toward a final design. Mock-ups usually show the design layout, site navigation, technical components, themes and color, and graphic images or other media elements, but they don't contain the final contents of the page.

- The table tools and Layout view in Dreamweaver let you quickly design web pages by drawing and then rearranging the page structure. See Chapter 16, “Presenting Content with Tables,” on page 227 and Chapter 17, “Laying Out Pages in Layout View,” on page 241.
- If you want to display multiple documents at once in a web browser, you can lay out documents using frames. See Chapter 18, “Using Frames,” on page 255.

About adding content

Using Dreamweaver, you can easily add a variety of content to web pages. Add assets and design elements, such as text, images, colors, movies, sound, and other forms of media.

- Type directly in a Dreamweaver document, or import text from other documents, then format the text using the Dreamweaver Property inspector, or HTML Styles panel. You can also easily create your own Cascading Style Sheets. See Chapter 19, “Inserting and Formatting Text,” on page 271.
- Use the Assets panel to easily organize the assets in a site; you can then drag most assets directly from the Assets panel into a Dreamweaver document. See “Using the Assets panel” on page 420.
- Insert images, including rollover images, image maps, and Fireworks sliced images. Use alignment tools to position images in a page. See Chapter 20, “Inserting Images,” on page 297 and Chapter 21, “Dreamweaver Integration with Other Applications,” on page 305.
- Insert other types of media in a web page, such as Flash, Shockwave, and QuickTime movies, sound, and applets. See Chapter 22, “Inserting Media,” on page 319.
- Be sure to make your content accessible to people with disabilities. See Chapter 23, “Dreamweaver and Accessibility,” on page 333.

About interactivity and animation

Many web pages are static, containing only text and images. Dreamweaver allows you to go beyond static pages, using interactivity and animation to capture visitors' interest. You can give visitors feedback as they move and click, demonstrate concepts, animate page elements—in short, you can let visitors see and do more within the page.

There are several ways to add interactivity and animation to your pages using Dreamweaver:

- Use behaviors to perform tasks in response to specific events, such as highlighting a button when the visitor passes the pointer over it, validating a form when the visitor clicks the Submit button, or opening a second browser window when the main page is finished loading. See Chapter 24, “Using JavaScript Behaviors,” on page 349.
- Use timelines to create animations that do not require plug-ins, ActiveX controls, or Java. Timelines use dynamic HTML to change the position of a layer or the source of an image over time or to call behavior actions automatically after the page has loaded. See Chapter 25, “Animating Layers,” on page 377.

About connecting the pages on your site

Individual pages don't make a site; you need to link your pages together, both by creating HTML links and by reusing content to give pages a common appearance.

- With Dreamweaver you can create standard HTML links, including anchor links and e-mail links, or easily set up graphical navigation systems, such as jump menus and navigation bars. See Chapter 26, “Linking and Navigation,” on page 399.
- Dreamweaver templates and library files let you easily apply reusable content and page designs to your site. You can create new pages based on a Dreamweaver template, then update the layout of those pages automatically when the template changes. See Chapter 27, “Managing Site Assets, Libraries, and Templates,” on page 419.

About testing and publishing your site

Your site is complete and ready for the world—but before you publish it on a server, you must test the site. Depending on the size of the project, client specifications, and kinds of browsers that visitors will use, you may need to move your site to a staging server where it can be tested and edited. When corrections have been made, you publish the site where the public can access it. Once the site is published, establish a maintenance cycle to ensure quality, respond to user feedback, and update the site's information.

Use the following Dreamweaver features to test and publish your site:

- To add new tags in a page or fix your code, use the Dreamweaver Reference panel to look up JavaScript, CSS and HTML code. See “Accessing language references” on page 200.
- Use the JavaScript Debugger to help you fix JavaScript errors in your code. The debugger lets you set breakpoints in the code, then allows you to view the code as a page is debugged right in Dreamweaver. See “Using the JavaScript debugger” on page 209.
- Run browser and plug-in checks, test and fix links in your documents, and run site reports to check HTML files for common mistakes. See Chapter 28, “Testing a Site,” on page 461.
- In the Dreamweaver Site panel you'll find many tools to help you manage your site, transfer files to and from a remote server, set up a Check In/Check Out process to prevent files from being overwritten, and synchronize the files on your local and remote sites. See Chapter 3, “Planning and Setting Up Your Site,” on page 55.

About creating dynamic pages

In Dreamweaver, you can define a variety of sources of dynamic content, including recordsets extracted from databases, form parameters, and JavaBeans components. To add the dynamic content to a page, simply drag it onto the page.

You can set your page to display one record or many records at a time, display more than one page of records, add special links to move from one page of records to the next (and back), and create record counters to help users keep track of the records.

- If you're unfamiliar with creating web applications in Dreamweaver, learn how to use Dreamweaver to build dynamic pages. See Chapter 29, “Optimizing the Workspace for Visual Development,” on page 473 and Chapter 30, “The Dreamweaver Workflow for Dynamic Page Design,” on page 483.

- Define and display dynamic content on your pages. See Chapter 31, “Storing and Retrieving Data for Your Page,” on page 491, Chapter 32, “Defining Sources of Dynamic Content,” on page 501, Chapter 33, “Adding Dynamic Content to Web Pages,” on page 519, and Chapter 34, “Displaying Database Records,” on page 527.
- Encapsulate application or business logic using leading-edge technologies such as ColdFusion components and web services. See Chapter 35, “Using ColdFusion Components,” on page 543 and Chapter 36, “Using Web Services,” on page 547.
- If you need more flexibility, you can create your own server behaviors and interactive forms. See Chapter 37, “Adding Custom Server Behaviors,” on page 557 and Chapter 38, “Creating Interactive Forms,” on page 573.

About rapid application development

Dreamweaver offers a number of rapid application development (RAD) tools, including server behaviors and application objects, that help you build sophisticated web applications without having to write any server-side code.

- Quickly create pages that search and modify databases and display the results. See Chapter 39, “Building Master/Detail Pages,” on page 599, Chapter 40, “Building Pages that Search Databases,” on page 607, and Chapter 41, “Building Pages that Modify Databases,” on page 619.
- Provide security by restricting access to your pages. See Chapter 42, “Building Pages that Restrict Access to Your Site,” on page 639.

Where to start

This guide includes information for readers from a variety of backgrounds. To get the most out of the documentation, start by reading the parts that are most appropriate for you.

For web-design novices:

- 1 Begin by reading the first two chapters of the *Getting Started* guide. If any section in those chapters seems too advanced or too complex, skip that section and come back to it later.
- 2 In *Using Dreamweaver*, read Chapter 2, “Exploring the Workspace,” on page 31, Chapter 3, “Planning and Setting Up Your Site,” on page 55, Chapter 4, “Managing Your Site,” on page 69, and Chapter 5, “Setting Up a Document,” on page 107.
- 3 Learn about page layout by reading Chapter 17, “Laying Out Pages in Layout View,” on page 241.
- 4 To learn about formatting text and including images in your pages, read Chapter 19, “Inserting and Formatting Text,” on page 271 and Chapter 20, “Inserting Images,” on page 297.
- 5 That’s all you really need to begin producing high-quality websites, but when you’re ready to learn how to use more advanced tools, you can proceed through the rest of the static-page chapters in order. You may want to wait to read the dynamic-page chapters until you’re more familiar with creating web pages.

For experienced web designers who are new to Dreamweaver:

- 1 Begin by reading the first two chapters of the *Getting Started* guide.
- 2 In *Using Dreamweaver*, read Chapter 2, “Exploring the Workspace,” on page 31 to learn more about the Dreamweaver user interface.
- 3 Although much of the material in Chapter 3, “Planning and Setting Up Your Site,” on page 55 and Chapter 4, “Managing Your Site,” on page 69 is probably familiar to you, skim those chapters to see how these familiar concepts are implemented in Dreamweaver. Pay particular attention to the sections on using Dreamweaver to set up a new site.
- 4 Follow the steps in any tutorials that interest you, to learn more about using Dreamweaver to perform specific tasks. To view the tutorials in the help system, choose Help > Tutorials.
- 5 Chapter 19, “Inserting and Formatting Text,” on page 271 and Chapter 20, “Inserting Images,” on page 297 provide useful information on the details of using Dreamweaver to create basic HTML pages.
- 6 Read the overview at the beginning of each of the other chapters to determine whether its topics are of interest to you.

For experienced hand-coders:

- 1 Begin by reading the first and third chapters of the *Getting Started* guide.
- 2 In *Using Dreamweaver*, read Chapter 2, “Exploring the Workspace,” on page 31 to learn more about the Dreamweaver user interface.
- 3 Although much of the material in Chapter 3, “Planning and Setting Up Your Site,” on page 55 and Chapter 4, “Managing Your Site,” on page 69 is probably familiar to you, skim those chapters to see how these familiar concepts are implemented in Dreamweaver. Pay particular attention to the sections on using Dreamweaver to set up a new site.
- 4 Read more about coding with Dreamweaver in Chapter 12, “Setting Up Your Coding Environment,” on page 175, Chapter 13, “Coding in Dreamweaver,” on page 189, Chapter 14, “Optimizing and Debugging Your Code,” on page 203, and Chapter 15, “Editing Code in Design View,” on page 217.
- 5 Read the overview at the beginning of each of the other chapters to determine whether its topics are of interest to you.

For experienced web designers, familiar with Dreamweaver, who want to learn about creating dynamic pages:

- 1 Begin by reading the first, fourth, and fifth chapters of the *Getting Started* guide.
- 2 In *Using Dreamweaver*, skim Chapter 2, “Exploring the Workspace,” on page 31 to learn about new aspects of the Dreamweaver user interface.
- 3 Set up a web server and application server; see Chapter 6, “Setting Up a Web Application,” on page 127.
- 4 Connect to a database. Depending on the server technology your application server uses, see Chapter 7, “Database Connections for ColdFusion Developers,” on page 139, Chapter 8, “Database Connections for ASP.NET Developers,” on page 145, Chapter 9, “Database Connections for ASP Developers,” on page 151, Chapter 10, “Database Connections for JSP Developers,” on page 165, or Chapter 11, “Database Connections for PHP Developers,” on page 171.

- 5 Read the overview at the beginning of each of the other chapters to determine whether its topics are of interest to you.
- 6 If you're interested in customizing Dreamweaver by hand, read "Customizing Dreamweaver" on the Macromedia Support Center at http://www.macromedia.com/go/customizing_dreamweaver. If you want to write extensions for Dreamweaver, read *Extending Dreamweaver* (Help > Extending Dreamweaver).

For experienced web application developers who have not used Dreamweaver or UltraDev:

- 1 Begin by quickly reading the *Getting Started* guide to familiarize yourself with the basics of using Dreamweaver.
- 2 In *Using Dreamweaver*, read Chapter 2, "Exploring the Workspace," on page 31 to learn more about the Dreamweaver user interface.
- 3 Although much of the material in Chapter 3, "Planning and Setting Up Your Site," on page 55 and Chapter 4, "Managing Your Site," on page 69 is probably familiar to you, skim those chapters to see how these familiar concepts are implemented in Dreamweaver. Pay particular attention to the sections on using Dreamweaver to set up a new site.
- 4 Follow the steps in any tutorials that interest you, to learn more about using Dreamweaver to perform specific tasks. To view the tutorials in the help system, choose Help > Tutorials.
- 5 Set up a web server and application server using Dreamweaver; see Chapter 6, "Setting Up a Web Application," on page 127.
- 6 Connect to a database. Depending on the server technology your application server uses, see Chapter 7, "Database Connections for ColdFusion Developers," on page 139, Chapter 8, "Database Connections for ASP.NET Developers," on page 145, Chapter 9, "Database Connections for ASP Developers," on page 151, Chapter 10, "Database Connections for JSP Developers," on page 165, or Chapter 11, "Database Connections for PHP Developers," on page 171.
- 7 Read the overview at the beginning of each of the other chapters to determine whether its topics are of interest to you.

For experienced web application developers who have used UltraDev 4:

- 1 Begin by reading "What's new in Dreamweaver MX" on page 26.
- 2 In *Using Dreamweaver*, skim Chapter 2, "Exploring the Workspace," on page 31 to learn about new aspects of the Dreamweaver user interface.
- 3 If you're interested in customizing Dreamweaver by hand, read "Customizing Dreamweaver" on the Macromedia Support Center at http://www.macromedia.com/go/customizing_dreamweaver. If you want to write extensions for Dreamweaver, read *Extending Dreamweaver* (Help > Extending Dreamweaver).

Typographical conventions

The following typographical conventions are used in this guide:

- Menu items are shown in this format: menu name > menu item name. Items in submenus are shown in this format: menu name > submenu name > menu item name.
- Code font indicates HTML tag and attribute names as well as literal text used in examples.
- *Italic code font* indicates replaceable items (sometimes called *metasymbols*) in code.
- **Bold roman text** indicates text for you to enter verbatim.

What's new in Dreamweaver MX

Dreamweaver MX contains a wide variety of new features, such as new enhanced templates to help visual designers and many new coding capabilities. Dreamweaver also now includes all of the application-development features of Dreamweaver UltraDev 4, and offers many new ones, including a code-centric workspace derived from Macromedia ColdFusion Studio, better runtime code, and support for the latest web application technologies.

Complete Control over Code and Design

You can build the site you want, the way you want, using either the Dreamweaver MX design tools or the rich coding environment.

A new integrated workspace layout (Microsoft Windows only) provides a more familiar working environment with a multiple-document interface (MDI), including fully dockable panels and tabbed document windows.

Macromedia MX panel management in Dreamweaver MX is consistent with the way panels work in Macromedia Flash MX and Fireworks MX. Group together collapsible, dockable panels and collapse or expand them as needed, for a smooth, highly-configurable workflow.

Predesigned sample web components include professional-quality layouts to give you a head start on your designs as well as JavaScript function libraries that give you access to sophisticated client-side interactivity without reading hundreds of pages of textbooks.

Enhanced Dreamweaver templates allow you to set up sophisticated rules for contributors to input content without compromising site design. Template inheritance allows for more customized layout control, and editable regions allow contributors to be more flexible in their input.

The Site Definition Wizard allows you to set up a site quickly and easily, whether you're building a dynamic site for the first time or setting up an account with an ISP.

Code Hints provide easy access to customized menus showing appropriate tag attributes, method properties, function parameters, and CSS styles while you edit in Code view.

Customizable Insert bar allows quick access to objects and behaviors in Dreamweaver MX, and is completely extensible.

The Document toolbar can now be customized to fit your needs and preferences, using XML and JavaScript.

A file explorer is now integrated into the Site panel, allowing you to browse for assets and files on the desktop and network volumes without having to leave Dreamweaver to use the native file explorer in your operating system.

Answers panel connects you directly with online resources from the Macromedia Support Center. You can retrieve new content at will to ensure that you always have access to the latest tips and tricks.

Enhanced table editing and manipulation generates better code for table edits in Layout view and Standard view, streamlining attribute handling and ensuring that all layouts are as lean as possible without compromising cross-browser compatibility.

Cascading JavaScript pop-up menus can now be created right in Dreamweaver MX, a capability formerly available only in Fireworks.

Syntax coloring is now completely configurable and customizable according to your individual needs.

Standard toolbar allows quick access to commonly used file commands (Open, Save, and New) and clipboard commands.

Printing from Code view now allows you to print out your source code, with appropriate formatting applied.

Robust support for all the latest server technologies

Rapidly build dynamic, database driven web applications both by hand coding and by using visual objects and behaviors.

Server code libraries for ColdFusion, ASP, ASP.NET, JSP, and PHP allow you to visually create dynamic websites using all of the leading server-side technologies. Use one development environment to integrate with a variety of sites and back-end technologies.

The Snippets panel allows you to store arbitrary pieces of code for later reuse. You can easily reuse complicated code, layouts, and functions.

The Tag Library Editor provides an integrated tag database for storing information on native and custom tags. It allows you to edit the properties of existing tags, and to import new ones.

Hundreds of tag editors allow you to quickly edit the relevant attributes of specific tags in HTML, CFML, and ASP.NET while in Code view.

ASP.NET custom tag support allows you to import custom ASP.NET server controls (custom tags) into ASP.NET pages, and to see their structure and attributes displayed in the Tag Chooser, Code Hints, and Tag Inspector.

ASP.NET Web Forms tags are now rendered on both Macintosh and Windows, to allow for sophisticated visualization and editing. Web Forms tag editors allow you to quickly author ASP.NET Web Forms using validation and server-side controls.

ASP.NET DataGrid and DataList objects can be easily created in Dreamweaver MX for complex data display and manipulation.

An integrated Database panel allows you to quickly look at the structure and content of your databases, to browse tables, stored procedures, and views before creating your queries.

The Tag Chooser shows the structure of all tags available to Dreamweaver MX, including custom tags imported through the tag library editor. Use it to quickly place and edit tags in your pages, and to view reference information for individual tags.

A customizable XML-based tag database stores information on the structure of all available tags. This central information storage mechanism allows features like Code Hints, the Tag Chooser, and the Tag Inspector to work with both native and custom imported tags.

ColdFusion MX detection allows the Site Definition Wizard to automatically configure itself to set up a ColdFusion site using a locally installed copy of Macromedia ColdFusion MX Developer Edition.

Live Server Debugging panel for ColdFusion MX streamlines the debugging workflow for ColdFusion MX by displaying debug output directly within Dreamweaver MX. View variable information, query information, and other key page statistics, and jump directly to problem areas, all without leaving the development environment.

The **ColdFusion runtime code** generated by Dreamweaver MX server behaviors and objects has been revamped to be more tag-oriented, making it nearly indistinguishable from code a developer would write by hand.

RDS support allows you to transfer files back and forth, and browse the structure of databases on ColdFusion servers, using RDS connections.

JSP tag libraries help developers by abstracting server logic away from JSP pages. Dreamweaver MX can import JSP tag library information from TLD (tag library descriptor) files, and make them available in Code Hints, the Tag Chooser, and the Tag inspector.

Full JavaBeans introspection shows `getProperty` and `setProperty` methods of JavaBeans in the DataBindings panel, and can display all methods and properties of JavaBeans in the Server Components panel to allow developers to fully integrate Beans with their JSP applications.

Dynamic form objects allow you to create database-driven select lists and other form elements simply by selecting the appropriate form object from the Insert bar.

Server Behavior Builder from UltraDev 4 has been enhanced to allow users to build server behaviors with more complicated user interface elements, including checkboxes, radio groups, and connection menus.

Integrated HomeSite 5 and ColdFusion Studio 5 bundle provides a fully licensed version of HomeSite+, the combination of ColdFusion Studio and HomeSite 5, fully integrated and included with Dreamweaver MX.

Easy access to the latest standards

Build websites compliant with the latest web standards, including XML and web services, and keep your skills up to date with cutting-edge development technologies.

Web services introspection for ASP.NET, ColdFusion, and Java allows you to introspect and interoperate with the latest XML-based web services standards in all leading server technologies. Incorporate complicated functionality into your web applications using the web service introspection feature in Dreamweaver MX.

XML editing and validation allows you to import DTDs and schemas, and ensure that documents are well formed.

Enhanced CSS support, including TopStyle integration, helps you build sites compliant with the latest Cascading Style Sheets (CSS) standards. The Dreamweaver MX CSS panel has been enhanced to show internally and externally defined styles, and allow for design-time style sheets. Many new CSS2 constructs are also supported. The text Property inspector can also be used to apply CSS styles.

XHTML support allows you to ensure that your sites are up to date with the latest standards in HTML design by configuring Dreamweaver MX to generate XHTML-compliant code. Also, quickly convert legacy HTML files to XHTML using the XHTML converter.

Section 508 accessibility site reporting help you determine whether your site meets accessibility guidelines and pinpoint problems quickly and easily.

Integrated accessibility reference content in the Reference panel gives you context and background on best practices in designing for accessibility.

Accessibility option for generating compliant code helps during the editing cycle to ensure that the content you are creating meets accessibility guidelines. Configure Dreamweaver to prompt you for specific accessibility-related attributes when appropriate tags and objects are inserted on the page.

ColdFusion Component support offers you increased modularity and portability of server logic by exposing properties through CFML, as a web service, or through the Flash Application Server Gateway. Dreamweaver MX includes support for introspecting, creating, and editing ColdFusion Components.

Code Validator ensures that your documents are meeting the appropriate standards by providing a validation report against a specific browser profile or a specific version of a standard, including XHTML transitional and HTML 4.0.

Launch and edit in Macromedia Flash MX and Fireworks MX accelerates graphics, animation, and HTML editing between Dreamweaver MX, Fireworks MX, and Flash MX, allowing you to launch and optimize directly from Dreamweaver.

Tag Inspector for viewing XML, HTML, and CFML allows you to see the structure of a document quickly and easily. You can quickly spot and eliminate improper nesting and parent relationships without having to look at the raw source code.

Secure FTP login integrates with MacSSH and PUTTY for Windows to allow for SSH enabled secure transfer of login information from the integrated Dreamweaver FTP client.

Full UTF-8 character-set support in all languages allows you to create and transform pages using UTF-8 encoding.

Full Mac OS X and Windows XP support allows Dreamweaver MX to run at full speed in the latest operating systems, while supporting all of the latest user interface improvements.

Macromedia Sitespring integration helps you use best practices in production and client management in the Dreamweaver workspace.

HTML and web technologies resources

The following are some useful resources available on the web:

The HTML 4.01 specification (<http://www.w3.org/TR/REC-html40/>) is the official specification for HTML from the World Wide Web Consortium.

Index DOT HTML (<http://www.blooberry.com/indexdot/html/>) is a comprehensive listing of HTML tags, attributes, and values, as well as their compatibility with the various browsers.

The XHTML 1.0 specification (<http://www.w3.org/TR/xhtml1/>) is the official specification for Extensible HyperText Markup Language.

The O'Reilly XML.com site (<http://www.xml.com/>) provides information, tutorials, and tips about Extensible Markup Language (XML), as well as other web technologies.

The Cascading Style Sheets, level 1 specification (CSS1) (<http://www.w3.org/TR/REC-CSS1>) and **level 2 specification (CSS2)** (<http://www.w3.org/TR/REC-CSS2/>) are the official specifications for style sheets from the World Wide Web Consortium.

WebReview.com's Style Sheet Reference Guide (<http://www.webreview.com/style/index.shtml>) explains what CSS styles are and which browsers they work in.

Microsoft ASP Overview pages (<http://msdn.microsoft.com/workshop/server/asp/ASPOver.asp>) provide information about Active Server Pages (ASP).

Microsoft ASP.NET page (<http://www.asp.net/>) provides information about ASP.NET.

Sun JSP page (<http://java.sun.com/products/jsp/>) provides information about JavaServer Pages (JSP).

The PHP pages (<http://www.php.net/>) provide information about PHP: Hypertext Preprocessor.

The MySQL site (<http://www.mysql.com/>) provides information about MySQL.

Entities Table (<http://www.bbsinc.com/iso8859.html>) lists the entity names used in ISO 8859-1 (Latin-1).

The Dynamic HTML Events page (http://msdn.microsoft.com/workshop/author/dhtml/reference/events.asp#om40_event) provides information about events in Microsoft Internet Explorer.

Macromedia ColdFusion product page (<http://www.macromedia.com/software/coldfusion/>) provides information about ColdFusion.

Macromedia JRun Server product page (<http://www.macromedia.com/software/jrun/>) provides information about the JRun Java application server.

IBM WebSphere page (<http://www.ibm.com/software/webservers/appserv/>) provides information about the WebSphere application server.

Chili!Soft product page (<http://www.Chilisoft.com/products/>) provides information about the Sun Chili!Soft ASP application server.

JavaScript Bible, by Danny Goodman (IDG Books), comprehensively covers the JavaScript 1.2 language.

JavaScript: The Definitive Guide, by David Flanagan (O'Reilly & Associates), provides reference information for every JavaScript function, object, method, property, and event handler.

CGI Scripts for Fun and Profit (<http://www.hotwired.lycos.com/webmonkey/99/26/index4a.html>) is an article from the Hotwired Webmonkey site on incorporating ready-made CGI (Common Gateway Interface) scripts into your pages.

The CGI Resource Index (<http://www.cgi-resources.com/>) is a repository of all things related to CGI, including ready-made scripts, documentation, books, and even programmers for hire.

The Common Gateway Interface site (<http://hoohoo.ncsa.uiuc.edu/cgi/>) includes an introduction to CGI.

CHAPTER 2

Exploring the Workspace

To get the most out of your Macromedia Dreamweaver MX experience, you should understand the basic concepts behind the Dreamweaver workspace and how to choose options, use inspectors and panels, and set preferences that fit your work style best.

This chapter contains the following topics:

- “Using the Dreamweaver workspace” on page 31
- “Using windows and panels in Dreamweaver” on page 37
- “Using Dreamweaver with other applications” on page 49
- “Customizing Dreamweaver: Basics” on page 50

Using the Dreamweaver workspace

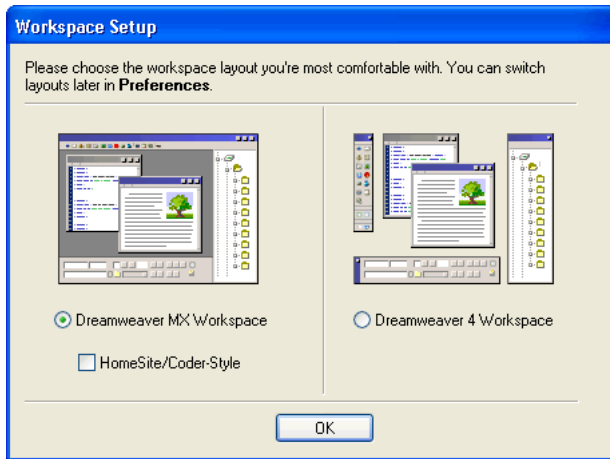
This section describes the primary elements of the workspace in both Windows and Macintosh, and explains how to choose an initial workspace layout in Windows.

In Windows, Dreamweaver MX provides two workspace layouts to choose between: an all-in-one-window integrated layout and a floating layout much like that of Dreamweaver 4.

On the Macintosh, only the floating layout is available. See “About the floating workspace layout (Windows and Macintosh)” on page 35.

Choosing a workspace layout (Windows only)

In Windows, the first time you start Dreamweaver, a dialog box appears that allows you to choose a workspace layout. If you change your mind later, you can switch to a different workspace using the Preferences dialog box.



To choose a workspace layout:

Select one of the following layouts:

Dreamweaver MX Workspace is an integrated workspace using MDI (Multiple Document Interface), in which all Document windows and panels are integrated into one larger application window, with the panel groups docked on the right. Recommended for most users.

Note: Most of this user guide assumes that you're using the integrated Dreamweaver MX workspace.

Dreamweaver MX Workspace, HomeSite/Coder-Style is the same integrated workspace, but with the panel groups docked on the left, in a layout similar to that used by Macromedia HomeSite and Macromedia ColdFusion Studio, and with Document windows showing Code view by default. This layout is recommended for HomeSite or ColdFusion Studio users and other hand-coders who want a familiar workspace layout. To choose this layout, select the Dreamweaver MX Workspace option, then select the HomeSite/Coder-Style option.

Note: You can dock panel groups on either side of the workspace in either of the integrated workspace layouts.

Dreamweaver 4 Workspace is a workspace layout similar to the one used in Dreamweaver 4, with each document in its own separate floating window. Panel groups are docked together, but are not docked into a larger application window. Recommended only for Dreamweaver 4 users who prefer to use a more familiar workspace.

To switch to a different workspace after you've chosen one:

- 1 Choose Edit > Preferences.
- 2 Select the General category in the Category list on the left, if it isn't already selected.
- 3 Click the Change Workspace button.

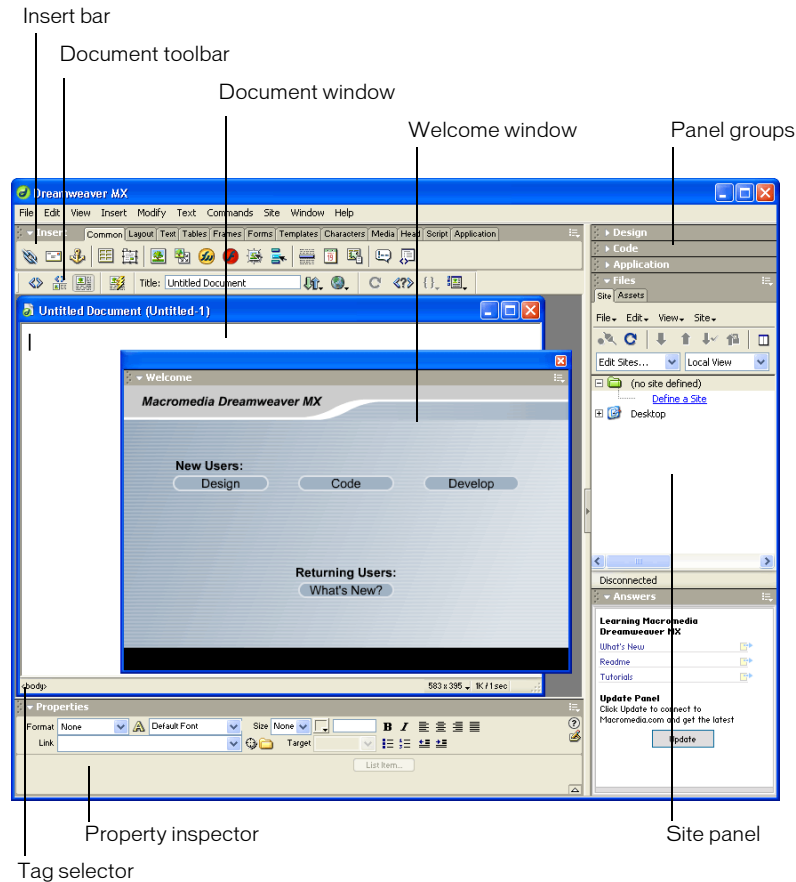
- 4 Select a workspace layout and click OK.

An alert message appears to tell you that the new layout will appear after you restart Dreamweaver.

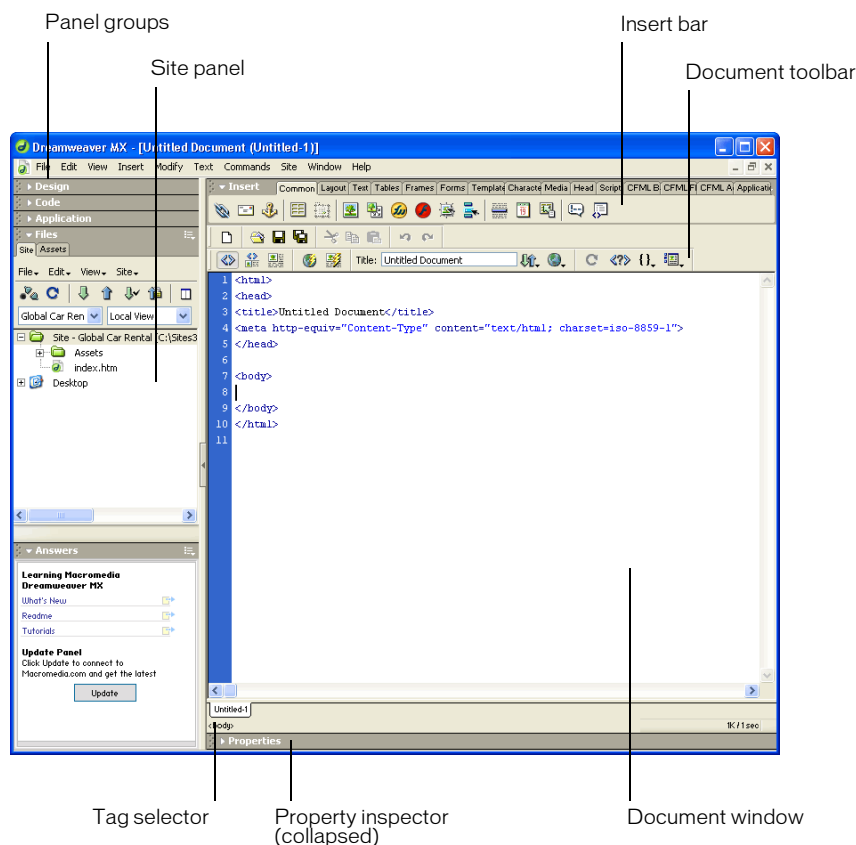
- 5 Click OK to dismiss the alert message.
- 6 Click OK again to dismiss the Preferences dialog box.
- 7 Exit Dreamweaver and restart it.

About the integrated Dreamweaver MX workspace (Windows only)

In the integrated Dreamweaver MX workspace in Windows, all windows and panels are integrated into a single larger application window.



The HomeSite/Coder-style workspace has the same elements, but with the panel groups docked on the left side of the main window instead on the right side. In this workspace layout, the Property inspector is collapsed by default, and the Document window appears in Code view by default.

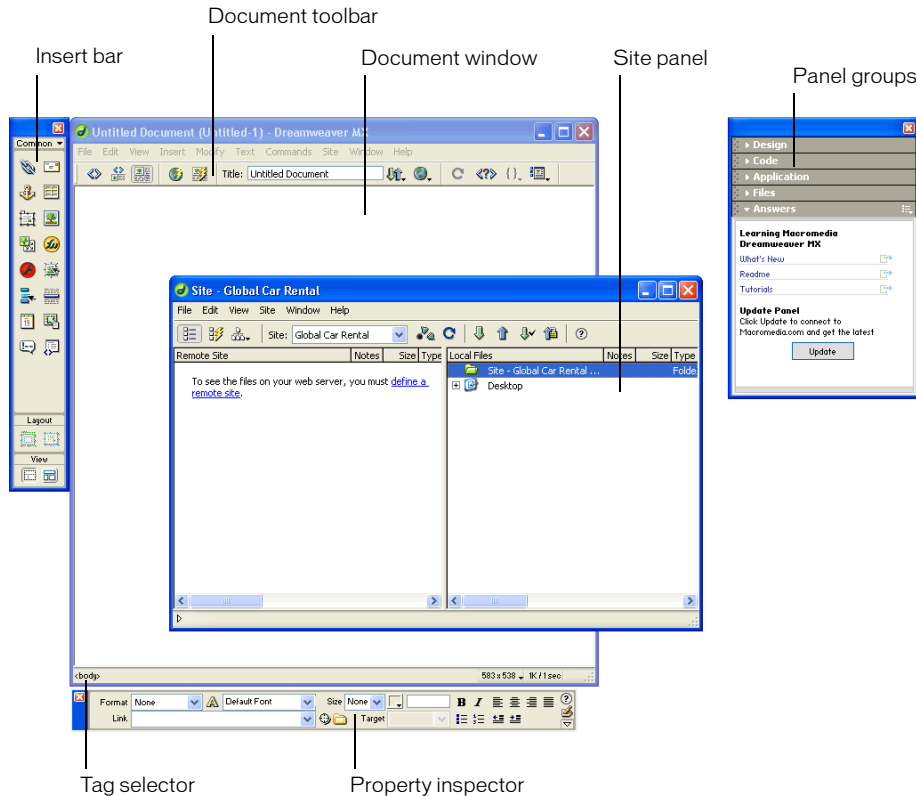


The integrated workspace is not supported on the Macintosh. The workspace on the Macintosh consists of floating windows and panels. Windows users can switch to this floating workspace by using the Preferences dialog box; for more information, see “Choosing a workspace layout (Windows only)” on page 32.

For a brief overview of the windows and panels, see “About windows and panels” on page 36. For more details about specific windows and panels, see “Using windows and panels in Dreamweaver” on page 37.

About the floating workspace layout (Windows and Macintosh)

In the Dreamweaver 4-style workspace layout, each document is in its own individual window. Panel groups are initially docked together, but can be undocked into their own windows.



In the floating workspace, windows “snap” automatically to each other, to the sides of the screen, and to the Document window as you drag or resize them. This makes it easier to move and arrange the different floating panels and windows in Dreamweaver.

Windows users can switch to an all-in-one-window integrated workspace by using the Preferences dialog box; for more information, see “Using the Dreamweaver workspace” on page 31.

For a brief overview of the windows and panels, see “About windows and panels” on page 36. For more details about specific windows and panels, see “Using windows and panels in Dreamweaver” on page 37.

About windows and panels

This section briefly describes some elements that appear in both the floating and integrated workspace layouts. For more information on specific windows and panels, see “Using windows and panels in Dreamweaver” on page 37.

The Welcome window provides tips on setting up your workspace for various purposes, and information on new features for those who have used previous versions of Dreamweaver.

The Insert bar contains buttons for inserting various types of “objects,” such as images, tables, and layers, into a document. Each object is a piece of HTML code that allows you to set various attributes as you insert it. For example, you can insert an image by clicking the Image icon in the Insert bar. If you prefer, you can insert objects using the Insert menu instead of the Insert bar.

The Document toolbar contains buttons and pop-up menus that provide different views of the Document window (such as Design view and Code view), various viewing options, and some common operations such as previewing in a browser.

The Standard toolbar (not displayed in the default workspace layout) contains buttons for common operations from the File and Edit menus: New, Open, Save, Save All, Cut, Copy, Paste, Undo, and Redo. To display the Standard toolbar, choose View > Toolbars > Standard.

The Document window displays the current document as you create and edit it.

The Property inspector lets you view and change a variety of properties for the selected object or text. Each kind of object has different properties. (Not expanded by default in Coder-style workspace layout.)

The tag selector in the status bar at the bottom of the Document window shows the hierarchy of tags surrounding the current selection in Design view. Click any tag in the hierarchy to select that tag and all its contents. (Not visible in Code view, so not visible by default in Coder-style workspace layout.) For more information, see “Editing code with the tag selector” on page 221.

Panel groups are sets of related panels grouped together under one heading. To expand a panel group, click the expander arrow at the left of the group’s name; to undock a panel group, drag the gripper at the left edge of the group’s title bar.

The Site panel allows you to manage the files and folders that make up your site. For more information, see “About the Site panel” on page 70. It also provides a view of all the files on your local disk, much like Windows Explorer (Windows) or the Finder (Macintosh).

The Launcher bar (not displayed in the default workspace layout) displays buttons in the status bar for opening and closing your most frequently used panels and inspectors. To display the Launcher bar and specify which icons appear in it, see “Setting Panels preferences” on page 47.

Dreamweaver provides many other panels, inspectors, and windows not shown here, such as the History panel and the Code inspector. Most panels can be docked together into panel groups. To open Dreamweaver panels, inspectors, and windows, use the Window menu. A check mark next to an item in the Window menu indicates that the named item is currently open (though it may be hidden behind other windows). To display an item that isn’t currently open, choose the item name from the menu.

If you can’t find a panel, inspector, or window that’s marked as open, choose Window > Arrange Panels to neatly lay out all open panels.

Using windows and panels in Dreamweaver

This section explains how to use a variety of specific windows, toolbars, panels, inspectors, and other elements of the Dreamweaver workspace.

About the Document window

The Document window shows the current document. You can choose any of the following views:

- Design view, a design environment for visual page layout, visual editing, and rapid application development. In this view, Dreamweaver displays a fully editable, visual representation of the document, similar to what you would see viewing the page in a browser. You can configure the Design view to display dynamic content while you're working on the document (see "Viewing live data in Design view" on page 476).
- Code view, a hand-coding environment for writing and editing HTML, JavaScript, server-language code—such as Microsoft Active Server Pages (ASP) or ColdFusion Markup Language—and any other kind of code. For more information, see Chapter 13, "Coding in Dreamweaver," on page 189.
- Code and Design view allows you to see both Code view and Design view for the same document in a single window.

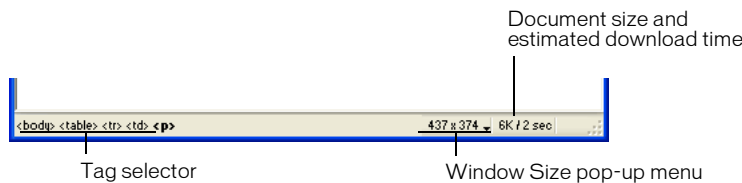
To toggle quickly between the two views, press Control+tilde (~) (Windows) or Command+backquote (`) (Macintosh).

When the Document window has a title bar, the title bar displays the page title and, in parentheses, the file's path and filename. After the filename, Dreamweaver displays an asterisk if you've made changes that you haven't saved yet.

When the Document window is maximized in the integrated workspace layout (Windows only), it has no title bar; in that case the page title and the file's path and filename appear in the title bar of the main workspace window.

About the status bar

The status bar at the bottom of the Document window provides additional information about the document you are creating.



The **tag selector** shows the hierarchy of tags surrounding the current selection. Click any tag in the hierarchy to select that tag and all its contents. Click <body> to select the entire body of the document. To set the class or id attributes for a tag in the tag selector, right-click (Windows) or Control-click (Macintosh) the tag and choose a class or ID from the context menu.

The **Window Size** pop-up menu (visible only when Design view is showing) lets you resize the Document window to predetermined or custom dimensions. See "Resizing the Document window" on page 38.

To the right of the Window Size pop-up menu are the estimated document size and estimated download time for the page, including all dependent files such as images and other media files. See “Setting download time and size” on page 467.

For information on creating a new document (and on using the predesigned documents supplied with Dreamweaver), see Chapter 5, “Setting Up a Document,” on page 107.

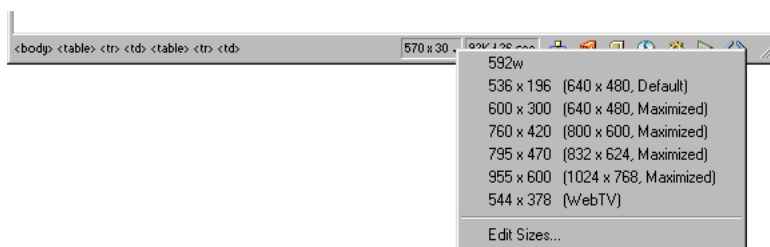
Resizing the Document window

The status bar displays the current Document window’s current dimensions (in pixels). To set the window size to fit any of several common monitor sizes, click the window size and choose a size from the Window Size pop-up menu. For less precise resizing, use your operating system’s standard methods of resizing windows, such as dragging the lower right corner of a window. In the integrated workspace (Windows only), you can maximize a Document window so that it fills the entire document area of the integrated window.

Note: You can’t resize a Document window when it is maximized in the integrated workspace.

When a Document window is maximized, tabs appear at the bottom of the Document window area showing the filenames of all open documents. To switch to a document, click its tab.

To design a page that looks its best at a specific size, you can adjust the Document window to any of the predetermined sizes, edit those predetermined sizes, or create new sizes.



Note: The window size shown reflects the inside dimensions of the browser window, without borders; the monitor size is listed in parentheses. For example, you would use the size “536 x 196 (640 x 480, Default)” if your visitors are likely to be using Microsoft Internet Explorer or Netscape Navigator in their default configurations on a 640 x 480 monitor.

To resize the Document window to a predetermined size:

Choose one of the sizes from the pop-up menu at the bottom of the Document window.

To change the values listed in the Window Size pop-up menu:

- 1 Choose Edit Sizes from the Window Size pop-up menu.
- 2 Click any of the width or height values in the Window Sizes list and type a new value.
To make the Document window adjust only to a specific width (leaving the height unchanged), select a height value and delete it.
- 3 Click the Description text box to enter descriptive text about a specific size.
- 4 Click OK to save the change and return to the Document window.

To add a new size to the **Window Size pop-up menu**:

- 1 Choose Edit Sizes from the Window Size pop-up menu.
- 2 Click the blank space below the last value in the Width column.
- 3 Enter values for Width and Height. To set the Width or Height only, simply leave one field empty.
- 4 Click the Description field to enter descriptive text about the size you added.
- 5 Click OK to save the change and return to the Document window.

For example, you might type **SVGA** or **average PC** next to the entry for an 800 x 600 pixel monitor, and **17-in. Mac** next to the entry for an 832 x 624 pixel monitor. Most monitors can be adjusted to a variety of pixel dimensions.

To maximize a **Document window (Windows integrated workspace only)**:

Click the Maximize button in the Document window's title bar.

Setting Status Bar preferences

To set preferences for the status bar, choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X) and select Status Bar from the Category list on the left. (For more information, see "About the status bar" on page 37.)

Window Sizes lets you customize the window sizes that appear in the status bar's pop-up menu. See "Resizing the Document window" on page 38.

Connection Speed determines the connection speed (in kilobits per second) used to calculate the download size. The download size for the page is displayed in the status bar. When an image is selected in the Document window, the image's download size is displayed in the Property inspector.

Using the Launcher bar

The Launcher bar appears in the status bar, and contains buttons for opening and closing various panels, inspectors, and windows.

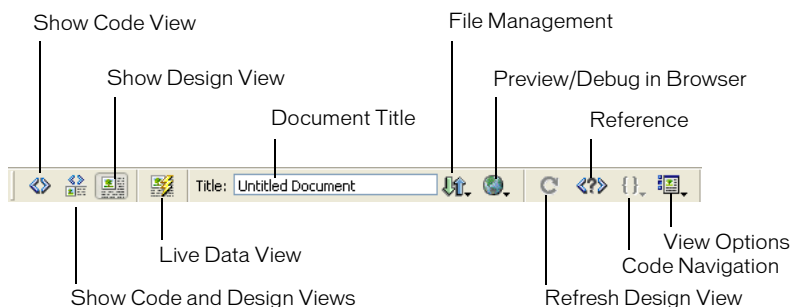


To display the Launcher bar, enable the Show Icons in Panels and Launcher option in Panels preferences. You can specify which items appear in the Launcher bar. For more information, see "Setting Panels preferences" on page 47.

Using the Document toolbar

The Document toolbar contains buttons that let you toggle between different views of your document quickly: Code, Design, a split view that shows both Code and Design view, and Live Data view (see "Viewing live data in Design view" on page 476). The toolbar also contains some common commands and options related to viewing the document, transferring it between the local and remote sites, and debugging JavaScript code.

To view or hide the Document toolbar, choose View > Toolbars > Document.



The following options appear in the Document toolbar:

- **Show Code View** displays only the Code view in the Document window.
- **Show Code and Design Views** displays Code view in part of the Document window and Design view in another part.

When you select this combined view, the option Design View on Top becomes available in the View Options menu. Use this option to specify which view appears at the top of your Document window.

- **Show Design View** displays only the Design view in the Document window.
You can also choose a view in the View menu.
- **Title** allows you to enter a title for your document, to be displayed in the browser's title bar. If your document already has a title, it appears in this field.
- **File Management** displays the File Management pop-up menu.
- **Preview/Debug in Browser** allows you to preview or debug your document in a browser. Choose a browser from the pop-up menu.

To add a browser to the menu or change the browsers listed, choose Edit Browser List.

Refresh Design View refreshes the document's Design view after you make changes in Code view.

Changes you make in Code view don't automatically appear in Design view until you perform certain actions, such as saving the file or clicking this button.

- **Reference** displays the Reference panel.
The Reference panel contains reference information on HTML, CSS, JavaScript, CFML, ASP, and JSP. For more information, see "Accessing language references" on page 200.
- **Code Navigation** allows you to navigate through your JavaScript code. For more information, see "Using the JavaScript debugger" on page 209.
- **View Options** allows you to set options for Code view and Design view, including which view should appear above the other.

For more information on Code view options, see "Setting viewing preferences" on page 176.

The Design view options allow you to hide all visual aids (such as table, layer, and frame borders) at once, or to show and hide individual visual aids as desired. These options also allow you to view head content and visual guides. For more information about these options, which can also be found in the View menu, see “Selecting elements in the Document window” on page 116, “Viewing and editing head content” on page 119, and “Using visual guides in the design process” on page 117.

When both Code view and Design view are showing, both sets of options are available in the menu.

Using the Standard toolbar

The Standard toolbar contains buttons for common operations from the File and Edit menus: New, Open, Save, Save All, Cut, Copy, Paste, Undo, and Redo. Use these buttons just as you would use the equivalent menu commands. For more information about operations such as Open and Save, see Chapter 5, “Setting Up a Document,” on page 107.

To display the Standard toolbar, choose View > Toolbars > Standard.

Using context menus

Dreamweaver makes extensive use of context menus, which provide convenient access to the most useful commands and properties related to the object or window you’re working with. Context menus list only those commands that pertain to the current selection.

To use a context menu:

- 1 Right-click (Windows) or Control-click (Macintosh) the object or window.

The context menu for the selected object or window appears.

- 2 Choose a command from the context menu.



Using the Insert bar

The Insert bar contains buttons for creating and inserting objects such as tables, layers, and images. The buttons are organized into tabs.

To show or hide the Insert bar:

Choose **Window > Insert**.

To expand or collapse the Insert bar:



Click the expander arrow in the left corner of the Insert bar's title bar.

To change the Insert bar's orientation, do one of the following (Macintosh only):



- To change the Insert bar from horizontal to vertical, click the vertical-orientation icon near the upper right corner.



- To change the Insert bar from vertical to horizontal, click the horizontal-orientation icon near the lower right corner.

To show the buttons in a particular tab, do one of the following:

- If the Insert bar is in its tabbed horizontal configuration, click a tab.
- If the Insert bar is in its vertical configuration (Macintosh only), click the category name at the top of the bar and choose a category from the pop-up menu that appears.

To insert an object:

- 1 Select the appropriate tab in the Insert bar.
- 2 Click an object button or drag the button's icon into the Document window.

Depending on the object, a corresponding object-insertion dialog box may appear, prompting you to browse to a file or specify parameters for an object.

To bypass the object-insertion dialog box and insert an empty placeholder object:

Control-click (Windows) or Option-click (Macintosh) the button for the object.

For example, to insert a placeholder for an image without specifying an image file, Control-click or Option-click the Image button.

Note: This procedure does not bypass all object-insertion dialog boxes. Many objects, including navigation bars, layers, Flash buttons, and framesets, do not insert placeholders or default-valued objects.

The Insert bar contains several tabs: Common, Layout, Text, Tables, Frames, Forms, Templates, Characters, Media, Head, Script, and Application. Additional tabs appear when the current document contains server code, such as ASP or CFML documents.



- The **Application** tab allows you to insert dynamic elements such as recordsets, repeated regions, and record insertion and update forms. For information about dynamic content, see Chapter 33, "Adding Dynamic Content to Web Pages," on page 519.

- The **Characters** tab contains special characters such as the copyright symbol, curved quotation marks, and trademark symbols. Note that some of these symbols may not display correctly in versions 3.0 and earlier of the Internet Explorer and Netscape Navigator browsers.
- The **Common** tab contains buttons for creating and inserting the most commonly used objects, such as images, tables, and layers.
- The **Forms** tab contains buttons for creating forms and inserting form elements.
- The **Frames** tab contains common frameset layouts.
- The **Head** tab contains buttons for adding various head elements, such as meta and base tags.
- The **Layout** tab allows you to insert tables and layers, and lets you choose between two views of tables: Standard (default) view and Layout view. When Layout view is selected, you can use the Dreamweaver layout tools: Draw Layout Cell and Draw Layout Table. For more information, see Chapter 17, “Laying Out Pages in Layout View,” on page 241.
- The **Media** tab contains buttons for inserting animated or interactive media objects such as Flash buttons and text, Java applets, and ActiveX objects. For more information, see Chapter 22, “Inserting Media,” on page 319.
- The **Script** tab allows you to insert a script, a noscript section, or a server-side include.
- The **Tables** tab allows you to insert an entire table or a specific table tag (such as tr, th, or td).
- The **Templates** tab allows you to insert editable, optional, and repeating regions in template files. For more information, see “About Dreamweaver templates” on page 435.
- The **Text** tab allows you to insert a variety of text- and list-formatting tags, such as b, em, p, h1, and ul.

Note: Although some buttons in the Text tab look similar to some buttons in the text Property inspector, they serve different functions. Buttons in the Text tab merely insert code, and do not reflect the current state of the selection; if the selection is bold text, the Bold button in the Property inspector appears selected, while the Bold button in the Text tab does not.

- Server-code tabs, available only for pages that use a particular server language, include **ASP**, **ASP.NET**, **CFML Basic**, **CFML Flow**, **CFML Advanced**, **JSP**, and **PHP**. Each of these tabs provides server-code objects that you can insert in Code view.

For information on the objects in each tab, see the related topics in Dreamweaver Help.

If not all objects in a given tab are visible at once, a small arrow appears at the lower left corner of the Insert bar; to view the rest of the objects in the tab, click the arrow.

When you click an object button, Dreamweaver inserts code into the document. In some cases, the code is inserted immediately; in other cases, a tag editor (see “Editing tags with tag editors” on page 194) or other dialog box appears, allowing you to specify further information before the code is inserted. For some objects, no dialog box appears if you insert the object in Design view, but a tag editor appears if you insert the object in Code view. For a few objects, inserting the object in Design view causes Dreamweaver to switch to Code view before inserting the object.

Some objects, such as named anchors, are not visible when the page is viewed in a browser window. To display icons in Design view that mark the locations of such invisible objects, choose View > Visual Aids > Invisible Elements. To select invisible objects in the Design view, click their icons. For more information, see “About invisible elements” on page 116.

Some of the General preference settings affect the Insert bar. To modify these preferences, choose **Edit > Preferences** or **Dreamweaver > Preferences (Mac OS X)** and then select **General** from the Category list on the left.

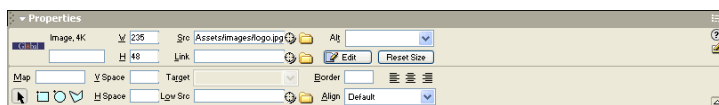
- When you insert objects such as images, tables, scripts, and `head` elements, a dialog box prompts you for additional information. You can suppress these dialog boxes by turning off the **Show Dialog When Inserting Objects** option or by holding down the **Control** key (Windows) or the **Option** key (Macintosh) while creating the object. When you insert an object with this option off, the object is given default attribute values. Use the Property inspector to change object properties after inserting the object.
- The **Insert Bar** option lets you display the contents of the Insert bar as icons only, text only, or text and icons.

You can modify any object in the Insert bar or create your own objects; see “Customizing Dreamweaver” on the Macromedia Support Center at http://www.macromedia.com/go/customizing_dreamweaver.

Using the Property inspector

The Property inspector lets you examine and edit properties for the currently selected page element, such as text or an inserted object. You can select page elements in either **Design** view or **Code** view.

To show or hide the Property inspector, choose **Window > Properties**.



Most changes you make to properties are immediately applied in the Document window. For some properties, however, changes are not applied until you click outside the property-editing text fields, press **Enter** (Windows) or **Return** (Macintosh), or press **Tab** to switch to another property.

The contents of the Property inspector vary depending on the element selected. For information on specific properties, select an element in the Document window and then click the **Help** icon in the upper right corner of the Property inspector.

The Property inspector initially displays most of the properties of the selected element. Click the expander arrow in the lower right corner of the Property inspector to collapse the Property inspector to show only the most commonly used properties.

Tip: In a few cases, certain obscure properties may not appear even in the expanded Property inspector; in these cases, use **Code** view or the **Code** inspector to code these properties by hand, or select the tag in **Code** view and choose **Modify > Edit Tag**.

Managing panels and panel groups

Panels in Dreamweaver are grouped together into panel groups. Each panel group can be expanded or collapsed, and can be docked or undocked with other panel groups. Panel groups can also be docked to the integrated application window (Windows integrated workspace only). This makes it easy to access the panels you need without cluttering your workspace. Panels within a panel group appear as tabs.

Note: When a panel group is floating (undocked), a narrow blank bar appears at the top of the panel group. In this documentation, the term “panel group’s title bar” refers to the area where the panel group’s name appears, rather than to this narrow blank bar.

To expand or collapse a panel group, do one of the following:

- Click the expander arrow on the left side of the panel group's title bar.
- Click the panel group's title.

To select a panel within an expanded panel group:

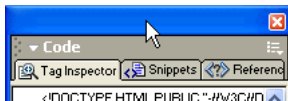
Click the panel's tab.

To undock a panel group:

- Drag the panel group by its gripper (on the left side of the panel group's title bar) until its outline indicates that it's no longer docked.

To dock a panel group to other panel groups (floating workspace) or to the integrated window (Windows only):

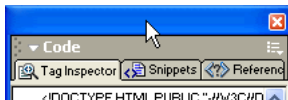
Drag the panel group by its gripper until its outline indicates that it's docked.



Most panels can be docked only to either the left or the right of the Document window area in the integrated workspace, while others (such as the Property inspector and the Insert bar) can be docked only to the top or bottom of the integrated window.

To drag a floating (undocked) panel group without docking it:

Drag the panel group by the bar above its title bar. The panel group doesn't dock as long as you don't drag it by its gripper.



To see a panel group's Options menu:

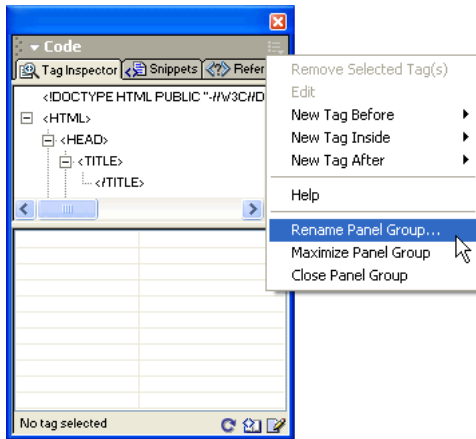
- Expand the panel group by clicking its expander arrow.

The Options menu is visible only when the panel group is expanded.

Tip: Some options are available in the panel group's context menu even when the group is collapsed; right-click (Windows) or Control-click (Macintosh) the panel group's title bar to view the context menu.

To rename a panel group:

- 1 Choose Rename Panel Group from the Options menu on the right side of the panel group's title bar.



- 2 Enter a new name and click OK.

To undock a panel from a panel group:

Drag the panel by its tab until its outline indicates that it's no longer docked.

The panel appears in a new panel group of its own.

To dock a panel in a panel group:

Drag the panel by its tab until its outline indicates that it's docked.

To maximize a panel group, do one of the following:

- Choose Maximize Panel Group from the Options menu in the panel group's title bar.
- Double-click anywhere in the panel group's title bar.

The panel group grows vertically to fill all of the available vertical space.

To close a panel group, making it disappear completely:

Choose Close Panel Group from the Options menu in the panel group's title bar.

The panel group disappears from your screen.

To open a panel group that isn't visible on your screen:

Choose the name of a panel from the Window menu.

To change the size of the entire set of panel groups (floating workspace only):

Drag to resize the set of panel groups just as you would drag to resize any window in your operating system. For example, you can drag the resize area at the lower right corner of the set of panel groups.

Setting Panels preferences

Use Panels preferences to specify which panels and inspectors always appear in front of the Document window, and which ones may be obscured by the Document window. You can also use Panels preferences to specify whether icons appear in the tabs for panels, whether the Launcher bar appears, and which panels and inspectors appear in the Launcher bar.

To set preferences for panels, choose Edit > Preferences and select Panels from the Category list on the left. Then select any of the following options:

Always on Top lets you specify which panels always appear in front of the Document window. By default, all panels, inspectors, and other such windows always “float” in front of the Document window. If you deselect an item in this list, then you can move the Document window in front of that item.

For example, to allow the Document window to obscure the Property inspector, deselect the Properties option. The Property inspector now appears in front of the Document window only when the Property inspector is active.

To allow the Document window to obscure any floating panels that you have added by customizing Dreamweaver, deselect All Other Panels.

Note: The Always on Top option doesn't apply in the integrated workspace when all panels are docked.

Show Icons in Panels and Launcher determines whether the Launcher bar appears. When this option is selected, the Launcher bar appears in the status bar area, and a small icon appears in the tab for each panel. (See “About the status bar” on page 37.) The buttons in the Launcher bar open panels and inspectors.

Show in Launcher specifies which items appear in the Launcher bar.

To specify which items appear in the Launcher bar:

- 1 To add an item to the Launcher bar, click the plus (+) button.
- 2 To remove an item from the Launcher bar, select the item and click the minus (-) button.
- 3 To change the order of the items in the Launcher bar, select an item in the list and click an arrow button.

For example, to move an item to the right in the Launcher bar, move the item down in the list.

- 4 Click OK.

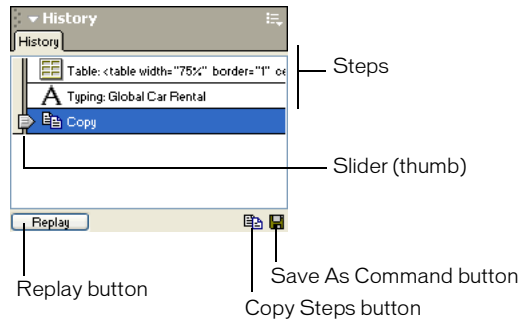
The Launcher bar changes to show the items you specified.

About the Site panel

The Site panel allows you to define a site, manage the local files in your site, upload and download files from a remote site, and browse the files on your local disk outside of your site. For details on using the Site panel, see Chapter 4, “Managing Your Site,” on page 69.

About the History panel

The History panel shows a list of the steps you've performed in the active document since you created or opened that document, up to a specified maximum number of steps. (The History panel doesn't show steps you've performed in other frames, in other Document windows, or in the Site panel.) It allows you to undo one or more steps; it also allows you to replay steps and to create new commands to automate repetitive tasks.



The slider, or thumb, in the History panel initially points to the last step that you performed.

Using the History panel

The History panel keeps track of every step of your work in Dreamweaver. You can use the History panel to undo multiple steps at once.

If you want to undo the last operation you performed in a document, choose Edit > Undo, just as you would do in any other application. (The name of the Undo command changes in the Edit menu to reflect the last operation you performed.)

The History panel also lets you replay steps you've already performed and automate tasks by creating new commands. For more information, see "About automating tasks" on page 120.

To open the History panel:

Choose Window > Others > History.

To undo the last step:

Drag the History panel slider up one step in the list. This has the same effect as choosing Edit > Undo.

The undone step turns gray.

To undo multiple steps at once, do one of the following:

- Drag the slider to point to any step.
- Click to the left of a step along the path of the slider; the slider scrolls automatically to that step, undoing steps as it scrolls.

Note: To scroll automatically to a particular step, you must click to the left of the step; clicking the step itself selects the step. Selecting a step is different from going back to that step in your undo history.

As with undoing a single step, if you undo a series of steps and then do something new in the document, you can no longer redo the undone steps; they disappear from the History panel.

To set the number of steps that the History panel retains and shows:

- 1 Choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X) and select General from the Category list.
- 2 Enter a number for Maximum Number of History Steps.

The default value should be sufficient for most users' needs. The higher the number, the more memory the History panel requires. This can affect performance and slow your computer down significantly. When the History panel reaches this maximum number of steps, the earliest steps are discarded.

Note: You can't rearrange the order of steps in the History panel. Don't think of the History panel as an arbitrary collection of commands; think of it as a way to view the steps you've performed, in the order in which you performed them.

To erase the history list for the current document:

In the History panel's context menu, choose Clear History.

This command also clears all undo information for the current document; after choosing Clear History, you will be unable to undo the steps that are cleared. (Note that Clear History does not undo steps; it merely removes the record of those steps from Dreamweaver's memory.)

About the Answers panel

The Answers panel provides quick access to information that helps you work effectively with Dreamweaver. This includes tutorials, TechNotes, Dreamweaver extensions, and other useful content.

To get the latest Dreamweaver information from macromedia.com, click the Update button.

Using Dreamweaver with other applications

Dreamweaver accommodates your web design and development process by making it easy for you to work with other applications. For information about working with other applications such as browsers, HTML editors, image editors, and animation tools, see the following topics:

- For information about using Dreamweaver with other HTML editors, such as HomeSite or BBEdit, see "Using an external HTML editor with Dreamweaver" on page 187.
- You can specify preferred browsers for previewing your site. See "Previewing pages in browsers" on page 463.
- You can launch an external image editor, such as Macromedia Fireworks, from within Dreamweaver. See "Using an external image editor" on page 302.
- You can configure Dreamweaver to launch a different editor for each file type. See "Launching an external editor for media files" on page 320.
- For information about adding interactivity to your site using Macromedia Flash, see "About Flash content" on page 322.
- To learn how to add animation to your site using Macromedia Shockwave movies, see "Inserting Shockwave movies" on page 326.
- For information about using ColdFusion, see Chapter 6, "Setting Up a Web Application," on page 127.

Customizing Dreamweaver: Basics

There are some basic techniques you can use to customize Dreamweaver to suit your needs without knowing complex code or editing text files. For example, you can set preferences, create your own keyboard shortcuts, and add extension to Dreamweaver.

For information about customizing configuration files by hand, see “Customizing Dreamweaver” on the Macromedia Support Center at http://www.macromedia.com/go/customizing_dreamweaver.

Setting preferences

Dreamweaver has preference settings that control the general appearance and behavior of the Dreamweaver user interface as well as options related to specific features such as layers, style sheets, displaying HTML and JavaScript code, external editors, and previewing in browsers. Information about specific preference options is provided throughout this guide with the associated feature or topic.

This guide describes only the most common of the preference options. For information on a specific preference option not covered here, see the corresponding topic in Dreamweaver Help.

Setting General preferences

The General preferences control the general appearance of Dreamweaver. To change these preferences, choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X), and then click General. The General preferences are divided into two subcategories: Document Options and Editing Options. For detailed information on these preferences, see Dreamweaver Help.

Use and in place of and <i>

specifies that Dreamweaver should apply the `strong` tag whenever you perform an action that would normally apply the `b` tag, and should apply the `em` tag whenever you perform an action that would normally apply the `i` tag. Such actions include clicking the Bold or Italic buttons in the text Property inspector and choosing Text > Style > Bold or Text > Style > Italic. To use the `b` and `i` tags in your documents, deselect this option.

Note: The World Wide Web Consortium discourages use of the `b` and `i` tags; the `strong` and `em` tags provide more semantic information than the `b` and `i` tags do.

Setting Fonts preferences

Use Fonts preferences to set the fonts that Dreamweaver uses to display each font encoding. A document's encoding determines how the document is displayed in a browser. Dreamweaver font settings let you view a given encoding in the font and size you prefer without affecting how the document appears when viewed by others in a browser. See “Setting Up a Document” on page 107 for information on setting a default encoding for new documents.

To set the fonts to use in Dreamweaver for each type of encoding:

- 1 Choose Edit > Preferences and click Fonts in the Category list.
- 2 Select an encoding type (such as Western (Latin1) or Japanese) from the Font Settings list, then choose fonts to use within Dreamweaver for that encoding, using the font pop-up menus below the Font Settings list.

Note: The fonts you choose do not affect how the document appears in a visitor's browser.

For more information about Fonts preferences, see Dreamweaver Help.

Using the Keyboard Shortcut Editor

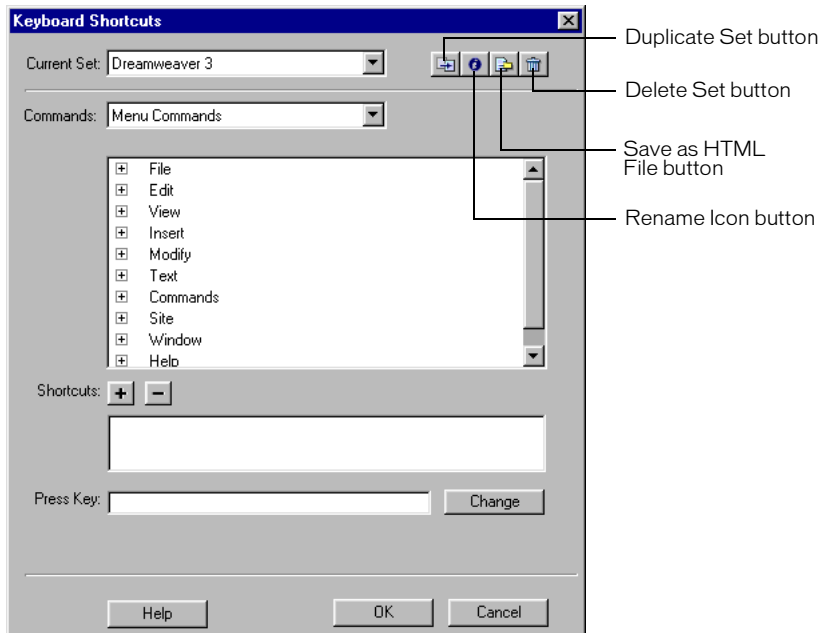
Use the Keyboard Shortcut Editor to create your own shortcut keys, edit existing shortcuts, or use a predetermined set of shortcuts.

Note: Charts showing all the keyboard shortcuts in the default Dreamweaver configuration are available on the Macromedia website at http://www.macromedia.com/go/dreamweaver_mx_shortcuts.

To edit keyboard shortcuts:

- 1 Choose Edit > Keyboard Shortcuts.

The Keyboard Shortcuts dialog box appears.



- 2 Set any of the following options:

- **Current Set** allows you to choose a set of predetermined shortcuts included with Dreamweaver, or any custom set you've defined. The predetermined sets are listed at the top of the menu. For example, if you are familiar with the shortcuts found in HomeSite or BBEdit, you can use those shortcuts by choosing the corresponding predetermined set.
- **Commands** allows you to select a category of commands to edit. For example, you can edit menu commands, such as the Open command, or code editing commands, such as Balance Braces.
- **The command list** displays the commands associated with the category you selected from the Commands pop-up menu, along with the assigned shortcuts. The Menu Commands category displays this list as a tree view that replicates the structure of the menus. The other categories list the commands by name (such as Quit Application), in a flat list.
- **Shortcuts** displays all shortcuts assigned to the selected command.

- **Add Item (+)** adds a new shortcut to the current command. Click this button to add a new blank line to the Shortcuts text box. Enter a new key combination and click Change to add a new keyboard shortcut for this command. You can assign two different keyboard shortcuts for each command; if there are already two shortcuts assigned to a command, the Add Item button does nothing.
- **Remove Item (-)** removes the selected shortcut from the list of shortcuts.
- **Press Key** displays the key combination you enter when you're adding or changing a shortcut.
- **Change** adds the key combination shown in the Press Key text box to the list of shortcuts, or changes the selected shortcut to the specified key combination.
- **Duplicate** duplicates the current set. Give the new set a name; the default name is the current set's name with the word *copy* appended.
- **Rename** renames the current set.
- **Save as HTML File** saves the current set in an HTML table format for easy viewing and printing. You can open the HTML file in your browser and print the shortcuts for easy reference.
- **Delete** deletes a set. (You cannot delete the active set.)

To remove a shortcut from a command:

- 1 From the Commands pop-up menu, select a command category.
The Commands list displays the commands in that category.
- 2 In the Commands list, select a command.
The shortcuts assigned to the command appear in the Shortcuts text box.
- 3 Select a shortcut.
- 4 Click the Remove Item (-) button.

To add a shortcut to a command:

- 1 From the Commands pop-up menu, select a command category.
The Commands list displays the commands in that category.
- 2 In the Commands list, select a command.
The shortcuts assigned to the command appear in the Shortcuts text box.
- 3 Prepare to add a shortcut by doing one of the following:
 - If there are fewer than two shortcuts already assigned to the command, click the Add Item (+) button. A new blank line appears in the Shortcuts text box and the insertion point moves to the Press Key text box.
 - If there are already two shortcuts assigned to the command, select one of them. (That one will be replaced by the new shortcut.) Then click in the Press Key text box.
- 4 Press a key combination.
The key combination appears in the Press Key text box.

Note: If there is a problem with the key combination (for example, if the key combination is already assigned to another command), an explanatory message appears just below the Shortcuts field and you may be unable to add or edit the shortcut.

5 Click Change. The new key combination is assigned to the command.

To edit an existing shortcut:

1 From the Commands pop-up menu, select a command category.

The Commands list displays the commands in that category.

2 In the Commands list, select a command.

The shortcuts assigned to the command appear in the Shortcuts text box.

3 Select a shortcut to change.

4 Click in the Press Key text box and enter a new key combination.

5 Click the Change button to change the shortcut.

Note: If there is a problem with the key combination (for example, if the key combination is already assigned to another command), an explanatory message appears just below the Shortcuts field and you may be unable to add or edit the shortcut.

Adding extensions to Dreamweaver

Extensions are new features that you can add easily to Dreamweaver. You can use many types of extensions; for example, there are extensions that let you reformat tables, connect to back-end databases, or help you write scripts for browsers.

Note: To install extensions that all users can use in a multiuser operating system, you must be logged in as Administrator (Windows) or root (Mac OS X). For more information about multiuser systems, see "Customizing Dreamweaver in multiuser systems" on page 54.

To find the latest extensions for Dreamweaver, use the Macromedia Exchange website at <http://www.macromedia.com/exchange/dreamweaver/>. Once there, you can log in and download extensions (many of which are free), join discussion groups, view user ratings and reviews, and install and use the Extension Manager. You must install the Extension Manager before you can install extensions.

The Extension Manager is a separate application that lets you install and manage extensions in Macromedia applications. Launch the Extension Manager from Dreamweaver by choosing Commands > Manage Extensions.

To install and manage extensions:

1 On the Macromedia Exchange website, click the download link for an extension.

Your browser may allow you to choose to open and install it directly from the site or save it to disk.

- If you are opening the extension directly from the site, the Extension Manager handles the installation automatically.
 - If you are saving the extension to disk, a good place to save the extension package file (.mxd) is the Downloaded Extensions folder within the Dreamweaver application folder on your computer.
- 2 Double-click the extension package file or open the Extension Manager and choose File > Install Extension.

The extension is installed in Dreamweaver. Some extensions aren't accessible until Dreamweaver has restarted; you may be prompted to quit and restart the application.

Use the Extension Manager to remove extensions or to see more information about an extension.

Customizing Dreamweaver in multiuser systems

You can customize Dreamweaver to suit your needs even in a multiuser operating system such as Windows NT, Windows 2000, Windows XP, or Mac OS X. Dreamweaver prevents any user's customized configuration from affecting any other user's customized configuration. To accomplish this goal, the first time you run Dreamweaver in one of the multiuser operating systems that it recognizes, the application creates copies of a variety of configuration files for you. These user configuration files are stored in a folder belonging to you. For example, in Windows XP they're stored in C:\Documents and Settings\username\Application Data\Macromedia\Dreamweaver MX\Configuration (which may be inside a hidden folder), and in Mac OS X they're stored inside your home folder; specifically, in Users/username/Library/Application Support/Macromedia/Dreamweaver MX/Configuration.

Note: In older operating systems (Windows 98, Windows ME, and Mac OS 9.x), a single set of Dreamweaver configuration files is shared by all users, even if the operating system is configured to support multiple users.

If you reinstall or upgrade Dreamweaver, Dreamweaver automatically makes backup copies of existing user configuration files, so that if you've customized those files by hand, you still have access to the changes you made.

CHAPTER 3

Planning and Setting Up Your Site

A website is a set of linked documents with shared attributes, such as related topics, a similar design, or a shared purpose.

Macromedia Dreamweaver MX is a site creation and management tool, so you can use it to create complete websites, in addition to individual documents. For best results, design and plan your website before you create any of the pages that the site will contain.

Note: If you can't wait to start creating documents, you can try out some of the Dreamweaver document-creation tools and make a sample document (see "Creating a new blank document" on page 109). But don't start serious document development until you've set up a site.

The first step in creating a website is planning (see "About site planning and design" on page 55). The next step is to set up the basic structure of the site (see "Setting up a Dreamweaver site" on page 59). When you begin to develop site content, consider basing your site on a Dreamweaver template (see "About Dreamweaver templates" on page 435 and "Creating a Dreamweaver template" on page 440).

If you already have a site on a web server and you want to begin using Dreamweaver to edit that site, see "Editing a Dreamweaver site" on page 65.

This chapter includes the following sections:

"About site planning and design" on page 55

"Setting up a Dreamweaver site" on page 59

"Setting up a local folder" on page 61

"Setting up a remote folder" on page 63

"Editing a Dreamweaver site" on page 65

"Editing existing websites in Dreamweaver" on page 66

About site planning and design

In Dreamweaver, the term *site* can refer either to a website or to a local storage location for the documents belonging to a website. When you begin thinking about creating a website, you should follow a series of planning steps to make sure your site succeeds. Even if you are just creating a personal home page that only friends and family will see, it is to your advantage to plan the site carefully to make sure everyone will be able to use it successfully.

Determining your goals

Deciding what your site goals are is the very first step you should take when you create a website. Ask yourself or your client questions about the site. What do you hope to accomplish by having a website? Write down your goals so that you remember them as you go through the design process. Goals help you focus and target your website to your particular needs.

A website that provides news about a specific subject should have a very different look and navigation than that of a website that sells products. The complexity of your goals will affect the navigation, the authoring tools that you use (Flash, Director, and so on), and even the look and feel of your site.

Choosing a target audience

After you have decided what you want to accomplish with your website, you need to decide who you want to visit your site. This may seem to be a silly question, since most people want everyone to visit their website. However, it is difficult to create a website that every single person in the world will be able to use. People use different browsers, connect at different speeds, and may or may not have media plug-ins. All of these factors can affect the use of your site. That is why you need to determine a target audience.

Think about the people who will be attracted to your website, or who you hope to attract. What kinds of computers do you think they will be using? What platform might be the dominant one (Macintosh, Windows, Linux, and so on)? What is the average connection speed (33.6 Kbps modem or DSL)? What kinds of browsers and monitor sizes will they be using? Are you creating an intranet site where everyone will be using the same computer OS and browser? All of these factors can greatly affect the way your web page appears to visitors.

Once you choose the audience and have determined what types of computers, connection speeds, and browsers they will be using, you can target your design.

For example, say your target audience is predominantly made up of Windows users with 17-inch monitors, using Microsoft Internet Explorer 3.0 or above. As you design your web page, you should test that your site works best in Internet Explorer on a Windows computer with a screen size of 800 x 600 pixels. Fewer viewers may use Netscape Navigator on a Macintosh platform, but you should still make sure your site works on those machines, although it may not display as precisely as it does for your target audience.

Creating sites for browser compatibility

As you create your site, you should be aware of the variety of web browsers your visitors are likely to use. Where possible, design sites for maximum browser compatibility, given other design constraints.

There are over two dozen different web browsers in use, most of which have been released in more than one version. Even if you target only Netscape Navigator and Microsoft Internet Explorer, which are used by the majority of web users, be aware that not everyone uses the very latest versions of those browsers. If your site is on the web, sooner or later someone will visit it using Netscape Navigator 2.0, or the browser that AOL provides its customers, or a text-only browser such as Lynx.

There are some circumstances under which there's no need to create sites that are cross-browser-compatible. For example, if your site is available only on your company's intranet, and you know that all of your company's employees use the same browser, you can optimize your site to rely on features of that browser. Similarly, if you're creating HTML content to be distributed on CD-ROM, and you distribute a browser on the CD, you can assume that all of your customers have access to that particular browser.

Under most circumstances, for websites designed for public viewing, it's a good idea to make your site viewable in as many browsers as possible. Pick one or two browsers as your target browsers, and design the site for those browsers, but try exploring the site in other browsers now and then to avoid including too much incompatible content. You can also post a message on a discussion board to ask others to view your site. This can be a good way to get feedback from a wide audience.

The more sophisticated your site is—in terms of layout, animation, multimedia content, and interaction—the less likely it is to be cross-browser-compatible. Not all browsers can run JavaScript, for example. A page of plain text that uses no special characters will probably display well in any browser, but such a page may have much less aesthetic appeal than one that uses graphics, layout, and interaction effectively. Try to strike a balance between designing for maximum effect and designing for maximum browser compatibility.

One useful approach is to provide multiple versions of certain important pages, such as your site's home page. For example, you can design both a framed version and a frameless version of such a page. You can then include in your web page a behavior that automatically shifts visitors without frame-capable browsers to the frameless version. For more information, see “Using the behavior actions that come with Dreamweaver” on page 354.

Organizing the site structure

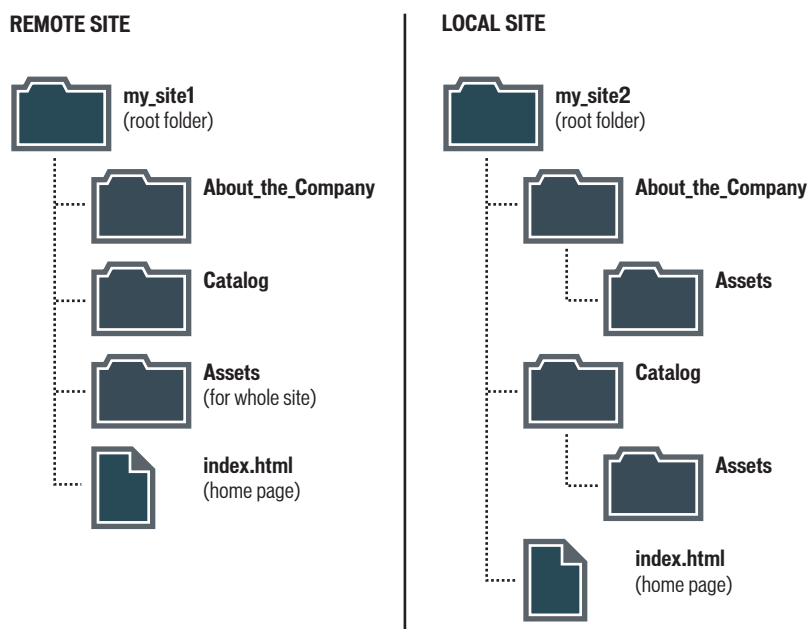
Organizing your site carefully from the beginning can save you frustration and time later on. If you begin creating documents without thinking about where in your folder hierarchy they should go, you may end up with a huge, unwieldy folder full of files, or with related files scattered through a lot of similarly named folders.

The usual way to set up a site is to create a folder on your local disk that contains all the files for your site (referred to as the local site), and to create and edit documents within that folder. You then copy those files to a web server when you are ready to publish your site and allow the public to view it. This approach is better than creating and editing files on the live public website itself, because it allows you to test changes in the local site before making them publicly viewable, and then when you're finished, you can upload the local site files and update the entire public site at once.

To coordinate your local site with Dreamweaver, see “Setting up a Dreamweaver site” on page 59. Once you set up the local site with Dreamweaver, you can more easily manage your site files, track links, update pages, and more.

Break down your site into categories Put related pages in the same folder. For example, your company press releases, contact information, and job postings might go all in one folder, and your online catalog pages in another folder. Use subfolders where necessary. This type of organization will make your site easier to maintain and navigate.

Decide where to put items such as images and sound files It's convenient to place all your images, for example, in one location, so that when you want to insert an image into a page you know where to find it. Designers sometimes place all of the non-HTML items to be used on a site in a folder called Assets. That folder may contain other folders—for example, an Images folder, a Shockwave folder, and a Sound folder. Or you might have a separate Assets folder for each group of related pages on your site, if there aren't many assets shared among such groups.



Use the same structure for local and remote sites Your local site and your remote website should have exactly the same structure. If you create a local site using Dreamweaver and then upload everything to the remote site, Dreamweaver ensures that the local structure is precisely duplicated in the remote site.

Creating your design look

You save a lot of time later in the process if you plan your design and layout before you actually begin working in Dreamweaver. It can be as simple as creating a mock-up drawing of how you want the site layout to look on a piece of paper. A more advanced approach would be to create a composite drawing of your site using software such as Macromedia FreeHand or Fireworks. The important thing is to have a mock-up of your layout and design so you can follow it later as you build your site.

It is important to maintain consistency in your page layout and design. You want users to be able to click through the pages in your site without getting confused because all the pages have a different look, or the navigation is in a different place on each page.

Designing the navigation scheme

Another area where planning pays off is navigation. As you design your site, think about the experience you want your visitors to have. Think about how a visitor to your site will be able to move from one area to another. Consider the following points:

“**You are here**” information enables visitors to easily know where they are in your site and how to return to your top-level page.

Searching and indexes make it easy for visitors to find any information they are looking for.

Feedback provides a way for visitors to contact the webmaster (if appropriate) if something is wrong with the site, and to contact other relevant people associated with the company or the site.

Design the way your navigation will look. Navigation should be consistent throughout your site. If you place a navigation bar across the top of your home page, try to keep it there for all the linked pages.

Dreamweaver has two navigation tools that you can use to create your navigation scheme. For more information, see “Linking and Navigation” on page 399.

Planning and gathering assets

Once you know what your design and layout will look like, you can create and gather the assets that you will need. Assets can be items such as images, text, or media (Flash, Shockwave, and more). Make sure you have all of these items gathered and ready to go before you begin developing your site. Otherwise, you’ll have to continually stop development to find an image or to create a button.

If you are using images and graphics from a clip-art site or someone else is creating them, make sure you collect them and put them in a folder on your site (see “Organizing the site structure” on page 57). If you are creating the assets yourself, make sure you create them all before you start development, including any images you need if you are using rollovers. Then organize your assets so you can access them easily while creating your site in Dreamweaver.

Dreamweaver can make it easier for you to reuse page layouts and page elements in various documents by using templates and libraries. However, it is easier to create new pages with templates and libraries than it is to apply them to existing documents.

Use templates if many of your pages will use the same layout. Plan and design a template for that layout, and then you can create new pages based on that template. If you decide to change the layout for all the pages, you can simply change the template.

Note: There are certain restrictions on what changes you can make to documents that are based on templates. Templates are best used in collaborative environments, to ensure that everyone is using the same page layout. Library items may provide more flexibility for use outside of collaborative environments.

Use library items if you know that a certain image or other content will appear on many pages throughout your site; design that content ahead of time and make it a library item. Then if you change that item later, the updated version appears on all pages that use it.

For more information on reusing page layouts and elements, see “Managing Site Assets, Libraries, and Templates” on page 419.

Setting up a Dreamweaver site

After you plan your site structure (see “Organizing the site structure” on page 57), or if you already have an existing site, you must designate a new site in Dreamweaver before you start developing.

A Dreamweaver site is a way to organize all of the documents associated with a website. You can think of it as a project. You need to set up a site for each website you develop. Organizing your files in a site enables you to use Dreamweaver with FTP to upload your site to the web server, automatically track and maintain your links, manage files, and share files. You cannot take full advantage of Dreamweaver features unless you define a site.

A Dreamweaver site consists of as many as three parts, depending on your environment and the type of website you are developing:

- **Local folder** is your working directory. Dreamweaver refers to this folder as your “local site.” For more information, see “Setting up a local folder” on page 61.
- **Remote folder** is where you store your files, depending on your environment, for testing, production, collaboration, or so on. Dreamweaver refers to this folder as your “remote site.” For more information, see “Setting up a remote folder” on page 63).
- **Folder for dynamic pages** is the folder where Dreamweaver processes dynamic pages. For more information, see “Specifying where dynamic pages can be processed” on page 134.

To get started, you can set up a Dreamweaver site completely or you can just start with the first step, setting up your local folder. You must at least set up a local folder before you start using Dreamweaver.

There are two ways you can set up a Dreamweaver site: use the Site Definition Wizard, which steps you through the setup process, or use the Site Definition Advanced settings, which enable you to set up local, remote, and testing folders individually, as necessary.

To set up a Dreamweaver site:

- 1 Select Site > New Site.

The Site Definition dialog box appears.

- 2 Click the Basic tab to use the Site Definition Wizard, or click the Advanced tab to use the Advanced settings.

- 3 Complete the Dreamweaver site setup process:

- For the Site Definition Wizard, answer the questions, and click Next to advance through the setup process (see “Using the Site Definition Wizard” on page 60).
- For Advanced settings, complete the Local Info category (see “Setting up a local folder” on page 61), the Remote Info category (see “Setting up a remote folder” on page 63), and the Testing Server category (see “Specifying where dynamic pages can be processed” on page 134), as necessary.

Note: Users who are new to Dreamweaver are encouraged to use the Site Definition Wizard; experienced Dreamweaver users might prefer to use the Advanced settings.

Using the Site Definition Wizard

You can use the Site Definition Wizard to set up a Dreamweaver site. The Site Definition Wizard steps you through the site setup process. This method is recommended for users who are new to Dreamweaver.

The wizard has three sections. Each section might have multiple screens, referred to as parts, depending on the settings you select. The three main sections include the following:

- **Editing Files** is where you set up your local folder. For more information, see “Setting up a local folder” on page 61.
- **Testing Files** is where you set up a folder for Dreamweaver to process dynamic pages. For more information, see “Specifying where dynamic pages can be processed” on page 134.
- **Sharing Files** is where you set up your remote folder. For more information, see “Setting up a remote folder” on page 63.

At the top of the dialog box, the bold section name indicates where you are in the setup process. You might not need to set up folders in all three sections. Each section asks you questions to help you determine your needs.

To use the Site Definition Wizard to set up a Dreamweaver site:

1 Select Site > New Site.

The Site Definition dialog box appears.

2 Click the Basic tab to use the Site Definition Wizard.

3 Answer questions on each screen, and then click Next to advance. Click Back to return to a previous screen, if necessary.

Setting up a local folder

The local folder is your working directory for your Dreamweaver site (see “Setting up a Dreamweaver site” on page 59). This folder can be on your local machine or it can be on a network server. It is the place where you store your “in progress” files for a Dreamweaver site.

When you set up a local folder you establish a Dreamweaver site. You might also add remote and testing folders (see “Setting up a remote folder” on page 63 and “Creating a root folder for the application” on page 132), but you need to at least set up a local folder before you start developing in Dreamweaver.

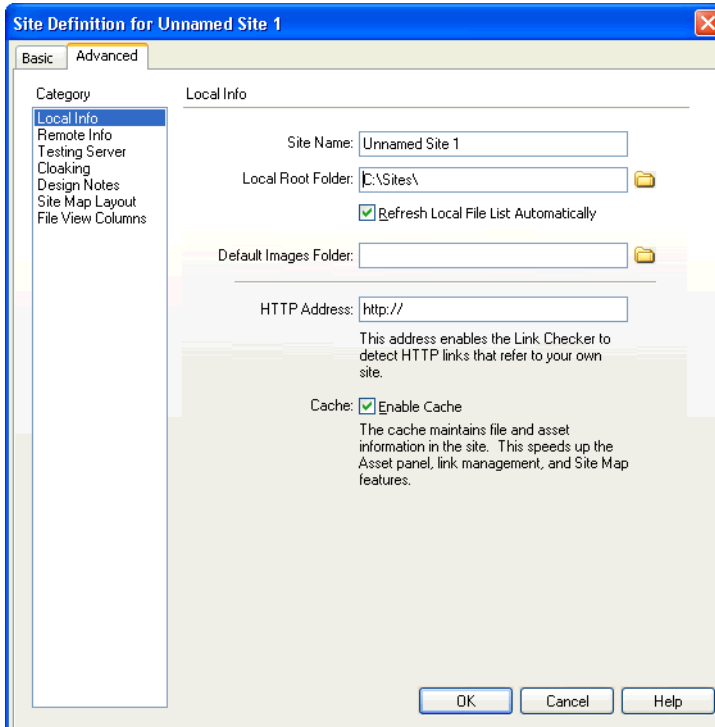
Setting up a local folder:

- 1 Choose Site > New Site.

The Site Definition dialog box appears.

- 2 Click the Advanced button, if the Advanced settings aren't showing.

The Advanced tab of the Site Definition dialog displays the Local Info category options.



- 3 Enter the Local Info options.

For more information, click the Help button in the dialog box.

- 4 Click OK.

Dreamweaver creates the initial site cache, and the new Dreamweaver site appears in the Site panel. For more information about the site panel and site management, see “Managing Your Site” on page 69.

Later, when you're ready to publish the site on a remote server, you need to add additional information about the site. For information, see “Setting up a remote folder” on page 63.

Setting up a remote folder

After you set up a local folder for a Dreamweaver site (see “Setting up a local folder” on page 61), you can set up a remote folder. The local and remote folders enable you to transfer files between your local disk and web server; this makes it easy for you to manage files in your Dreamweaver sites.

Depending on your environment, the remote folder is where you store files for testing, collaboration, production, deployment, or a number of other scenarios. Typically, your remote folder is on the machine where your web server is running.

Note: You don’t need to specify a remote folder if the folder you specified as your local folder in the Local Info category (see “Setting up a local folder” on page 61) is the same folder you created for your site files on the system running your web server. This implies the web server is running on your local computer.

Determine how you will access the remote folder and note the connection information. When you’ve gathered this information, use the Edit Sites command to set up your remote folder. If you encounter problems, see “Troubleshooting remote folder setup” on page 65.

After you’ve set up a remote folder, you can connect to it, and then browse and manage files in your Dreamweaver site. If you are developing a dynamic site, you need to add a folder for processing dynamic pages (see “Specifying where dynamic pages can be processed” on page 134).

To set up a remote folder:

- 1 Choose Site > Edit Sites.

The Edit Sites dialog box appears.

- 2 Select an existing Dreamweaver site.

If you have not defined any Dreamweaver sites, create a local folder before proceeding (see “Setting up a local folder” on page 61).

- 3 Click Edit.

The Site Definition dialog box appears.

- 4 Click the Advanced button if Dreamweaver displays the Site Definition Wizard.

- 5 Select Remote Info from the Category list on the left.

- 6 Choose an Access option.

For more information, click the Help button in the dialog box.

- 7 Click OK.

To connect to a remote folder:

In the Site panel, choose Site > Connect or click the Connect to Remote Host button on the Site panel toolbar.

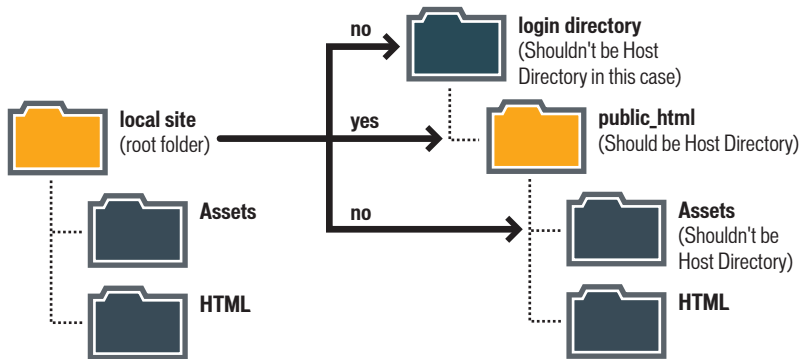
Note: If your site uses FTP with SSH to access your remote folder, when you try to connect to your remote server, a command prompt launches for you to login to the SSH server. Click OK in the Dreamweaver dialog box after you log in.

To disconnect from a remote folder:

In the Site panel, choose Site > Disconnect or click the Disconnect button on the Site panel toolbar.

Choosing the host directory for FTP access

If you choose FTP to access your remote folder (see “Setting up a remote folder” on page 63), you must determine the remote folder’s host directory. The host directory you specify should correspond to the root folder of the local folder. The following diagram shows a sample local folder on the left and a sample remote folder on the right.



If the structure of your folder site doesn’t match the structure of your local folder, Dreamweaver uploads your files to the wrong place, and the files aren’t visible to visitors to the site. Also, your image and link paths are broken.

The remote root directory must exist before Dreamweaver can connect to it. If you don’t have a root directory for your remote folder, create one or ask the server’s administrator to create a root directory for you.

If you’re uncertain what to enter in the Host Directory text box, contact the server’s administrator. Or, you can try leaving the text box blank. On some servers, your root directory is the same as the directory you first connect to with FTP. To find out, connect to the server. If a folder with a name like public_html, or www, or your login name, appears in the Remote File view in your Site panel, that’s probably the directory you should use in the Host Directory text box.

Using Secure Shell for FTP access

When you select FTP in the Site Definition dialog box to access your remote folder, in Windows you can choose to use SSH (Secure Shell). To activate SSH in Windows, click the Help button in the dialog box.

To use SSH in Macintosh, download the Macintosh SSH client from the The Dreamweaver Support Center website at <http://www.macromedia.com/support/dreamweaver/>.

Troubleshooting remote folder setup

A web server can be configured in a wide variety of ways. The following list provides information on some common issues you may encounter in setting up a remote folder, and how to resolve them:

- The Dreamweaver FTP implementation may not work properly with certain proxy servers, multilevel firewalls, and other forms of indirect server access.

If you encounter problems with FTP access, ask your local system administrator for help.

- For the Dreamweaver FTP implementation, you must connect to the remote system's root folder. (In many applications, you can connect to any remote directory, then navigate through the remote file system to find the directory you want.)

Be sure that you indicate the remote system's root folder as the host directory.

If you have problems connecting, and you've specified the host directory using a single slash (/), you might need to specify a relative path from the directory you are connecting to and the remote root folder.

For example, if the remote root folder is a higher level directory, you may need to specify a ../../ for the host directory.

- File and folder names that contain spaces and special characters often cause problems when transferred to a remote site.

Use underscores in place of spaces, and avoid special characters in file and folder names wherever possible. In particular, colons, slashes, periods, and apostrophes in file or folder names can cause problems. Special characters in file or folder names may also sometimes prevent Dreamweaver from creating a site map.

- If you encounter problems with long filenames, rename them with shorter names. On the Macintosh, filenames cannot be more than 31 characters long.
- Note that many servers use symbolic links (UNIX), shortcuts (Windows), or aliases (Macintosh) to connect a folder on one part of the server's disk with another folder elsewhere.

For example, the public_html subdirectory of your home directory on the server may really be a link to another part of the server entirely. In most cases, such aliases have no effect on your ability to connect to the appropriate folder or directory, but if you can connect to one part of the server but not another, there may be an alias discrepancy.

- If you encounter an error message such as "cannot put file," your remote folder may be out of space. Look at the FTP log for more detailed information.

In general, when you encounter a problem with an FTP transfer, examine the FTP log by choosing Window > Results > FTP Log in the Site panel (Windows) or Site > FTP Log (Macintosh).

Editing a Dreamweaver site

Use the Site Definition Advanced settings to edit your Dreamweaver sites.

To edit a Dreamweaver site, do one of the following:

- Choose Site > Edit Sites, select a site, and then click Edit.
- Choose Site > Open Site, and then select a site.

Editing existing websites in Dreamweaver

You can use Dreamweaver to edit existing sites, even if you didn't use Dreamweaver to create the original site. You can editing existing sites that are on your local system or sites that are on a remote system.

Editing an existing local website in Dreamweaver

You can use Dreamweaver to edit an existing website on your local disk, even if you didn't use Dreamweaver to create the original site.

To edit an existing local website:

1 Open the Site Definition dialog box by doing one of the following:

- Choose Site > Edit Sites, and then click New.
- Choose Site > Open Site > Define Sites.

The Site Definition dialog box displays the Local Info options.

2 Complete the dialog box.

For more information, click the Help button in the dialog box.

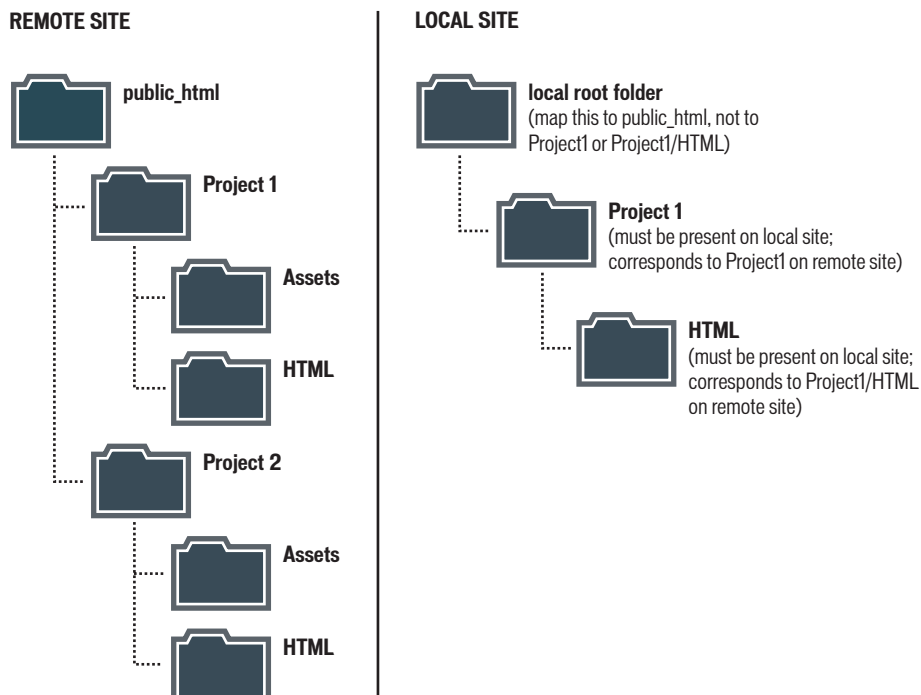
3 Click OK.

Editing an existing remote website in Dreamweaver

You can use Dreamweaver to copy an existing remote site (or any branch of a remote site) to your local disk and edit it there, even if you didn't use Dreamweaver to create the original site.

Even if you intend to edit only part of the remote site, you must locally duplicate the entire structure of the relevant branch of the remote site, from the remote site's root folder down to the files you want to edit.

For example, if your remote site's root folder, named `public_html`, contains two folders, `Project1` and `Project2`, and you want to work only on the HTML files in `Project1`, you don't need to download the files in `Project2`, but you must map your local root folder to `public_html`, not to `Project1`.



To edit an existing remote site:

- 1 Create a local folder to contain the existing site, then set it up as the local root folder for the site (see “Setting up a local folder” on page 61).
- 2 Set up a remote folder, using information about your existing site (see “Setting up a remote folder” on page 63). Make sure to choose the correct root folder for the remote site.
- 3 Click the Connect button in the Site panel to connect to the remote site.
- 4 Depending on how much of the remote site you want to edit, do one of the following:
 - If you want to work with the entire site, select the root folder of the remote site, and click Get to download the entire site to your local disk.
 - If you want to work with just one of the files or folders of the site, locate the file or folder in the Remote view of the Site panel, and click Get File(s) to download that file to your local disk.Dreamweaver automatically duplicates as much of the remote site's structure as is necessary to place the downloaded file in the correct part of the site hierarchy. When editing only one part of a site, you should generally choose to include dependent files.
- 5 Proceed as if creating a site from scratch: edit documents, preview and test them, and upload them again to the remote site.

CHAPTER 4

Managing Your Site

Macromedia Dreamweaver MX helps you organize the files in your local and remote folders (also called “sites”) using the Site panel. It lets you duplicate the structure of your local folder on a remote server, or duplicate a remote folder’s structure on your local system. The relative links you create in your local site continue to work after you transfer files to the remote site, because the structure of the two sites is identical.

You create a local site in Dreamweaver by using the New Site command to create a local root folder for the site or by making an existing folder the local root folder (see “Setting up a local folder” on page 61). You define a remote site when creating a new site, or add that information later using the Edit Sites command (see “Setting up a remote folder” on page 63).

Dreamweaver includes a number of features for managing a site and transferring files to and from a remote server. When you transfer files between local and remote sites, Dreamweaver maintains parallel file and folder structures between the sites. When transferring files between sites, Dreamweaver automatically creates necessary folders when they do not yet exist in a site. You can also synchronize the files between your local and remote sites; Dreamweaver copies files in both directions as necessary, and removes unwanted files when appropriate.

Dreamweaver contains features to make collaborative work on a website easier. You can check files in and out of a remote server so that other members of a web team can see who is working on a file. You can add Design Notes to your files to share information with team members about a file’s status, priority, and so on. You can also use the Workflow Reports feature to run reports on your site to display information on the check-in/check-out status, and to search for Design Notes attached to files.

Once you’ve published your site, you or someone on your team can continue to maintain it. Before and after publishing, you’ll also want to troubleshoot your site on an ongoing basis (for more information, see “Testing a Site” on page 461).

This chapter contains the following sections:

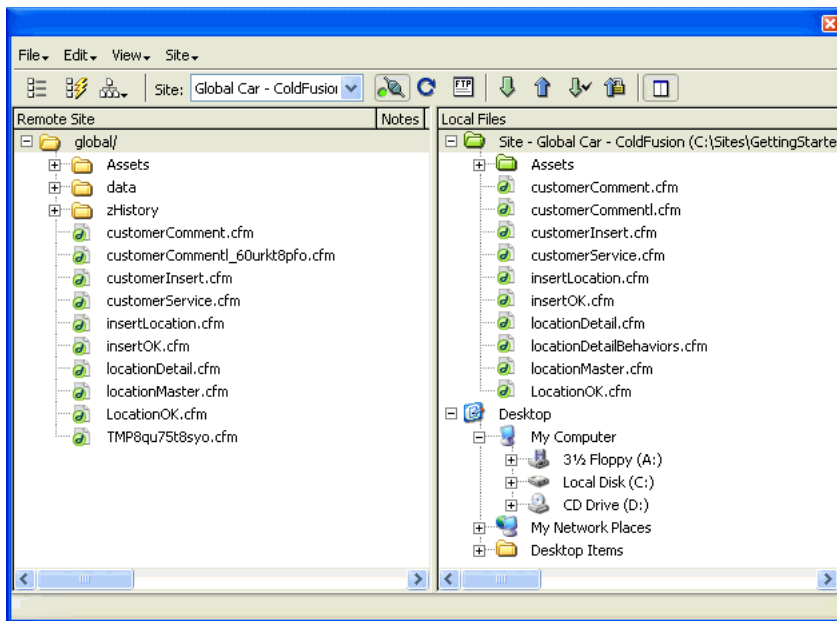
- “About the Site panel” on page 70
- “About the site map” on page 80
- “Importing and exporting sites” on page 86
- “Removing a site from your list of sites” on page 87
- “Using Check In/Check Out” on page 87
- “Getting and putting files” on page 90
- “Synchronizing the files on your local and remote sites” on page 93

- “Cloaking folders and files in your site” on page 94
- “Using Design Notes” on page 97
- “Using reports to improve workflow” on page 101
- “About the Sitespring panel in Dreamweaver” on page 103
- “Using the Sitespring panel” on page 104

About the Site panel

The Site panel enables you to view a site, including all local, remote, and testing server files associated with a selected site. The local site appears by default; change the Site panel layout to view the remote site or testing server (see “Changing Site panel layout” on page 76). Expand the site panel to get a split view (see “Using the Site panel” on page 71).

Note: On the Macintosh, the site panel is always in split view; it does not collapse.



Use the Site panel for standard file maintenance operations, such as:

- Creating new HTML documents
- Viewing, opening, and moving files
- Creating folders
- Deleting items
- Transferring files among local sites, remote sites, and testing servers
- Designing your site navigation with the site map (see “Setting up a remote folder” on page 63 and “Using the site map” on page 81)

The Site panel contains an integrated file browser. This file browser enables you to browse your local disk and network, in addition to the current site (see “Using the integrated file browser” on page 80).

Using the Site panel

The Site panel is integrated into the Dreamweaver integrated workspace by default “Exploring the Workspace” on page 31. You can move the panel or open and close it as necessary (see “Using windows and panels in Dreamweaver” on page 37). You can also expand the panel to get a split view of a site. In the split view, choose to view the local site and the remote site, testing server, or site map (“Changing Site panel layout” on page 76).

To open or close the Site panel:

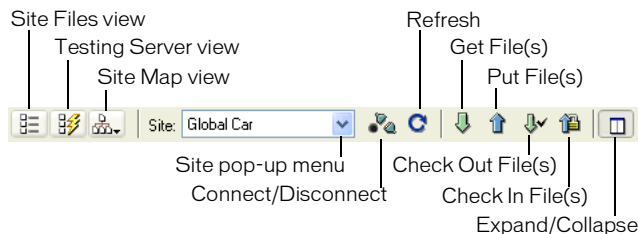
Choose Window > Site.

To expand or collapse the Site panel:

(Windows only) Click the Expand/Collapse button on the Site panel toolbar.

Note: If you click the Expand/Collapse button to expand the panel while it is docked, the panel maximizes so that you cannot work in the Document window. To return to the Document window, click the Expand/Collapse button again to collapse the panel. If you click the Expand/Collapse button to expand the panel while it is floating, the panel remains a floating panel and you can still work in the Document window. Before you can dock the panel again, you must first collapse the panel.

Use the following buttons and options on the Site panel toolbar to determine what the Site panel displays and to transfer files:



Note: The Site Files, Testing Server, and Site Map buttons appear only in the expanded Site panel.

Site Files view displays the file structure of the remote and local sites in the panes of the Site panel. (A preference setting determines which site appears in the left pane and which appears in the right pane; see “Setting Site preferences” on page 73.) Site Files view is the default view for the Site panel.

Testing Server view displays the directory structure of the testing server and the local site.

Site Map view displays a graphical map of your site based on how the documents are linked to one another. Hold this button down to choose Map Only or Map and Files from the pop-up menu.

The Site pop-up menu lists sites you’ve defined. To switch sites, choose a different site from the list. To add a site or edit the information for an existing site, choose Edit Sites from the bottom of the menu (see “Setting up a remote folder” on page 63).

Connect/Disconnect (FTP, RDS, WebDAV protocol, and SourceSafe) connects to or disconnects from the remote site. By default, Dreamweaver disconnects from the remote site if it has been idle for more than 30 minutes (FTP only).

To change the time limit, choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X), then select Site from the category list on the left.

Refresh refreshes the local and remote directory lists. Use this button to manually refresh the directory lists if you deselected either Refresh Local File List Automatically or Refresh Remote File List Automatically in the Site Definition dialog box (see “Setting up a remote folder” on page 63).

Get File(s) copies the selected files from the remote site or testing server to your local site (overwriting the existing local copy of the file, if any). If Enable File Check In and Check Out is turned on, the local copies are read-only; the files remain available on the remote site for other team members to check out. If Enable File Check In and Check Out is turned off, getting a file transfers a copy that has both read and write privileges.

The files Dreamweaver copies are the files you select in the active pane of the Site panel. If the Remote or Testing Server panes are active, the selected remote or testing server files are copied to the local site; if the Local pane is active, Dreamweaver copies the remote or testing server version of the selected local files to the local site. For more information, see “Getting files from a remote or testing server” on page 90.

Note: You can get remote and testing server files, but check-in and check-out functionality applies to remote files only.

Put File(s) copies the selected files from the local site to the remote site or testing server.

The files Dreamweaver copies are the files you select in the active pane of the Site panel. If the Local pane is active, the selected local files are copied to the remote site or testing server; if the Remote or Testing Server panes are active, Dreamweaver copies the local versions of the selected remote or testing server files to the remote site. For more information, see “Putting files on a remote or testing server” on page 91.

Note: You can put files on the remote and testing servers, but check-in and check-out functionality applies to remote files only.

If you are putting a file that doesn’t already exist on the remote site, and Enable File Check In and Out is on, Dreamweaver adds the file to the remote site as “checked out.” Click the Check In Files button if you want to add a file without the checked out status.

Check Out File(s) transfers a copy of the file from the remote server to your local site (overwriting the existing local copy of the file, if any) and marks the file as checked out on the server. This option is not available if Enable File Check In and Check Out in the Site Definitions dialog box is turned off for the current site. For more information, see “Checking files in to and out of a remote site” on page 89.

Check In File(s) transfers a copy of the local file to the remote server and makes the file available for editing by others. The local file becomes read-only. This option is not available if the Enable File Check In and Check Out option in the Site Definitions dialog box is turned off for the current site. For more information, see “Checking files in to and out of a remote site” on page 89.

The **Expand/Collapse** button (Windows only) expands or collapses the Site panel to display one or two panes.

Setting Site preferences

You can control Site panel file-transfer features in the Preferences dialog box.

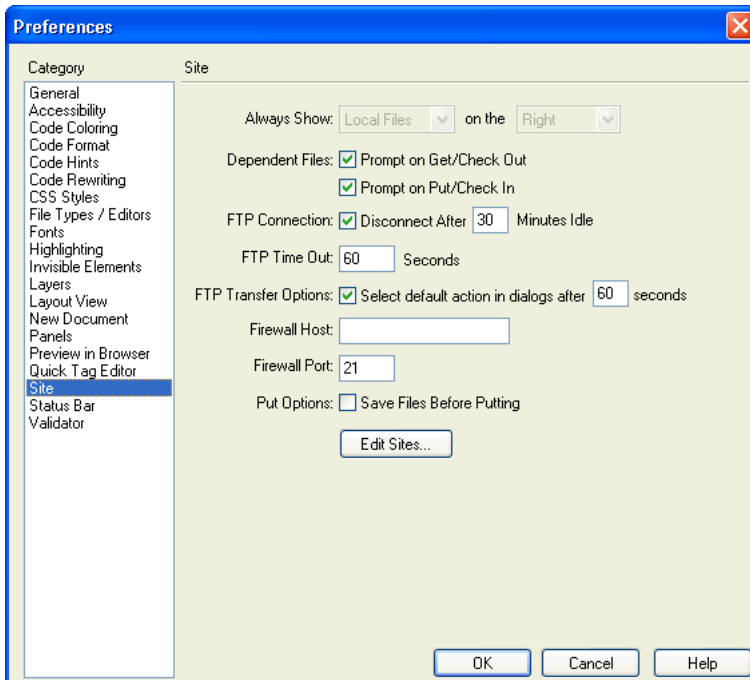
To edit Site panel preferences:

- 1 Choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X).

The Preferences dialog box appears.

- 2 Select Site from the category list on the left.

The Site preference options appear.



- 3 Change options as necessary.

For more information, click the Help button in the dialog box.

- 4 Click OK.

You can define whether the types of files that you transfer are transferred as ASCII (text) or binary, by customizing the FTPExtensionMap.txt file in the Dreamweaver/Configuration folder (on Macintosh, FTPExtensionMapMac.txt). For more information, see “Welcome to Dreamweaver” on page 17.

Changing file view columns

You can customize the columns displayed in the Site panel's Local and Remote Site lists. You can do any of the following:

- Reorder columns
- Add new columns (for a maximum of 10 columns)
- Delete columns
- Hide columns
- Designate columns to be shared with all users connected to a site

The default columns are Name, Notes, Size, Type, Modified, and Checked Out By. To sort by any column, click the column heading in the Site panel. Clicking a column more than once reverses the order (ascending or descending) by which Dreamweaver sorts the column.

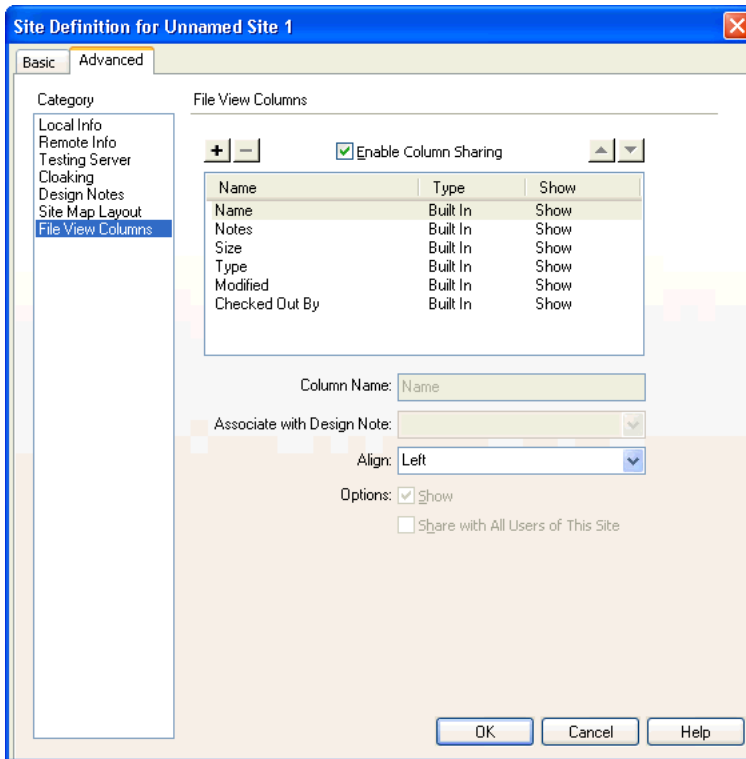
Note: You cannot delete a default column, rename a default column, or associate a Design Note with a default column. You can change the ordering and alignment of default columns, as well as hide them, with the exception of the Name column, which you cannot hide.

To add, delete, or change columns:

- 1 In the Site panel, do one of the following to access File Views Columns options:
 - Choose View > File View Columns (Windows) or Site > Site Files View > File View Columns (Macintosh).

- Choose Site > Edit Sites, select a site, and click Edit. Then, select File View Columns from the category list on the left in the Site Definition dialog box.

The Site Definition dialog box displays the File View Columns options.



- 2 Complete the dialog box.

For more information, click the Help button in the dialog box.

- 3 Click OK.

Related topic

“Changing Site panel layout” on page 76

Opening and viewing sites in the Site panel

When the Site panel is collapsed it displays the contents of the local site, the remote site, or the testing server as a list of files. When the Site panel is expanded, it displays the local site and either the remote site or testing server in a split view. When expanded, the Site panel can also display a visual site map of the local site.

Note: Before you can view a remote site or a testing server, you must set up a remote site or testing server (see “Setting up a remote folder” on page 63 or “Specifying where dynamic pages can be processed” on page 134). Before you can view a site map, you must set up a home page (see Using the site map).

To open an existing Dreamweaver site:

In the Site panel, select a site from the Site pop-up menu (this is where the current site name appears).

Note: To define a Dreamweaver site, see “Setting up a Dreamweaver site” on page 59.

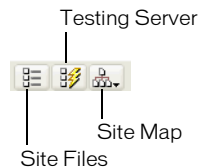
To change the current site view when the Site panel is collapsed (Windows only):

In the Site panel, with the panel collapsed, select Local View, Remote View, or Testing Server from the pop-up menu where the current view appears.

Note: Local View appears in the pop-up menu by default.

To change site views when the Site panel is expanded:

In the Site panel, with the panel expanded, click the Site Files button (for the remote site), Testing Server button, or Site Map button on the Site panel toolbar.



Note: If you use the Site Map button, you can choose to view the site map with site files or to view the site map only. For more information, see “Using the site map” on page 81.

Changing Site panel layout

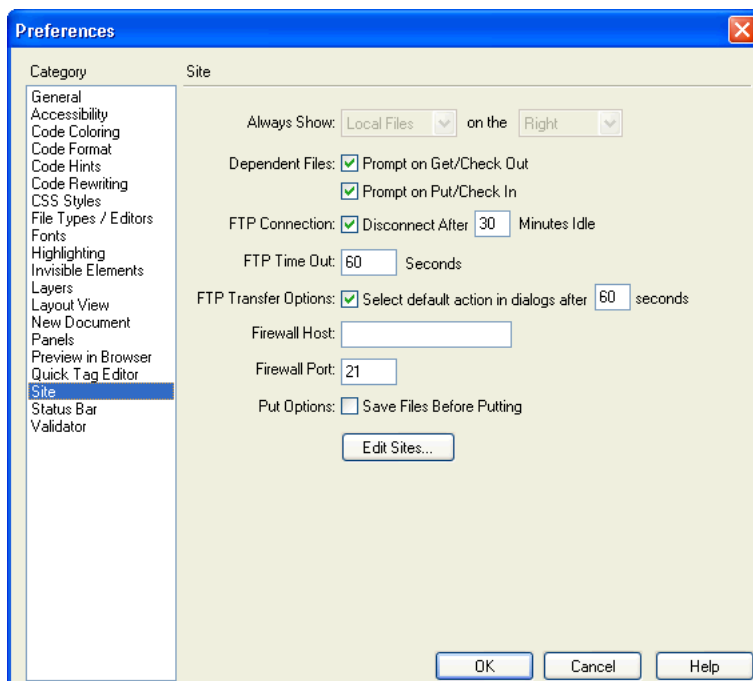
When the Site panel is in split view, the local site appears on the right side of the Site panel, by default. The remote site, testing server, or site map appears on the left side by default. You can switch these displays.

To change the Site panel layout:

- 1 (Windows only) In the Site panel, click the Expand/Collapse button to expand the panel, if it isn't already expanded.
- 2 Choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X).
The Preferences dialog box appears.

- 3 Select Site from the category list on the left.

The Site preference options appear.



- 4 Select either Local Files or Remote Files from the Always Show pop-up menu.
- 5 Select where you want your Always Show files to appear, either on the right or on the left in the expanded Site panel.
- 6 Click OK.

To change the size of the view area:

In the expanded Site panel, do one of the following:

- Drag the bar that separates the two views to increase or decrease the view area of the right or left pane.
- Use the scroll bars at the bottom of the Site panel to scroll through the views' contents.
- In the site map, drag the arrow above a file to change the space between files.

Related topic

“Changing file view columns” on page 74

Working with files in the Site panel

Use the Site panel to view sites as lists of files, to open files, to rename files, to add new folders or files to a site, or to refresh the view of a site after changes have been made.

You can also use the Site panel to determine which files (on either the local or remote site) have been updated since the last time they were transferred. For information on synchronizing the local site with the remote site, see “Synchronizing the files on your local and remote sites” on page 93.

To open a file from the Site panel, do one of the following:

- Double-click the file’s icon.
- Right-click (Windows) or Control-click (Macintosh) the file’s icon, and then choose Open.
- Choose File > Open (Windows) or Site > Open (Macintosh).

To add a new file or folder to a site:

- 1 Select a file or folder in the Site panel.

Dreamweaver will create the new file or folder inside the currently selected folder, or in the same folder as the currently selected file.

- 2 For a new folder, choose File > New Folder in the Site panel (Windows) or Site > Site Files View > New Folder (Macintosh).

For a new file, choose File > New File in the Site panel (Windows) or Site > Site Files View > New File (Macintosh).

Note: Alternatively, choose New File or New Folder from the Site panel’s context menu.

- 3 Enter a name for the new file or folder.
- 4 Press Enter (Windows) or Return (Macintosh).

To rename a file or folder in a site:

- 1 In the Site panel, select the file or folder you want to rename.
- 2 Do one of the following to activate the name of the file or folder:
 - Choose File > Rename (Windows) or Site > Rename (Macintosh).
 - Click in the filename, pause, then click again.
 - Right-click (Windows) or Control-click (Macintosh) the file’s icon, and choose Rename.
- 3 Type in the new name.
- 4 Press Enter (Windows) or Return (Macintosh).

To refresh the Site panel, do one of the following:

- Choose View > Refresh (Windows) or Site > Site Files View > Refresh (Macintosh).
- Click the Refresh button in the Site panel (this option refreshes both panes).

Note: Dreamweaver automatically refreshes the Site panel when you make changes in another application and then return to Dreamweaver.

To locate and select checked out files:

In the Site panel, choose Edit > Select Checked Out Files (Windows) or Site > Site Files View > Select Checked Out Files (Macintosh).

To locate and select files that are newer in the local site than in the remote site:

In the Site panel, choose Edit > Select Newer Local (Windows) or Site > Site Files View > Select Newer Local (Macintosh).

To locate and select files that are newer in the remote site than in the local site:

In the Site panel, choose Edit > Select Newer Remote (Windows) or Site > Site Files View > Select Newer Remote (Macintosh).

Finding files in the Site panel

You can search for a file in your local and remote sites from the Site panel. For more information on searching for and replacing text within files, see “Searching and replacing text” on page 294.

To find a file in your local site:

- 1 Open the file from the remote site or select the file in the Remote view of the Site panel.
- 2 Do one of the following:
 - If you opened the file in the Document window, choose Site > Locate in Site.
 - If you selected the file in the Site panel, right-click (Windows) or Control-click (Macintosh), and then select Locate in Local Site.

Dreamweaver highlights the file in the Local view of the Site panel.

To find a file in your remote site:

- 1 Open the file from the local site or select the file in the Local view of the Site panel.
- 2 Do one of the following:
 - If you opened the file in the Document window, choose Site > Locate in Site.
 - If you selected the file in the Site panel, right-click (Windows) or Control-click (Macintosh), and then select Locate in Remote Site.

Dreamweaver highlights the file in the Remote Site view of the Site panel.

Note: If you select Site > Locate in Site while the Document window is active, and if the current file is not part of the currently open site, Dreamweaver attempts to determine which of your locally defined sites the current file belongs to; if the current file belongs to only one local site, Dreamweaver opens that site and then locates the file in it.

Using the integrated file browser

The integrated file browser in the Site panel gives you access to your desktop and local network, including other defined sites and files not associated with a site. Use the integrated file browser to perform the following operations on files outside of the current site:

- Drag files
- Delete files
- Rename files
- Browse the network
- Open files in Dreamweaver or another applications
- Site operations, such as file transfer

Note: You perform these operations the same way you perform operations on files in a defined site (see “About the Site panel” on page 70). The best way to manage your files is to create a Dreamweaver site (see “Setting up a Dreamweaver site” on page 59).

When you drag a file from one site to another or to a folder that is not associated with a site, Dreamweaver *copies* the file you drag to the location where you drop it. If you drag a file within the same site, Dreamweaver *moves* the file you drag from its location to the location where you drop it.

When you drag a file that is not associated with a site, to a site, Dreamweaver copies the file to the location where you drop the file. If you drag a file that is not associated with a site to another folder that is not associated with a site, Dreamweaver moves the file to the location where you drop the files.

Note: To move a file that Dreamweaver copies by default, hold down the Shift key (Windows) or the Command key (Macintosh) while you drag. To copy a file that Dreamweaver moves by default, hold the Control key (Windows) or the Option key (Macintosh) while you drag.

In the file browser, site folders appear in a different color than non-site folders so that you can distinguish between the two. If you try to perform a site operation on a folder or file that is not associated with a site, Dreamweaver asks you to define a Dreamweaver site before performing the operation.

To use the integrated file browser:

In the Site panel file tree, click the plus sign (+) beside Desktop (Windows) or the expander arrow beside Computer (Macintosh).

About the site map

Use the site map to view a local folder for a Dreamweaver site as a visual map of linked icons, to add new files to a Dreamweaver site, or to add, modify, or remove links. The site map is ideal for laying out a site structure. You can quickly set up the entire structure of the site and then create a graphic image of the site map.

Note: The site map applies to local sites only. To create a map of a remote site, copy the contents of the remote site into a folder on your local disk, and then use the Edit Sites command to define the site as a local site. For more information, see “Setting up a local folder” on page 61.

Using the site map

You must define a “home page” for your site before you can display the site map. The site’s home page can be any page in your site; it does not have to be the main page for your site. In this case, the home page is simply the starting point of the map.

To define a home page for your site:

- 1 Choose Site > Edit Sites.

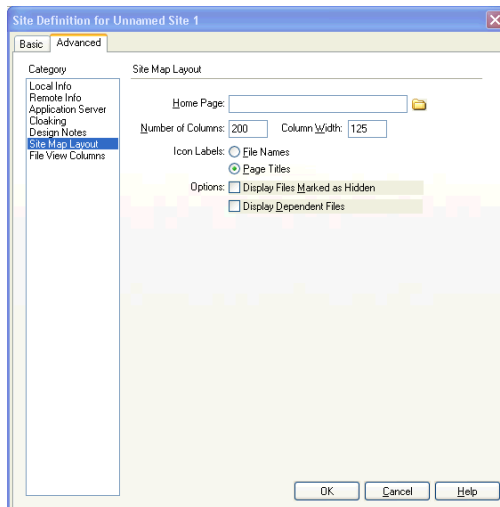
The Edit Sites dialog box appears.

- 2 Choose Edit to open an existing site.

The Site Definition dialog box appears.

- 3 Select Site Map Layout from the category list on the left.

The Site Definition dialog box displays the Site Map Layout options.



- 4 Click the folder icon to browse for a home page for the site, or type a file path in the Home Page text box.

- 5 Click OK.

To view a site map:

- 1 (Windows only) In the Site panel, click the Expand/Collapse button to expand the Site panel, if it isn't already expanded.
- 2 Select View > Site Map or click the Site Map button, and then select Map Only or Site and Map. Select Map Only, to view the site map without the local file structure, or Site and Map, to view the site map with the local file structure.

Note: If you have not defined a home page, or if Dreamweaver can't find an index.html or index.htm page in the current site to use as the home page, a dialog box appears prompting you to select a home page by clicking Edit Sites. Choose a site, click Edit, and then select Site Map Layout from the category list of the left of the Site Definition dialog box.

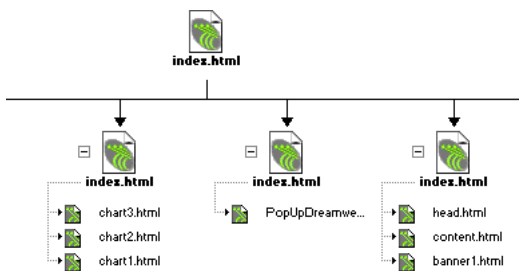
The Site panel displays a site map for the current site or a site map for the current site, along with the local file structure.

Viewing the site map

The site map shows HTML files and other pages as icons. The site map displays links in the order in which they are encountered in the HTML source code, as follows:

- Text displayed in red indicates a broken link.
- Text displayed in blue and marked with a globe icon indicates a file on another site or a special link (such as an e-mail or script link).
- A green check mark indicates a file checked out by you.
- A red check mark indicates a file checked out by someone else.
- A lock icon indicates a file that is read-only (Windows) or locked (Macintosh).

By default, the site map displays the site structure two levels deep, starting from the current home page. Click the plus (+) and minus (-) signs next to a page to show or hide pages linked below the second level.



By default, the site map does not display hidden files and dependent files. Hidden files are HTML files marked as hidden. Dependent files are non-HTML page content such as images, templates, Macromedia Shockwave files, or Macromedia Flash files. For more information, see “Showing and hiding site map files” on page 85.

Modifying the site map layout

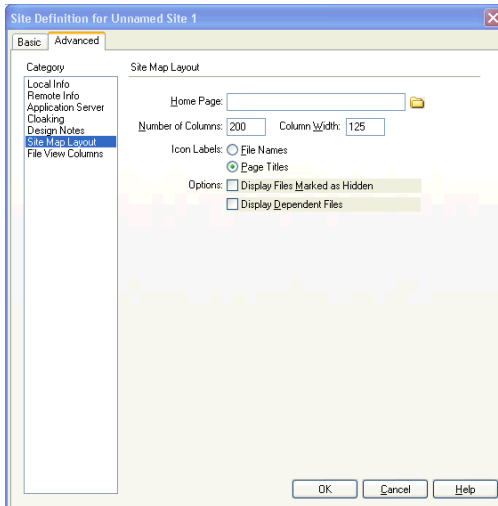
Use the Site Map Layout options to customize the appearance of your site map. You can specify the home page, the number of columns displayed, whether the icon labels display the filename or the page title, and whether to show hidden and dependent files.

To modify the site map layout:

- 1 Do one of the following to open the Site Definition dialog box:
 - Choose Site > Edit Sites, and then click Edit. Select Site Map Layout in the Category list on the left.

- In the Site panel, choose Site > Layout (Windows) or Site > Site Map View > Link to New File (Macintosh).

The Site Definition dialog box displays Site Map Layout options.



- 2 Make changes as necessary.

For more information, click the Help button in the dialog box.

- 3 Click OK to close the Site Definition dialog box.
- 4 Click Done to close the Edit Sites dialog box, if it is open.

Working with pages in the site map

When working in the site map, you can select pages, open a page for editing, add new pages to the site, create links between files, and change page titles.

To select multiple pages in the site map, do one of the following:

- Shift-click to select a range of adjacent pages.
- Starting from a blank part of the view, drag around a group of files to select them.
- Control-click (Windows) or Command-click (Macintosh) to select nonadjacent pages.

To open a page in the site map for editing, do one of the following:

- Double-click the file.
- Select the file, and then choose File > Open (Windows) or Site > Open (Macintosh).

To add an existing file to the site, do one of the following:

- Drag a file from the Windows Explorer or the Macintosh Finder to a file in the site map. The page is added to the site, and a link is created between it and the file you dragged it to.
- Select Site > Link to Existing File (Windows) or Site > Site Map View > Link to Existing File (Macintosh), and then browse to the file.

To create a new file and add a link in the site map:

- 1 Select an HTML file in the site map, then do one of the following:
 - Choose Site > Link to New File (Windows) or Site > Site Map View > Link to New File (Macintosh).
 - Choose Link to New File from the context menu.

The Link to New File dialog box appears.



- 2 Enter a name, title, and text for the link.
- 3 Click OK.

Dreamweaver saves the file in the same folder as the selected file. If you add a new file to a hidden branch, the new file is also hidden (see “Showing and hiding site map files” on page 85).

To modify the title of a page in the site map:

- 1 Make sure the Show Page Titles option is selected by choosing View > Show Page Titles (Windows) or Site > Site Map View > Show Page Titles (Macintosh) in the Site panel.
- 2 Select a page, and then do one of the following:
 - Click the title. When the title becomes editable, enter a new title.
 - Choose File > Rename (Windows) or Site > Rename (Macintosh).
- 3 Press Enter (Windows) or Return (Macintosh) after you type the new name.

Note: When you work in the Site panel, Dreamweaver automatically updates all links to files whose names have changed.

To change the home page in the site map, do one of the following:

- In the Local view of the Site panel, select a file, and then right-click (Windows) or Control-click (Macintosh), and choose Set as Home Page.
- In the Local view of the Site panel, click the file you want to make the new home page, and then choose Site > Set as Home Page (Windows) or Site > Site Map View > Set as Home Page (Macintosh).
- Select a file in the Site map, and choose Site > New Home Page (Windows) or Site > Site Map View > New Home Page (Macintosh).
- Choose Site > Edit Sites, and click Edit. Select Site Map Layout in the Site Definition dialog box category list. Browse to a new home page, and then click OK.

To update the site map display after making changes:

- 1 Click anywhere in the site map to deselect any files.
- 2 Choose View > Refresh (Windows) or Site > Site Map View > Refresh Local (Macintosh).

Showing and hiding site map files

You can modify the layout of the site map to show or hide hidden and dependent files. This is useful when you want to emphasize key topics or content and de-emphasize less important material.

To hide a file using the site map, you must mark the file as hidden. When you hide a file, its links are also hidden. When you display a file marked as hidden, the icon and its links are visible in the site map, but the names appear in italics.

Note: By default, dependent files are already hidden.

To mark files as hidden in the site map:

- 1 In the site map, select one or more files.
- 2 Choose View > Show/Hide Link (Windows) or Site > Site Map View > Show/Hide Link (Macintosh).

To show or hide files marked as hidden in the site map, do one of the following:

- Choose View > Show Files Marked as Hidden (Windows) or Site > Site Map View > Show Files Marked as Hidden (Macintosh).
- Choose View > Layout (Windows) or Site > Site Map View > Layout (Macintosh) to open the Site Definition dialog box, and then select the Display Files Marked as Hidden option.

To show dependent files in the site map, do one of the following:

- Choose View > Show Dependent Files (Windows) or Site > Site Map View > Show Dependent Files (Macintosh).
- Choose View > Layout (Windows) or Site > Site Map View > Layout (Macintosh) to open the Site Definition dialog box, and select the Display Dependent Files option.

To unmark files marked as hidden in the site map:

- 1 In the site map, select one or more files.
- 2 Choose View > Show Files Marked as Hidden (Windows) or Site > Site Map View > Show Files Marked as Hidden (Macintosh).
- 3 Choose View > Show/Hide Link (Windows) or Site > Site Map View > Show/Hide Link (Macintosh).

Viewing the site from a branch

You can view the details of a specific section of a site by making a branch the focus of the site map.

To view a different branch in the site map:

Select the page you want to view, and then choose View > View as Root (Windows) or Site > Site Map View > View as Root (Macintosh).

The site map is redrawn in the window as if the specified page were at the root of the site. The Site Navigation text box above the site map displays the path from the home page to the specified page. Select any item in the path to view the site map from that level by clicking once.

To expand and contract branches in the site map:

Click a branch's plus (+) or minus (–) sign.

Saving the site map

You can save the site map as an image, and then view the image in (or print it from) an image editor.

To create an image file of the current site map:

- 1 From the site map, do one of the following:
 - In Windows, choose File > Save Site Map. In the Save Site Map dialog box, enter a name in the File Name text box. In the File Type pop-up menu, select .bmp or .png.
 - In Macintosh, select Site > Site Map View > Save Site Map > Save Site Map As PICT or Site > Site Map View > Save Site Map > Save Site Map As JPEG.
- 2 Choose a location to save the file, and enter a name for the image.
- 3 Click Save.

Importing and exporting sites

With Dreamweaver you can export a site as an XML file, and then import it back into Dreamweaver. This enables you to move sites between machines and product versions or to share with other users.

To export a site:

- 1 Choose Site > Edit sites.
The Edit Sites dialog box appears.
- 2 Click the Export button.
The Export Site panel appears.
- 3 Browse to a location to save the site.
- 4 Click Save.
Dreamweaver saves the site as an XML file, with an STE file extension, in the specified location.
- 5 Click Done to close the Edit Sites dialog box.

To import a site:

- 1 Choose Site > Edit sites.
The Edit Sites dialog box appears.
- 2 Click the Import button.
The Import Site panel appears.
- 3 Browse to and select a site (saved as an XML file) to import.
Note: You must export the site from Dreamweaver, which saves the site as an XML file, before you can import the file.
- 4 Click open.
Dreamweaver imports the site, and the site name appears in the Edit Sites dialog box.
- 5 Click Done to close the Edit Sites dialog box.

Removing a site from your list of sites

If you no longer want to work with a site in Dreamweaver, you can remove the site from your list of sites. The files in the site are not removed.

Note: When you remove a site from the list, all setup information about the site is permanently lost.

To remove a site from your site list:

1 Choose Site > Edit Sites.

The Edit Sites dialog box appears.

2 Select a site name.

3 Click Remove.

A dialog box appears asking you to confirm the removal.

4 Click Yes to remove the site from your list, or click No to leave the site name.

5 Click Done to close the Edit Sites dialog box.

Using Check In/Check Out

If you're working in a collaborative environment, you can check files in and out from local and remote servers. If you're the only person working on the remote server, you can use the Put and Get commands without checking files in or out (see "Getting and putting files" on page 90).

Checking out a file is the equivalent of declaring "I'm working on this file now—don't touch it!" When a file is checked out, Dreamweaver displays the name of the person who checked out the file in the Site panel, along with a red check mark (if a team member checked out the file) or green check mark (if you checked out the file) next to the file's icon.

Checking in a file makes the file available to other team members to check out and edit. When you check in a file after editing it, your local version becomes read-only and a lock symbol appears beside the file in the Site panel to prevent you from making changes to the file.

Dreamweaver does not make checked-out files read-only on the remote server. If you transfer files with an application other than Dreamweaver, you can overwrite checked-out files. However, in applications other than Dreamweaver, the .LCK file is visible next to the checked-out file in the file hierarchy to help prevent such accidents.

To enable or disable Check In/Check Out for some sites, see "Setting up the Check In/Check Out system" on page 88. For information about transferring files between local and remote sites without checking them in or out, see "Getting files from a remote or testing server" on page 90 and Putting files on a remote or testing server.

Note: You can use Get and Put functionality with a testing server, but you cannot use the Check In/Check Out system with a testing server.

Setting up the Check In/Check Out system

Before you can use the Check In/Check Out system, you must associate your local site with a remote server (see “Setting up a remote folder” on page 63).

To set up the Check In/Check Out system:

- 1 Choose Site > Edit Sites.

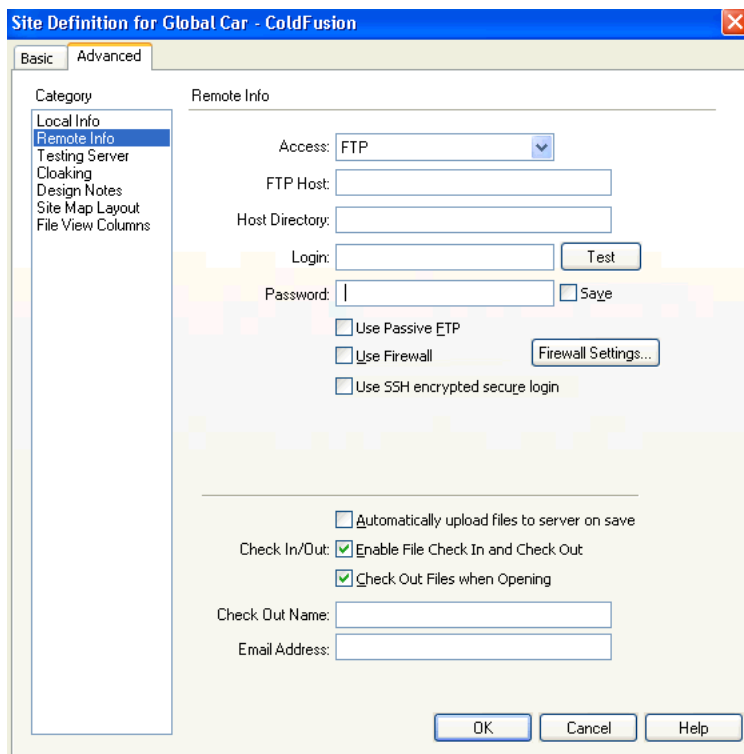
The Edit Sites dialog box appears.

- 2 Select a site, and then click Edit.

The Site Definition dialog box appears.

- 3 In the Category list at the left, click Remote Info.

The Site Definition dialog box displays Remote Info options. The Check In/Out section appears at the bottom of the dialog box.



Note: If you do not see Check In/Out options, you have not set up your remote server (see “Setting up a remote folder” on page 63).

- 4 Complete the Check In/Out section.

For more information, click the Help button in the dialog box.

- 5 Click OK.

Checking files in to and out of a remote site

Use the Site panel or the Document window to check in and check out files on a remote server.

The Dreamweaver Check In/Check Out system uses the following symbols in the Site panel:

- Green check mark indicates that you have the file checked out.
- Red check mark indicates that another team member has the file checked.
- Lock symbol indicates that the file is read-only (Windows) or locked (Macintosh).

If you check out a file and then decide not to edit it (or decide to discard the changes you made), you can undo a check out.

To check out files from a remote server from the Site panel:

- 1 Choose a site from the Site pop-up menu.
- 2 Select files to check out.

Note: You can select files in the Local or Remote view.

- 3 Do one of the following to check out the file(s):

- Click the Check Out button on the Site panel toolbar.
- Right-click (Windows) or Control-click (Macintosh), and then choose Check Out from the context menu.

- 4 Click Yes at the prompt to download dependent files along with the selected files or click No to refrain from downloading dependent files.

Note: It's usually a good idea to download dependent files when checking out a new file, but if the latest versions of the dependent files are already on the local disk, there's no need to download them again.

A green check mark appears beside the file's icon indicating that you have checked it out.

To check in files to a remote server from the Site panel:

- 1 Choose a site from the Site pop-up menu.
- 2 Select checked-out or new file(s).

Note: You can select files in the Local or Remote view.

- 3 Do one of the following to check in the file(s):

- Click the Check In button on the Site panel toolbar.
- Right-click (Windows) or Control-click (Macintosh), and then choose Check In from the context menu.

- 4 Click Yes at the prompt to download dependent files along with the selected files or click No to refrain from downloading dependent files.

Note: It's usually a good idea to upload dependent files when checking in a new file, but if the latest versions of the dependent files are already on the remote server, there's no need to upload them again.

A lock symbol appears beside the file's icon indicating that it's now read-only.

To check in or check out an open file from the Document window, do one of the following:

- Choose Site > Check In or Site > Check Out.
- Click the File Management icon on the Document window's toolbar, and then choose Check In or Check Out from the pop-up menu.

Note: If you select Site > Check In or Site > Check Out, and the current file is not part of the currently open site, Dreamweaver attempts to determine which locally defined site the current file belongs to. If the current file belongs to only one local site, Dreamweaver opens that site, and then performs the check in or check out operation.

If you check out the currently active file, the currently open version of the file is overwritten by the new checked-out version. If you check in the currently active file, the file may be automatically saved before it's checked in, depending on the preference options you've set (see "Setting Site preferences" on page 73).

To undo a file check-out:

- 1 Select the file.
- 2 Do one of the following to undo the check-out:
 - Choose Site > Undo Check Out.
 - Right-click (Windows) or Control-click (Macintosh) the file in the Site panel, and then choose Undo Check Out from the context menu.

The local copy of the file becomes read-only, and any changes you've made to it are lost.

Getting and putting files

If you're working in a collaborative environment, use the Check In/Check Out system to transfer files between local and remote sites (see "Using Check In/Check Out" on page 87). If you're the only person working on the remote site, however, you can use the Get and Put commands to transfer files without checking them in or out.

Note: If you select Site > Get or Site > Put while the Document window is active, and if the current file is not part of the currently open site, Dreamweaver attempts to determine which of your locally defined sites the current file belongs to; if the current file belongs to only one local site, Dreamweaver opens that site, and then performs the Get or Put operation.

Getting files from a remote or testing server

The Get command copies files from the remote site or testing server to your local site, as follows:

- If you're using the Check In/Check Out system (that is, if the Enable File Check In and Check Out option is selected), getting a file results in a *read-only* local copy of the file; the file remains available on the remote site or testing server for other team members to check out (see "Using Check In/Check Out" on page 87).
- If you're not using the Check In/Check Out system (that is, if the Enable File Check In and Check Out option is not selected), getting a file transfers a copy that has both *read and write* privileges.

Note: If you're working in a collaborative environment—that is, if others are working on the same files—you should not turn off Enable File Check In and Check Out. If other people are using the Check In/Check Out system with the site, you should use that system as well.

You can get files by selecting files in either the Local or Remote view of the Site panel. If the Remote view is active, then Dreamweaver copies the selected remote files to the local site; if the Local view is active, then Dreamweaver copies the remote versions of the selected local files to the local site.

To get only those files for which the remote version is more recent than the local version, use the Synchronize command (see “Synchronizing the files on your local and remote sites” on page 93).

Dreamweaver records all FTP file transfer activity. If an error occurs when you are transferring a file using FTP, the Site FTP log can help you determine the problem.

To display the FTP log:

Select View > Site FTP Log.

To get files from a remote or testing server:

1 In the Site panel, choose a site from the Site pop-up menu.

2 If you're using FTP to transfer files, click the Connect button to open a connection to the remote server.

If a connection is already open (indicated by the Disconnect button), skip this step.

If the remote files are visible in the Remote pane from a previous connection, clicking Connect isn't required; when you click Get, Dreamweaver connects automatically.

3 Select the desired files to download.

Usually you select these files in the Remote or Testing Server view, but you can select the corresponding files in the Local view if you prefer.

4 Do one of the following to get the file:

- Click the Get button on the Site panel toolbar.
- Choose Get from the context menu.
- Choose Site > Get.

Note: If the file is currently open in a Document window, you can also choose Site > Get from the Document window.

A dialog box appears, asking if you want to get dependent files.

5 To download dependent files, click Yes; to skip them, click No.

If you already have local copies of the dependent files, click No.

To stop the file transfer at any time, click the Cancel button in the status dialog box. The transfer may not stop immediately.

Putting files on a remote or testing server

The Put command copies files from the local site to the remote site, generally without changing the file's checked out status. There are two common situations in which you might use Put instead of Check In:

- Use Put when you're not in a collaborative environment and you aren't using the Check In/Check Out system.

Note: If you want to put a file on a remote server and check it in, use the Check In command (see “Checking files in to and out of a remote site” on page 89).

- Use Put when you want to put the current version of the file on the server but you're going to keep editing it.

Note: If you put a file that didn't previously exist on the remote site and you're using the Check In/Check Out system, the file is copied to the remote site and is then checked out to you so that you can continue editing.

To put only those files for which the local version is more recent than the remote version, see *Synchronizing the files on your local and remote sites*.

Note: Do not use special characters (such as é, ç, or ¥) or punctuation, such as colons, slashes, or periods, in the names of files you intend to put on a remote or testing server. Many servers change these characters during the upload process, which causes any links to the files to break.

To put files on a remote or testing server:

- 1 In the Site panel, choose a site from the Site pop-up menu.
- 2 If you're using FTP to transfer files, you may click Connect to open a connection to the remote server, which enables you to see what's on the remote site before transferring any files.

Clicking Connect, however, is not required; when you click Put, Dreamweaver connects automatically.

- 3 Select files to upload.

Usually you select these in the Local view, but you may select the corresponding files in the Remote view if you prefer.

- 4 Do one of the following to put the file:

- Click the Put button on the Site panel toolbar.
- Choose Put from the context menu.
- Choose Site > Put.

If the file is currently open in a Document window, you can instead choose Site > Put from the Document window.

- 5 If the file hasn't been saved, a dialog box may appear (depending on your preference setting in the Site panel of the Preferences dialog box) allowing you to save the file before putting it on the remote server.

To save the file, click Yes; to put the previously saved version on the remote server, click No. If you choose not to save the file, any changes you've made since last time you saved will not be put onto the remote server. However, the file remains open, so you can still save the changes after putting the file on the server if you want.

A dialog box appears, asking if you want to put dependent files.

- 6 To upload dependent files, click Yes; to skip them, click No.

If the remote site already contains copies of the dependent files, click No. To avoid being asked about dependent files in future uploads, select the Don't Ask Me Again option.

To stop the file transfer click the Cancel button in the status dialog box. The transfer may not stop immediately.

Dreamweaver records all FTP file transfer activity. If an error occurs when you are transferring a file using FTP, the Site FTP log can help you determine the problem. To display the log, select View > Site FTP Log.

Synchronizing the files on your local and remote sites

Once you've created files in your local and remote sites, you can synchronize the files between the two sites. Use the Site > Synchronize command to transfer the latest versions of your files to and from your remote site.

If your remote site is an FTP server (rather than a networked server), then synchronizing your files uses FTP. For more information on using FTP, see “Setting up a remote folder” on page 63.

Before you synchronize your sites, Dreamweaver lets you verify which files you want to put onto or get from your remote server. Dreamweaver also confirms which files have been updated after you complete the synchronization.

To see which files are newer on the local site or the remote site, without synchronizing, do one of the following:

- Choose Edit > Select Newer Local or Edit > Select Newer Remote (Windows, from the Site panel).
- Choose Site > Site Files View > Select Newer Local or Site > Site Files View > Select Newer Remote (Macintosh).

To synchronize your files:

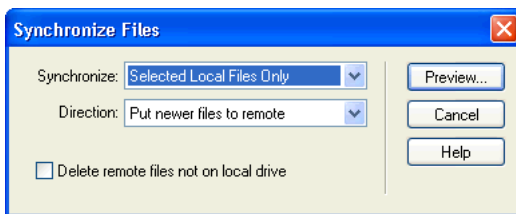
- 1 In the Site panel, choose a site from the Current Sites list pop-up menu.
- 2 Select specific files or folders.

If you want to synchronize the entire site, skip this step.

- 3 In the Site panel, do one of the following to synchronize:

- Choose Site > Synchronize.
- Right-click (Windows) or Control-click (Macintosh), and then select Synchronize from the context menu.

The Synchronize Files dialog box appears.



- 4 Complete the dialog box.

For more information, click the Help button in the dialog box.

Dreamweaver automatically synchronizes files. If the files are already in sync, Dreamweaver lets you know that no synchronization is necessary.

Cloaking folders and files in your site

Site cloaking enables you to exclude folders and file types in a site from operations such as Get or Put. You can cloak individual folders, but not individual files. To cloak files, you must select a file type, and Dreamweaver cloaks all files of that type. Dreamweaver remembers your settings for each site, so you don't have to make selections each time you work on that site.

For example, if you're working on a large site and don't want to upload your multimedia files each day, you can use site cloaking to cloak your multimedia folder, and the system excludes files in that folder from site operations you perform.

You can cloak folders and file types on the remote or local site. Cloaking excludes cloaked folders and files from the following operations:

- Put, Get, Check In, and Check Out
- Reports
- Select newer local and newer remote
- Sitewide operations, such as check and change links and search/replace
- Synchronize
- Asset panel contents
- Template and library updating

Note: Dreamweaver excludes cloaked templates and library items from Get and Put operations only. Dreamweaver does not exclude these items from batch operations, because it might cause them to get out of sync with their instances.

You can perform an operation on a specific cloaked folder by selecting the item in the Site panel and performing an operation on it. Performing an operation directly on a file or folder overrides cloaking.

Enabling and disabling site cloaking

Site cloaking is enabled by default. You can disable cloaking permanently or just temporarily to perform an operation on all files, including cloaked files. When you disable site cloaking, all cloaked files are uncloaked. When you enable site cloaking again, any previously cloaked files become cloaked again.

Note: You can also use the Uncloak All option to uncloak all files, but this does not disable cloaking; also there is no way to re-cloak all folders and files that were previously cloaked, except to set cloaking again for each folder and file type.

To enable or disable site cloaking:

- 1 In the Site panel, select a site from the Site pop-up menu.
- 2 In the Site panel, do one of the following:
 - Choose Site > Cloaking to display the submenu.
 - Select a folder or file, right-click (Windows) or Control-click (Macintosh) to display the context menu.
- 3 From the submenu, do one of the following:
 - Select Enable/Disable Cloaking.
 - Select Settings. Then, in the Advanced Site Definition dialog box, choose Cloaking from the category list on the left, select or deselect Enable Cloaking, and click OK.

Cloaking is enabled or disabled for the site.

Cloaking and uncloaking site folders

You can cloak specific folders, but you cannot cloak all folders or an entire site. When you cloak specific folders, you can cloak multiple folders at the same time.

To cloak or uncloak specific folders within a site:

- 1 In the Site panel, select a site from the Site pop-up menu that has site cloaking enabled.
- 2 Select the folder(s) you want to cloak or uncloak.
- 3 In the Site panel, do one of the following:
 - Choose Site > Cloaking > Cloak or Site > Cloaking > Uncloak.
 - Right-click (Windows) or control-click (Macintosh), and then choose Cloaking > Cloak or Cloaking > Uncloak from the context menu.

A red line through the folder icon appears or disappears, indicating that the folder is cloaked or uncloaked.

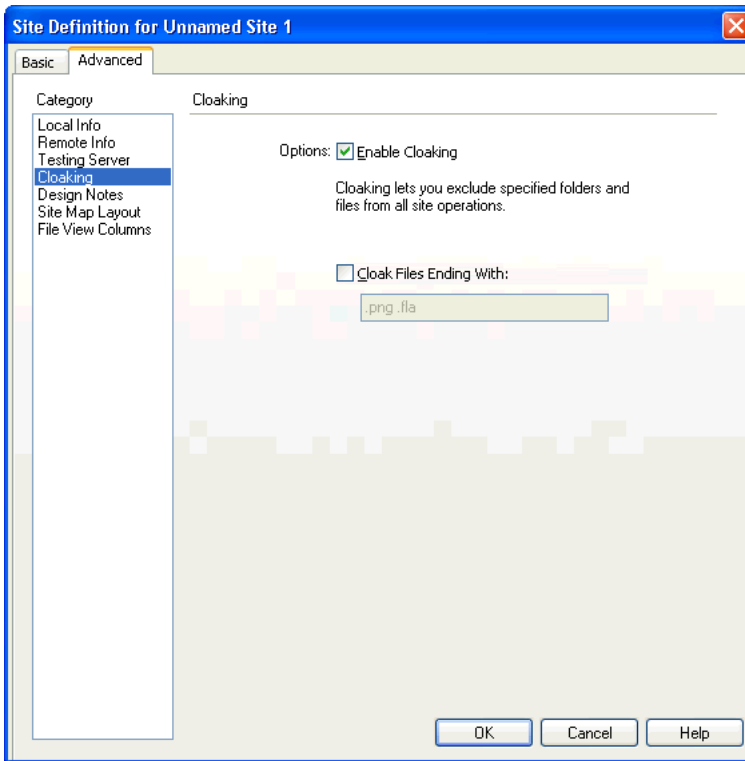
Cloaking and uncloaking specific file types

You can indicate specific file types to cloak, so that Dreamweaver cloaks all files ending with a specified pattern. For example, you can cloak all files ending with the .txt extension. The file types that you enter do not have to be file extensions; they can be any pattern at the end of a filename.

To cloak specific file types within a site:

- 1 In the Site panel, select a site from the Site pop-up menu that has site cloaking enabled.
- 2 Do one of following:
 - Choose Site > Cloaking > Settings.

- Right-click (Windows) or Control-click (Macintosh), and then select Settings. The Advanced Site Definition dialog box appears.



3 Select the Cloak Files Ending With checkbox.

4 Enter the file types to cloak in the text box.

For example, you might enter **.jpg** to cloak all files with names ending in **.jpg** in your site.

Note: Separate multiple file types with one space; do not use a comma or semicolon.

5 Click OK.

A red line appears through the affected files, indicating that they are cloaked.

Tip: Some software creates backup files ending in a particular suffix, such as **.bak**. You can cloak such files.

To uncloak specific file types within a site:

1 In the Site panel, select a site from the Site pop-up menu that has site cloaking enabled.

2 Do one of following:

- Choose Site > Cloaking > Settings.
- Right-click (Windows) or Control-click (Macintosh), and then select Cloaking > Settings.

The Advanced Site Definition dialog box appears.

3 Do one of the following:

- Deselect the Cloak Files Ending With checkbox to uncloak all the file types listed in the text box.
- Delete specific file types from the text box to uncloak those file types.

4 Click OK.

The red lines disappear from the affected files, indicating that they are uncloakd.

Uncloaking all folders and files

You can uncloak all folders and files in a site at the same time. This action cannot be undone; there is no way to re-cloak all items that were previously cloaked. You have to re-cloak items individually.

Note: If you want to temporarily uncloak all folders and files, and then re-cloak those items, disable site cloaking (see “Enabling and disabling site cloaking” on page 94).

To uncloak all folders and files within a site:

1 In the Site panel, select a site from the Site pop-up menu that has site cloaking enabled.

2 Select any file or folder in that site.

3 In the Site panel, do one of the following:

- Choose Site > Cloaking > Uncloak All.
- Right-click (Windows) or control-click (Macintosh), and then choose Cloaking > Uncloak All.

Note: This step also deselects the Cloak Files Ending With checkbox in the Site > Cloaking > Settings.

The red lines through folder and file icons disappear, indicating that all files and folders in the site are uncloakd.

Using Design Notes

Design Notes are notes that you create for a file. The Design Notes are associated with the file they describe, but stored in a separate file. You can use Design Notes to keep track of extra file information associated with your documents, such as image source-filenames and comments on file status.

For example, if you copy a document from one site to another, you can add Design Notes for that document, with the comment that the original document is in the other site folder. Then if you update that document (or someone else does), you’ll know that you need to update the original page as well.

You can also use Design Notes to keep track of sensitive information that you can’t put inside a document for security reasons. For example, you can include information about how the document was created, or notes on how a particular price or configuration was chosen, or what marketing factors influenced a design decision.

You can see which files have Design Notes attached in the Site panel: a Design Notes icon appears in the Notes column.

For general information on using Design Notes, see “Saving file information in Design Notes” on page 99. For information on using Design Notes with Fireworks, see “Using Design Notes in Fireworks and Flash with Dreamweaver” on page 101.

Enabling and disabling Design Notes

You enable and disable Design Notes for a site in the Design Notes category of the Site Definition dialog box. When you enable Design Notes, you can choose to use them locally only, if you want.

To enable or disable Design Notes for your site or to use Design Notes locally:

- 1 Choose Site > Edit Sites.

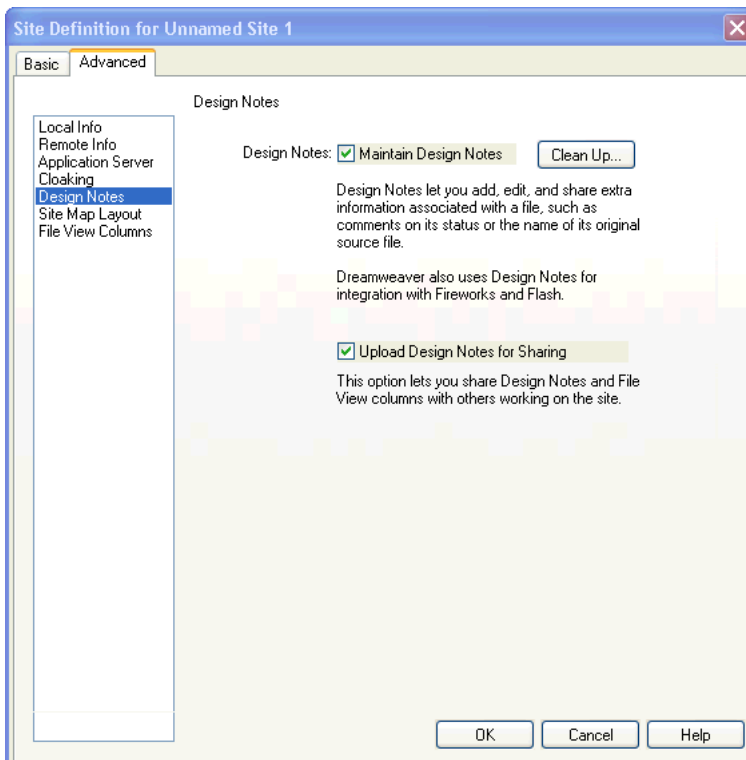
The Edit Sites dialog box appears.

- 2 Select a site, and then click Edit.

The Site Definition dialog box appears.

- 3 Click Design Notes in the Category list on the left.

The Site Definition dialog box appears with the Design Notes options.



- 4 Complete the dialog box.

For more information, click the Help button in the dialog box.

- 5 Click OK.

Saving file information in Design Notes

You can create a Design Notes file for each document or template in your site. You can also create Design Notes for applets, ActiveX controls, images, Flash movies, and Shockwave objects, and image fields in your documents.

Note: If you add Design Notes to a template file, documents you create with the template do not inherit the Design Notes.

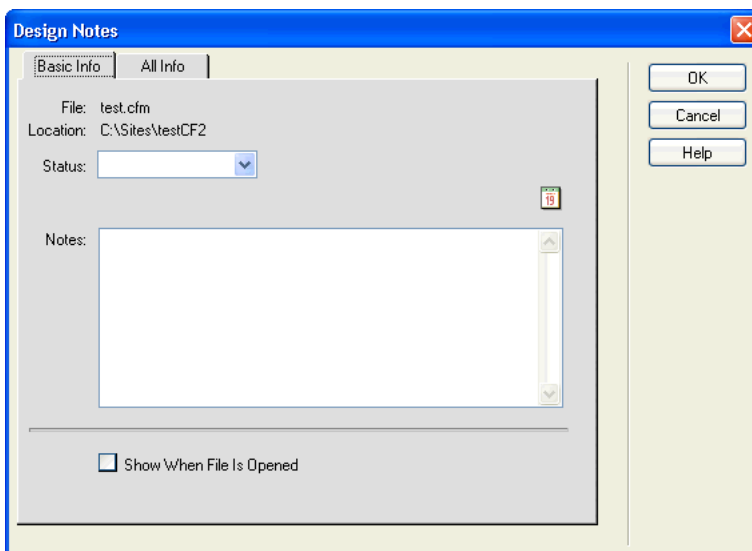
To add Design Notes to a document:

- 1 While the document is active in the Document window, choose File > Design Notes.

You can also select the file in the Site panel, and then choose File > Design Notes.

Note: If the file resides on a remote site, you must first check out or get the file, then select it in the local folder (see “Checking files in to and out of a remote site” on page 89 or “Getting and putting files” on page 90).

The Design Notes dialog box appears.



- 2 Complete the dialog box.

For more information, click the Help button in the dialog box.

- 3 Click OK to save the notes.

Dreamweaver saves your notes to a folder called `_notes`, in the same location as the current file. The filename is the document's filename, plus the extension `.mno`.

For example, if the filename is `index.html`, the associated Design Notes file is named `index.html.mno`.

To add Design Notes to an object:

- 1 Select the object.
- 2 Right-click (Windows) or Control-click (Macintosh), and then choose Design Notes from the object's context menu.

The Design Notes dialog box appears.

- 3 Complete the dialog box.

For more information, click the Help button in the dialog box.

- 4 Click OK to save the notes.

Dreamweaver saves an object's Design Notes file in a `_notes` folder in the same directory as the object's source file, which is not necessarily in the same directory as the document that the object appears in.

To open Design Notes associated with a file, do one of the following:

- Select the file in the Site panel, or open the file, and then choose File > Design Notes.
- In the Notes column of the Site panel, double-click the yellow Design Notes icon.

To assign a custom Design Notes status:

- 1 Open Design Notes for a file or object (see the previous procedure).
- 2 Click the All Info tab.
- 3 Click the plus (+) button.
- 4 In the Name field, enter the word **status**.
- 5 In the Value field, enter the status.

If a status value already existed, it's replaced with the new one.

- 6 Click the Basic Info tab and note that the new status value appears in the Status pop-up menu.

Note: You can have only one custom value in the status menu at a time. If you follow this procedure again, Dreamweaver replaces the status value you entered the first time with the new status value you enter.

To delete unassociated Design Notes from your site:

- 1 Choose Site > Edit Sites.

The Edit Sites dialog box appears.

- 2 Select the site, and then click Edit.

The Site Definition dialog box appears.

- 3 Click Design Notes in the category list on the left.

- 4 Click the Clean Up button.

Dreamweaver prompts you to verify that it should delete any Design Notes that are no longer associated with a file in your site.

If you use Dreamweaver to delete a file that has an associated Design Notes file, Dreamweaver deletes the Design Notes file too; so usually orphan Design Notes files occur only if you delete or rename a file outside of Dreamweaver.

Note: If you deselect the Maintain Design Notes option before you click Clean Up, Dreamweaver deletes all Design Notes files for your site.

Using Design Notes in Fireworks and Flash with Dreamweaver

If you open a file in Macromedia Fireworks or Macromedia Flash and export it to another format, Fireworks and Flash automatically save the name of the original source file in a Design Notes file.

For example, if you open `myhouse.png` in Fireworks and export it to `myhouse.gif`, Fireworks creates a Design Notes file called `myhouse.gif.mno`. This Design Notes file contains the name of the original file, as an absolute `file: URL`. So, the Design Notes for `myhouse.gif` might contain this line:

```
fw_source="file:///Mydisk/sites/assets/orig/myhouse.png"
```

A similar Flash Design Note might contain this line:

```
fl_source="file:///Mydisk/sites/assets/orig/myhouse fla"
```

Note: To share Design Notes, users should define the same site-root path. For example, `sites/assets/orig`.

When you import the graphic into Dreamweaver, the Design Notes file is automatically copied into your site along with the graphic. When you select the image in Dreamweaver and choose to edit it using Fireworks (see “Launching an external editor for media files” on page 320), Fireworks opens the original file for editing.

Using reports to improve workflow

You can use reports to improve collaboration among members of a web team.

You can run workflow reports that display who has checked out a file and which files have Design Notes associated with them. You can further refine Design Note reports by specifying `name/value` parameters.

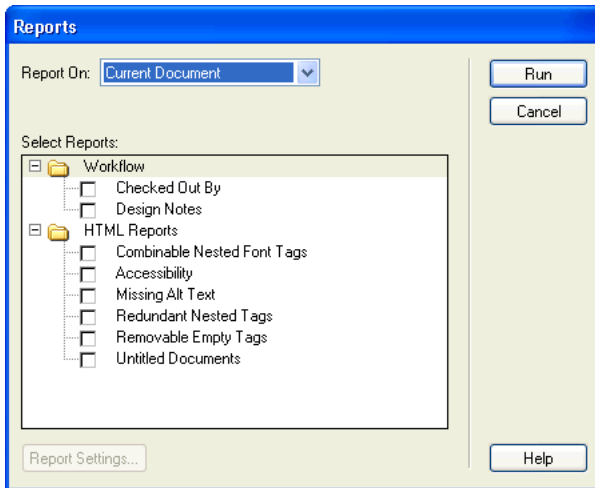
For information on running other types of reports, see “Using Reports to test a site” on page 468.

Note: You must have a remote site connection defined to run the Workflow reports.

To run a Checked Out By report:

- 1 Open a document.
- 2 Choose Site > Reports.

The Reports dialog box appears.



- 3 Select a report category to report on and report types you want to run.

For more information, click the Help button in the dialog box.

- 4 Click Run.

Depending on what you chose to report on, you may be prompted to save your file, define your site, or select a folder (if you haven't already done so).

The report appears in the Site Reports panel (in the Results panel group).

To use a report:

- 1 In the Site Reports panel, click the column heading you want to sort by to sort results.

You can sort by filename, line number, or description. You can also run several different reports and keep the different Results panel open.

- 2 Select any line in the report, and click the More Info button on the left side of the Site Reports panel for an description of the problem.

The information appears in the Reference panel (in the Code panel group).

- 3 Double click any line in the report to view the corresponding code in the Document window.

Note: If you are in Design view, Dreamweaver changes the display to split view to show the reported problem in code.

- 4 Click Save Report to save the report.

When you save a report, you can import it into an existing template file. You can then import the file into a database or spreadsheet and print it, or use the file to display the report on a website.

About the Sitespring panel in Dreamweaver

You can use the Sitespring panel to view all your Sitespring directly in Dreamweaver. You can also select a task from the panel and connect to your Sitespring web page and edit the task. You can save any changes you make in Sitespring, return to Dreamweaver and refresh the Sitespring panel to see the changes. The Sitespring panel displays your tasks, along with their priority, status, due date, and project name.

System requirements

To use the Sitespring panel with Dreamweaver, you must have Dreamweaver MX and Sitespring installed on your system or server. If you don't have Sitespring, you can download a free trial version from the Macromedia Sitespring website. Sitespring is only compatible with Flash-enabled, newer browsers, such as Netscape Communicator 4.06 or later and Microsoft Internet Explorer 5 or later.

Connecting to your Sitespring server

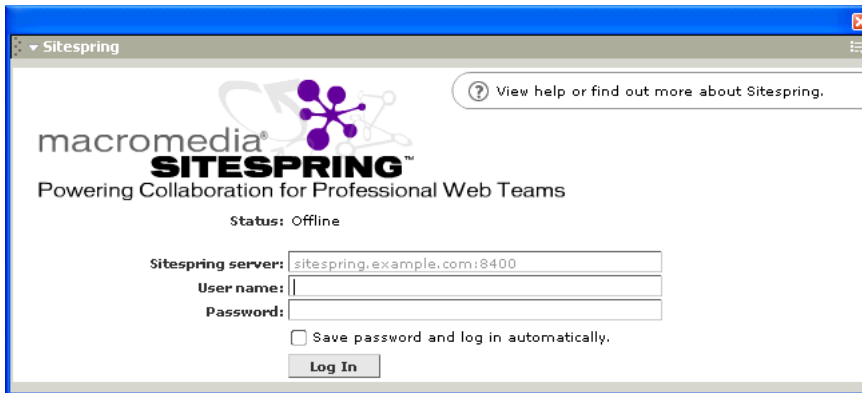
To use the Sitespring panel, you must first connect and log in to your Sitespring server.

Note: For more information on using Macromedia Sitespring, please refer to the Sitespring documentation.

To connect to your Sitespring server from Dreamweaver:

- 1 In the Document window, choose **Window > Others > Sitespring** or press F7.

The Sitespring Login appears.



- 2 In the Sitespring Server text box, enter the URL for the Sitespring server to which you will be connecting. If you are not sure what this URL is, contact your system administrator.
- 3 In the User Name and Password text boxes, enter your Sitespring server user name and password. The User Name text box is case-sensitive.
- 4 If you would like to save your password on your computer, select the Save Password and Login Automatically option.

Note: This option only lets you close and reopen the Sitespring panel without login information. If you log out of Sitespring completely, either through Dreamweaver or through the application itself, you will need to reenter this information.

- 5 Click Log In to connect to the Sitespring server.

The status of the connection appears above the Sitespring Server text box. The Sitespring panel displays your current tasks when the connection is made.

If you have difficulty connecting to Sitespring, contact your system administrator.

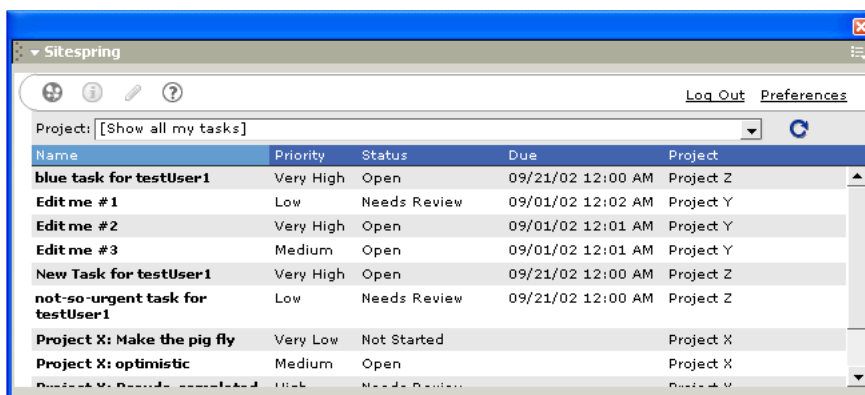
Using the Sitespring panel

You can use the Sitespring panel to view all your tasks or only those associated with a specific project. You can edit a task and refresh the task list to see the changes. The Tasks text box displays tasks, priority, status, due date, and project name.

To use the Sitespring panel:

- 1 In the Document window, choose Window > Others > Sitespring or press F7.
- 2 Log in to the Sitespring server, if you haven't already done so (see "Connecting to your Sitespring server" on page 103).

The Sitespring panel displays your current tasks. It consists of a Tasks text box displaying your current open tasks, and a menu and icons for performing certain actions. Tasks are sorted by task name. Click a column heading to sort by that column. The Sitespring panel does not display tasks that have a status of Suspended or Completed.



- 3 Select from the following options:

- Choose a specific project from the Project pop-up menu to display only those tasks associated with a specific project, or choose Show All Tasks to display all your current tasks.
- Click the Refresh icon to refresh the Sitespring panel and see any changes you have made to this task.

Note: You must save changes in Sitespring first before they appear in the panel.

- Select a task in the Name column, and then click the Edit Task button to edit that task.

To edit tasks, you must also log in to Sitespring via the Please Log In web page. If you are logged in, the Edit Task page associated with this task appears. If you haven't yet logged in, the Please Log In web page appears.

- Click the Sitespring icon to go directly to a Sitespring web page.
If you have already logged in, your Sitespring home page appears. If you haven't yet logged in, the Please Log In page appears.
- Press F7 to close the Sitespring panel, or click the panel's Close box to close the Sitespring panel.

Note: You may want to close the Sitespring panel rather than log out completely, so you don't have to log in each time.

- Click the Log Out link to log out of Sitespring or to switch users.

The Sitespring Login appears. You can enter a new user name and password to switch users, or close the dialog box to disconnect (see “Logging out of Sitespring in Dreamweaver” on page 105).

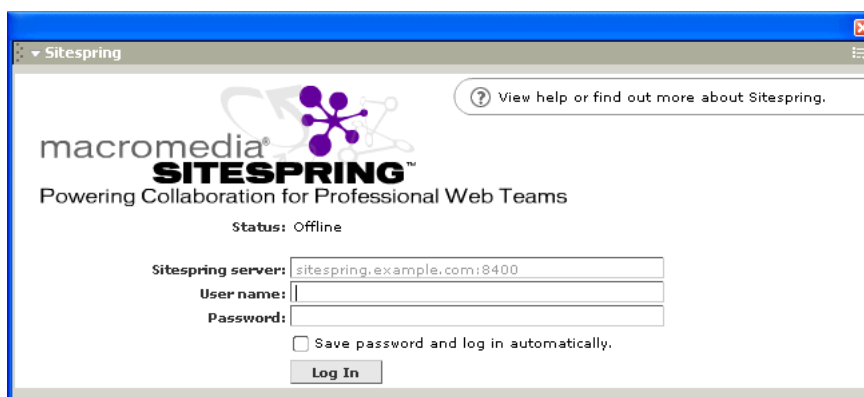
Logging out of Sitespring in Dreamweaver

When you click the Log Out link from the Sitespring panel, you also log out of Sitespring in Dreamweaver.

To log out of Sitespring:

- 1 In the Sitespring panel, click the Log Out link.

The Sitespring Log In appears.



- 2 Close the dialog box.

The next time you access the Sitespring panel, you don't have to reenter the URL when you connect (unless you want to access a different Sitespring server), but you will have to enter your password.

CHAPTER 5

Setting Up a Document

Macromedia Dreamweaver MX offers a flexible environment for working with a variety of web design and development documents. In addition to HTML documents, you can create and open CFML, ASP, JavaScript, CSS, or text-based documents.

The New Document dialog box provides several types of documents from which to create a new document. From this easy-to-use interface you can create a new blank document or template, a document based on one of your existing templates, or a document or template based on one of the designed page layouts that is included with Dreamweaver. You can use these basic designed page's layouts and templates to quickly get started developing professional looking web pages.

Other document options are also available in the New Document dialog box. You can select text-based documents, such as JavaScript or Cascading Style Sheet (CSS) documents, and dynamic page documents such as Macromedia ColdFusion, Microsoft Active Server Pages (ASP), and PHP pages. If you typically work with one type of document, you can set it as the default document type for new pages you create.

In Dreamweaver, you can easily define document properties, such as meta tags, document title, and background colors and several other page properties in either the Design view or Code view.

This chapter includes the following topics:

- “Working with the New Document dialog box” on page 108
- “Opening existing documents” on page 111
- “Setting document properties” on page 112
- “Selecting elements in the Document window” on page 116
- “Using visual guides in the design process” on page 117
- “About automating tasks” on page 120

Creating Dreamweaver documents

The New Document dialog box presents you with several types of documents from which to create a new page. Dreamweaver opens the New Document dialog box when you choose File > New. If you work predominately with one document type, for example HTML pages, ColdFusion pages, or Active Server Pages, you may want to set up a default document type. Setting up a default document type allows you to open a new document of a preferred document type when you create a new document without opening the New Document dialog box.

Related topics

“Creating a new blank document” on page 109

“Opening existing documents” on page 111

“Creating a document based on an existing template” on page 110

“Creating a document based on a Dreamweaver design file” on page 109

Working with the New Document dialog box

The New Document dialog box provides you with several choices for selecting a new document to work in. You can create a new document in the following ways:

- You can start with a blank document.
- You can use a template that defines how a document looks, and sets which parts of a document can be edited.

To open the New Document dialog box:

- Choose File > New.

The New Document dialog box contains two tabs: General and Templates.

The **General** tab contains various document types you can use to create a new blank page. You can select a new blank document from the Basic Page, Dynamic Page, and Frameset categories. Create a page based on a special file type such as CSS, JavaScript, VBScript or Text files by selecting a document in the Other category. The Templates category provides you with a choice of document types from which you can create a blank templates: HTML templates as well as templates in which you can insert server behaviors.

The CSS Style Sheets, Page Designs, and Pages Designs (Accessible) categories, provide you with pre-built design files you can use to create your own pages. In the case of CSS style sheets, you can copy a pre-designed styles sheets then apply it to your document.

Depending on the type of document you select, other options appear in the dialog box; for example, if you select HTML document you have the option of making the document XHTML compliant, and when you select a Page Designs file, you have the option of creating a document or creating a template.

The **Templates** tab contains a list of your defined Dreamweaver sites. You can choose a template from any of the listed sites to create new documents based on that template.

Select a category to see a list of available document types in the Document list to its right. Clicking a document in the list provides you with a description, and in the case of Page Designs, CSS Style Sheets, and Framesets categories document, a preview of the selected document.

You can use the Preferences option located at the bottom of the New Document dialog box to set default document preferences, such as a document type, encoding, and a file extension. Additionally, you can set an option to automatically open a new document without using the New Document dialog box. For information about setting document preferences see Setting New Document preferences in Dreamweaver Help.

Use the Get More Content link in the New Document dialog box to go to the Dreamweaver Exchange to download more page design content.

Creating a new blank document

You can use the New Document dialog box to select the type of document you want to create.

If you typically work with a specific document type, you can set a default document and automatically open a new document based on the default document you've defined. For information, see Setting New Document preferences in Dreamweaver Help.

To create a new blank document:

1 In Dreamweaver, choose File > New.

The New Document dialog box appears. The General tab is already selected.

2 In the Category list select the category of document you want to create.

For example, select Basic Page to create an HTML document, or select Dynamic page to create a ColdFusion or ASP document, and so on.

3 In the document list, select the page type you want to create, then do one of the following:

- Click Create.
- Double-click on the item in the document list.
- Press Enter.

The dialog box closes and a new document appears in the Document window.

Creating a document based on a Dreamweaver design file

Dreamweaver comes with several professionally developed page layout and design element files. You can use these *design files* as starting points for designing pages in your sites.

The design files include documents and templates which adhere to accessibility standards, table-based page layout documents, and CSS Style Sheets. You can preview a document and read a brief description of a document's design elements for items in the CSS Style Sheets, Framesets, Page Designs, and Page Designs (Accessible) categories.

When you create a document based on a design file, Dreamweaver creates a copy of the file. If the design file includes links to asset files (such as graphics, Flash elements, or an external CSS Style Sheet), when you save the document Dreamweaver prompts you to save a copy of the dependent files. You can choose your own location for the dependent files or use the default folder location Dreamweaver generates (which is based on the design file's source name).

You can also create a new template based on a design file. The template already has editable regions defined. You can create new editable regions in the template. You'll be prompted to save the file as a template for your current site. Linked files will also be copied to your site.

Note: If you create a document based on a predefined frameset only the frameset structure is copied, not the frame contents. Additionally, you will have to save each frame file separately. For information about saving frames, see "Saving frame and frameset files" on page 263.

To create a Cascading Style Sheet:

- 1 Open the New Document dialog box, choose File > New.
- 2 In the Category list, select CSS Style Sheets.
The document list to the right, updates with a selection of CSS Style Sheets.
- 3 In the document list, select a style sheet to preview its properties and description, then do one of the following:
 - Click Create.
 - Double-click on the item in the document list.
 - Press Enter.The dialog box closes and a new CSS document appears in the Document window. The CSS Style Sheet opens in Code view.
- 4 Choose File > Save As.
- 5 In the Save As dialog box, browse to the location where you want to save the file, enter a unique filename for the document, and click Save.

To create a Page Designs document:

- 1 Open the New Document dialog box, choose File > New.
- 2 In the Category list, select Page Design or Page Design (Accessible) depending on your design requirements.
- 3 In the Page Designs list select the document you want to base the new page on.
- 4 Do one of the following:
 - To create a document, select the Document radio button.
 - To create a new template, select the Template radio button.
- 5 Click Create.
A new document appears in the Document window. If you create a template, you'll see editable regions defined and you'll be prompted to save the document as a template for your current site.
- 6 Choose File > Save to save the document.
If the file contains links to asset files the Copy Dependent Files dialog box appears.
- 7 In the Copy Dependent Files dialog box, set options to copy the assets to your current site.
- 8 Click Copy to copy the assets to the selected folder and close the dialog box.

Creating a document based on an existing template

You can use the New Document dialog box to select, preview, and create a new document from an existing template. You can select a template from any of your Dreamweaver-defined sites.

Documents in the Page Design category of the New Document dialog box also let you create templates. Once you save a page design document as a template in your site you can create pages based on that template. For information, see "Creating a document based on a Dreamweaver design file" on page 109.

For more information about working with templates and documents based on templates, see Chapter 27, “About Dreamweaver templates,” on page 435.

To create a new document based on a template:

- 1 Choose File > New to open the New Document dialog box.
- 2 In the Templates tab, in the Templates For list, select the Dreamweaver site that contains the template you want use.
- 3 The Site list displays templates in the selected site, if there are any.
- 4 Select the template you want to use.
- 5 Deselect the Update Page When Template Changes option if you want to detach the new document from the template.
- 6 Click Create.
A new document is created.
- 7 Save the document.

Saving a document

When you save a document, avoid using spaces and special characters in file and folder names. In particular, do not use special characters (such as é, ç, or ¥) or punctuation (such as colons, slashes, or periods) in the names of files you intend to put on a remote server; many servers change these characters during upload, which will cause any links to the files to break. Also, do not begin a filename with a numeral.

To save a document:

- 1 Choose File > Save.
- 2 In the dialog box that appears, navigate to the folder where you want to save the file.
- 3 In the File Name text field, type a name for the file.
- 4 Click Save to save the file.

Opening existing documents

In Dreamweaver, you can open an existing HTML document or any dynamic document type, even if it wasn't created in Dreamweaver. You can open the document and use Dreamweaver to edit it taking advantage of Design view or Code view authoring.

You can also open non-HTML text files, such as JavaScript files, XML files, CSS Style Sheets, or text files saved by word processors or text editors.

Certain files open in Code view only and allow you to make edits to the code in the file, such as a CSS document. You can update the document while working in Dreamweaver, then save the changes in the file.

If the document you want to open is a Microsoft Word 97, Word 98 or Word 2000 file which was saved as HTML, you may want to import the document into Dreamweaver rather than open it. When you import a Word HTML file Dreamweaver prompts you to clean up the extraneous markup tags Word inserts into HTML files. You can use the Clean Up Word HTML command to set a page background color and to clean up CSS Style Sheet formatting in the imported document.

To open an existing file, do one of the following:

- 1 In Dreamweaver, choose File > Open.
- 2 In the dialog box that opens, navigate to and select the file you want to open.
- 3 Click Open.

The document opens in the Document window.

Note: JavaScript, text, and CSS Style Sheets open in Code view by default. For information about changing setting an external text editor for viewing these types of files, see “Launching an external editor for media files” on page 320.

To open and import a Microsoft Word HTML file:

- 1 In Dreamweaver, choose File > Import > Word HTML.

The Select Word HTML File to Import dialog box opens.

- 2 In the dialog box, navigate to and select the document you want to open.
- 3 Click Open.

Dreamweaver opens the document in a new window and the Clean Up Word HTML dialog box opens.

- 4 Choose options in the Basic or Detailed tab to select the cleanup options you want to apply to the document.

For more information about the cleanup options, see *Cleaning up Microsoft Word HTML* in Dreamweaver Help.

- 5 Click OK.

Dreamweaver applies the clean up settings to the new document and a log of the changes appears.

- 6 Save the document.

Setting document properties

Page titles, background images and colors, text and link colors, and margins are basic properties of every HTML document. The page title identifies and names the document. A background image or color sets the overall appearance of the document. Text and link colors help site visitors distinguish regular text from hypertext and to see which links have been visited and which have not.

For information about completing the Page Properties dialog box, see *Selecting Page Properties* in Dreamweaver Help.

Changing the document title

The title of an HTML page helps site visitors keep track of what they're viewing as they browse, and it identifies the page in the visitor's history and bookmark lists. If you don't title a page, the page will appear in the browser window, bookmark lists, and history lists as *Untitled Document*. Note that giving the document a filename (by saving it) is not the same as giving the page a title. To locate all untitled documents in your site, use the Site > Reports command; see "Using Reports to test a site" on page 468.

To change the title of a page:

- 1 With the document open, do one of the following:
 - Choose Modify > Page Properties.
 - Choose View > Toolbar (if it isn't already selected).
 - Right-click (Windows) or Control-click (Macintosh) in an empty area in the document, then select Page Properties.
- 2 In the Title text box, enter the title for the page, then press Enter or Return.
- 3 If you're editing the title in the Page Properties dialog box, click OK.

The title appears in the title bar of the Document window (and in the toolbar, if it's showing). The filename of the page and the folder the file is saved in appears in parentheses next to the title in the title bar. An asterisk indicates the document contains changes that have not yet been saved.

Setting a background image or background page color

To define an image or color for the page background, use the Page Properties dialog box. If you use both a background image and a background color, the color appears while the image downloads, and then the image covers up the color. If the background image contains any transparent pixels, the background color shows through.

To define a background image or color:

- 1 Choose Modify > Page Properties, or select Page Properties from the context menu in the Design view of the Document window.
- 2 To set a background image, click the Browse button, then browse to and select the image. Alternatively, enter the path to the background image in the Background Image box.

Dreamweaver tiles (repeats) the background image if it does not fill the entire window, just as browsers do. (To prevent the background image from tiling, use Cascading Style Sheets to disable image tiling. See Defining CSS Style Background properties in Dreamweaver Help.
- 3 To set a background color, click the Background color box and select a color from the color picker.

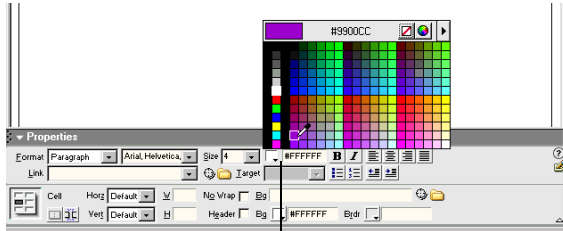
Working with colors

In Dreamweaver, many of the dialog boxes, as well as the Property inspector for many page elements, contain a color box, which opens a color picker. Use the color picker to choose a color for a page element.

To choose a color in Dreamweaver:

- 1 Click a color box in any dialog box or in the Property inspector.

The color picker appears.



Property inspector color box

- 2 Do one of the following:

- Use the eyedropper to select a color swatch from the palette. All colors in the Color Cubes (default) and Continuous Tone palettes are web-safe; other palettes are not. (For more information, see “About web-safe colors” on page 115.)
- Use the eyedropper to pick up a color from anywhere on your screen—even outside the Dreamweaver windows. (If you click the desktop or another application, Dreamweaver picks up the color where you clicked, but you may switch into the other application; in that case, click a Dreamweaver window to continue working in Dreamweaver, or hold down the mouse button when moving from Dreamweaver to the desktop to keep from switching out of Dreamweaver.)
- To expand your color selection, use the pop-up menu at the top right corner of the color picker. You can select Color Cubes, Continuous Tone, Windows OS, Mac OS, Grayscale, and Snap to Web Safe.

Note that the Color Cubes and Continuous Tone palettes are web safe, whereas Windows OS, Mac OS and Grayscale are not. If you are using a palette that isn't web-safe and then choose Snap to Web Safe, Dreamweaver replaces the selected color with the closest web-safe color. In other words, you may not get the color you see.

- To clear the current color without choosing a different color, click the Strikethrough button.
- To open the system color picker, click the Color Wheel button. For more information, see “About web-safe colors” on page 115.

About web-safe colors

In HTML, colors are expressed either as hexadecimal values (for example, #FF0000) or as color names (red). A web-safe color is one that appears the same in Netscape Navigator and Microsoft Internet Explorer on both Windows and Macintosh systems when running in 256-color mode. The conventional wisdom is that there are 216 common colors, and that any hexadecimal value that combines the pairs 00, 33, 66, 99, CC, or FF (RGB values 0, 51, 102, 153, 204, and 255, respectively) represents a web-safe color.

Testing, however, reveals that there are only 212 web-safe colors rather than a full 216, because Internet Explorer on Windows does not correctly render the colors #0033FF (0,51,255), #3300FF (51,0,255), #00FF33 (0,255,51), and #33FF00 (51,255,0).

The Color Cubes (default) and the Continuous Tone palettes in Dreamweaver use the 216-color web-safe palette; selecting a color from these palettes displays the color's hexadecimal value.

To choose a color outside the web-safe range, open the system color picker by clicking the Color Wheel button in the upper right corner of the Dreamweaver color picker. The system color picker is not limited to web-safe colors.

UNIX versions of Netscape Navigator use a different color palette than the Windows and Macintosh versions. If you are developing exclusively for UNIX browsers (or your target audience is Windows or Macintosh users with 24-bit monitors and UNIX users with 8-bit monitors), consider using hexadecimal values that combine the pairs 00, 40, 80, BF, or FF, which produce web-safe colors for SunOS.

Defining default text colors

Define default colors for regular text, links, visited links, and active links in the Page Properties dialog box, or choose a preset color scheme to define the page background and text colors. (See “Working with colors” on page 114.)

Note: The active link color is the color that a link changes to while it's being clicked. Some web browsers may not use the color you specify.

To define default text colors, do one of the following:

- Choose Modify > Page Properties and then choose colors for the Text Color, Link Color, Visited Links, and Active Links options.
- Choose Commands > Set Color Scheme and then choose a background color and a color set for text and links.

The sample box shows how the color scheme will look in the browser.

Selecting elements in the Document window

To select an element in the Design view of the Document window, you generally click the element. If an element is invisible, you must make it visible before you can select it.

To select elements, use these techniques:

- To select a visible element in the Document window, click the element or drag across the element.
- To select an invisible element, choose View > Visual Aids > Invisible Elements (if that menu item isn't already selected) and then click the element's marker in the Document window.

Some objects appear on the page in a place other than where their code is inserted. For example, a layer can be anywhere on the page, but the code defining the layer is in a fixed location. When invisible elements are showing, Dreamweaver displays markers in the Document window to show the location of the code for such elements. Selecting a marker selects the entire element; for example, selecting the marker for a layer selects the entire layer. (See "About invisible elements" on page 116.)

- To select a complete tag (including its contents, if any), click a tag in the tag selector at the bottom left of the Document window. (The tag selector appears in both Design view and Code view.) The tag selector always shows the tags that contain the current selection or insertion point. The leftmost tag is the outermost tag containing the current selection or insertion point. The next tag is contained in that outermost tag, and so on; the rightmost tag is the innermost one that contains the current selection or insertion point.

In the following example, the insertion point is in a paragraph tag, `<p>`. To select the table in which the paragraph marker exists, you would select the `<table>` tag to the left of the `<p>` tag.

```
<body> <table> <tr> <td> <btable> <tr> <td> <p>
```

To see the HTML code associated with the selected text or object, do one of the following:

- In the Document toolbar, click the Show Code View button.
- Select View > Code.
- In the Document toolbar, click the Show Code and Design Views button.
- Select View > Code and Design.
- Select Window > Others > Code Inspector.

For more information about the Code view, see "Viewing your code" on page 175.

When you select something in either code editor (the Code view or the Code inspector), it's generally also selected in the Document window. You may need to synchronize the two views before the selection appears; see "Viewing your code" on page 175.

About invisible elements

Some HTML code doesn't have a visible representation in a browser. For example, comment tags don't appear in browsers. However, it can be useful while you're creating a page to be able to select such invisible elements, edit them, move them, and delete them. Dreamweaver allows you to specify whether it shows icons marking the location of invisible elements in the Design view of the Document window.

To show or hide marker icons for invisible elements, choose View > Visual Aids > Invisible Elements. Showing invisible elements lets you select them and change their properties in the Property inspector; hiding them lets you see the page closer to the way it will appear in a browser. Note that showing invisible elements may slightly change the layout of a page, moving other elements by a few pixels, so for precision layout, hide the invisible elements.

To indicate which element markers appear when you choose View > Visual Aids > Invisible Elements, you can set options in Invisible Elements preferences. For example, you can specify that named anchors should be visible, but not line breaks. For an explanation of each Invisible Elements preference, see Setting Invisible Elements preferences in Dreamweaver Help.

You can create certain invisible elements (such as comments and named anchors) using buttons in the Common category of the Insert bar (see “Using the Insert bar” on page 42). You can then modify these elements using the Property inspector.

Using visual guides in the design process

Dreamweaver provides several kinds of visual guides to help you design documents and predict (approximately) how they will appear in browsers. You can do all of the following:

- Instantly snap the Document window to a desired window size to see how the elements fit on the page. See “Resizing the Document window” on page 38.
- Use rulers to provide a visual cue for positioning and resizing layers or tables. See “Displaying rulers” on page 117.
- Use a tracing image as the page background to help you duplicate a design created in an illustration or image-editing application such as Macromedia Fireworks. See “Using a tracing image” on page 118.
- Use the grid for precise positioning and resizing of layers. Grid marks on the page help you align layers, and when snapping is enabled, layers automatically snap to the closest grid point when moved or resized. (Other objects, such as images and paragraphs, do not snap to the grid.) Snapping works regardless of whether the grid is visible. See “Snapping layers to the grid” on page 383.

Displaying rulers

Rulers can be displayed on the left and top borders of the page, marked in pixels, inches, or centimeters.

To change ruler settings, do one of the following:

- To toggle rulers on and off, choose View > Rulers > Show.
- To change the origin, drag the ruler-origin icon anywhere on the page. (This icon appears at the upper left corner of the Design view of the Document window when rulers are showing.) To reset the origin to its default position, choose View > Rulers > Reset Origin.
- To change the unit of measure, choose Pixels, Inches, or Centimeters from the View > Rulers submenu.

Using a tracing image

Use a tracing image as a guide to re-create a page design that was mocked up in a graphics application. A tracing image is a JPEG, GIF, or PNG image that is placed in the background of the Document window. You can hide the image, set its opacity, and change its position.

The tracing image is visible only in Dreamweaver. It is never visible when you view the page in a browser. When the tracing image is visible, the page's real background image and color are not visible in the Document window; however, the background image and color will be visible when the page is viewed in a browser.

To place a tracing image in the Document window:

1 Do one of the following:

- Choose View > Tracing Image > Load.
 - Choose Modify > Page Properties and then click the Browse button next to the Tracing Image text box.
- 2 In the dialog box that appears, select an image file and click Select (Windows) or Choose (Macintosh).
- 3 The Page Properties dialog box appears. Specify the transparency for the image by dragging the Image Transparency slider; then click OK.

To switch to another tracing image or change the transparency of the current tracing image at any time, choose Modify > Page Properties.

To show or hide the tracing image:

Choose View > Tracing Image > Show.

To change the position of a tracing image:

Choose View > Tracing Image > Adjust Position. Then do one of the following:

- To precisely specify the position of the tracing image, enter coordinate values in the X and Y text boxes.
- To move the image one pixel at a time, use the arrow keys.
- To move the image five pixels at a time, press Shift and an arrow key.

To reset the position of the tracing image:

Choose View > Tracing Image > Reset Position. The tracing image returns to the upper left corner of the document window (0,0).

To align the tracing image to a selected element:

- 1 Select an element in the Document window.
- 2 Choose View > Tracing Image > Align with Selection.

The upper left corner of the tracing image is aligned with the upper left corner of the selected element.

Viewing and editing head content

HTML files are made up of two main sections: the `head` section and the `body` section. The `body` section is the main part of the document, the visible part containing text and images and so on. The `head` section is invisible, except for the document title, which appears in window title bars in browsers and in Dreamweaver. Give a title to every page you create.

The `head` section also contains other important information, including the document type, the language encoding, JavaScript and VBScript functions and variables, keywords and content indicators for search engines, and style definitions. You don't need to provide all of these elements for every page; you might, for example, provide keywords and content indicators for your home page only. You can view the elements in the `head` section using the View menu, the Document window's Code view, or the Code inspector.

To view elements in the head section of a document:

Choose View > Head Content. For each element of the `head` content, a marker appears at the top of the Document window in Design view.

Note: If your Document window is set to show only Code view, View > Head Content is dimmed.

To insert an element into the head section of a document:

- 1 Do one of the following:
 - In the Insert bar's Head category, select click one of the object buttons.
 - Choose an item from the Insert > Head Tags submenu.
- 2 Enter options for the element in the dialog box that appears, or in the Property inspector.

To edit an element in the head section of a document:

- 1 Choose View > Head Content.
- 2 Click one of the icons in the `head` section to select it.
- 3 Set or modify the properties of the element in the Property inspector.

For information about the properties of specific `head` elements, see the following topics in Dreamweaver Help:

- Setting Meta properties
- Setting Title properties
- Setting Keywords properties
- Setting Description properties
- Setting Refresh properties
- Setting Base properties
- Setting Link properties

About automating tasks

While creating your documents, you may want to perform the same task numerous times. This section explains how to use the History panel to automate repetitive tasks.

To repeat a series of steps once or twice, replay them directly from the History panel, which records your steps as you work on a document. (For basic information about the History panel, see “About the History panel” on page 48.) To automate a task that you perform often, you can create a new command that performs that task automatically.

Some mouse movements, such as selecting something in the Document window by clicking or dragging, can't be played back or saved as part of saved commands. When you make such a movement, a black line appears in the History panel (although the line does not become obvious until you perform another action). To avoid movements that can't be played back, use the arrow keys instead of the mouse to move the insertion point within the Document window. To make or extend a selection, hold down the Shift key while pressing an arrow key.

Note: If a black mouse-movement indicator line appears while you're performing a task you want to repeat later, you can undo back past that step and try another approach, perhaps using the arrow keys.

Certain other steps also aren't repeatable, such as dragging a page element to somewhere else on the page. When you perform such a step, a menu-command icon with a small red *X* appears in the History panel.

Steps are replayed exactly as they were originally performed; you can't modify the steps as you replay them. For example, if you previously changed the color of a table cell to red, applying that step to another table cell changes that cell's color to red as well; you can't specify a different color when you apply the step to a new cell.

Repeating steps

To repeat the last step you performed, use the Edit > Repeat command, or use the keyboard shortcut Control+Y (Windows) or Command+Y (Macintosh). The name of this command changes in the Edit menu to reflect the last step you performed; for example, if you've just typed some text, the command name is Repeat Typing. (You can't use the Repeat command immediately after an Undo or Redo operation.)

To repeat steps other than the most recent one, or to repeat multiple steps at once, use the History panel. (For basic information about the History panel, see “About the History panel” on page 48.)

Note that when you replay steps, the steps that are played are the steps that are selected (highlighted), not necessarily the step currently pointed to by the slider.

To repeat one step:

In the History panel, select a step and click the Replay button. The step is replayed, and a copy of it appears in the History panel.

To repeat a series of adjacent steps:

1 Select steps in the History panel by doing one of the following:

- Drag from one step to another. (Don't drag the slider; just drag from the text label of one step to the text label of another step.)
- Select the first step, then Shift-click the last step; or select the last step and then Shift-click the first step.

Note: Although you can select a series of steps that includes a black mouse-movement indicator line, that mouse movement is skipped when you replay the steps.

- 2 Click Replay.

The steps are replayed in order, and a new step, labeled Replay Steps, appears in the History panel.

To repeat nonadjacent steps:

- 1 Select a step, then Control-click (Windows) or Command-click (Macintosh) other steps.

You can also Control-click or Command-click to deselect a selected step.

- 2 Click Replay.

The selected steps are replayed in order, and a new step, labeled Replay Steps, appears in the History panel.

Applying steps to another object

You can apply a set of steps from the History panel to any object in the Document window.

To apply History panel steps to a new object:

- 1 Select the object.
- 2 Select the relevant steps in the History panel, then click Replay.

Applying steps to multiple objects

If you select multiple objects in a document and then apply steps to them from the History panel, the objects are treated as a single selection, and Dreamweaver attempts to apply the steps to that combined selection. For example, you can't select five images and apply the same size change to each of them all at once; a size change is an operation that must be applied to each individual image, not to a collective combination of images.

To apply a series of steps to each object in a set of objects, you must make the last step in the series select the next object in the set. The following procedure demonstrates this principle in a particular scenario: setting the vertical and horizontal spacing of a series of images.

To set the vertical and horizontal spacing of a series of images:

- 1 Start with a document in which each line consists of a small image (such as a graphical bullet or an icon) followed by text. The goal is to set the images off from the text and from the other images above and below them.

■ **Locations**

■ **Special Offers**

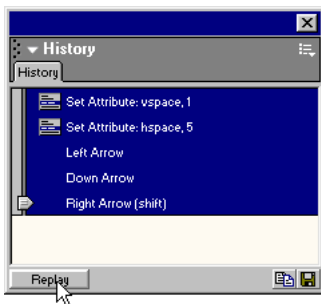
■ **Customer Service**

- 2 Open the Property inspector (Window > Properties), if it isn't already open.
- 3 Select the first image.
- 4 In the Property inspector, enter numbers in the V Space and H Space text boxes to set the image's spacing.
- 5 Click the image again to make the Document window active without moving the insertion point.

- 6 Press the Left Arrow key to move the insertion point to the left of the image. Then press the Down Arrow key to move the insertion point down one line, leaving it just to the left of the second image in the series. Then press Shift+Right Arrow to select that second image.

Note: Do not select the image by clicking it, or you won't be able to replay all the steps.

- 7 In the History panel, select the steps that correspond to changing the image's spacing and selecting the next image. Click the Replay button to replay those steps.



The current image's spacing changes, and the next image is selected.

■ Locations

■ Special Offers

■ Customer Service

- 8 Continue to click Replay until all the images are spaced correctly.

To apply steps to an object in another document, use the Copy Steps button; see “Copying and pasting steps between documents” on page 122.

Copying and pasting steps between documents

Each open document has its own history of steps. You can copy steps from one document and paste them into another.

Note: Copy Steps (a button in the History panel) is different from Copy (in the Edit menu). You can't use Edit > Copy to copy steps, though you do use Edit > Paste to paste them.


Closing a document clears its history. If you know you will want to use steps from a document after that document is closed, copy the steps with Copy Steps (or save them as a command; see “Creating new commands from history steps” on page 123) before you close the document.

Be careful when copying steps that include a Copy or a Paste command:

- Don't use Copy Steps if one of the steps is a Copy command; you may not be able to paste such steps the way you want.
- If your steps include a Paste command, you can't paste those steps, unless the steps also include a Copy command before the Paste command.

If you paste steps into a text editor or into the Code view or the Code inspector, they appear as JavaScript code. This can be useful for learning to write your own scripts. For more information on using JavaScript in Dreamweaver, see “Writing and editing code” on page 191.

To reuse steps from one document in another document:

- 1 Start from the document containing the steps you want to reuse.
- 2 Select the steps in the History panel.
-  3 Click the History panel’s Copy Steps button to copy those steps.
- 4 Open the other document.
- 5 Place the insertion point where you want it, or select an object to apply the steps to.
- 6 Choose Edit > Paste to paste the steps.

The steps are played back as they’re pasted into the document’s History panel. The History panel shows them as only one step, called Paste Steps.

Creating new commands from history steps

You can save a set of history steps as a named command, which then becomes available in the Commands menu.

Create and save a new command if there’s any chance you’ll want to use a given set of steps again in the future, especially if you want to use those steps again the next time you start Dreamweaver; saved commands are retained permanently (unless you delete them), while recorded commands are discarded when you exit from Dreamweaver, and copied sequences of steps are discarded when you copy something else.

You can edit the names of commands that you’ve placed in the Commands menu, and you can delete them from the Commands menu, using Commands > Edit Command List. It’s more complicated to edit and delete commands that are built into the Commands menu (that is, commands that you didn’t explicitly add).

To create a command:

- 1 Select a step or set of steps in the History panel.
- 2 Click the Save As Command button, or choose Save As Command from the History panel’s context menu.
- 3 Enter a name for the command and click OK.

The command appears in the Commands menu.

Note: The command is saved as a JavaScript file (or sometimes an HTML file) in your Dreamweaver/Configuration/Commands folder.

To use a saved command:

- 1 Select an object to apply the command to, or place the insertion point where you want it.
- 2 Choose the command from the Commands menu.

To edit the names of commands in the Commands menu:

- 1 Choose Commands > Edit Command List.
- 2 Select a command to rename and enter a new name for it.
- 3 Click Close.

To delete a name from the Commands menu:

- 1 Choose Commands > Edit Command List.
- 2 Select a command.
- 3 Click Delete, then click Close.

Recording commands

Dreamweaver allows you to record a temporary command for short-term use. The main differences between this approach and playing back steps from the History panel (see “Repeating steps” on page 120) are as follows:

- The steps are recorded as you perform them, so you don’t have to select them in the History panel before playing them back.
- During recording, Dreamweaver prevents you from performing nonrecordable mouse movements (such as clicking to select something in a window, or dragging a page element to a new location).
- If you switch to another document while recording, Dreamweaver doesn’t record the changes you make in the other document. To determine whether you’re recording or not at any given moment, look at the mouse pointer.

Dreamweaver retains only one recorded command at a time; as soon as you start recording a new command, the old one is lost. To save a new command without losing a recorded one, save the command from the History panel.

Once you’ve recorded a command, you can save it using the History panel.

To temporarily record a series of frequently used steps:

- 1 Choose Commands > Start Recording, or press Control+Shift+X (Windows) or Command+Shift+X (Macintosh).

The pointer changes to indicate that you’re recording a command.

- 2 When you finish recording, choose Commands > Stop Recording, or press Control+Shift+X (Windows) or Command+Shift+X (Macintosh).

To play back a recorded command:

Choose Commands > Play Recorded Command, or press Control+Shift+R (Windows) or Command+Shift+R (Macintosh).

To save a recorded command:

- 1 Choose Commands > Play Recorded Command to play the command back.
A step named Run Command appears in the History panel’s step list.
- 2 Select the Run Command step and click the Save As Command button.
- 3 Enter a name for the command and click OK.

Note: The command appears in the Commands menu.

Part II

Preparing to Build Dynamic Sites

If you're building a dynamic web application, start by setting up an application server and connecting to a database.

This part contains the following chapters:

- Chapter 6, “Setting Up a Web Application”
- Chapter 7, “Database Connections for ColdFusion Developers”
- Chapter 8, “Database Connections for ASP.NET Developers”
- Chapter 9, “Database Connections for ASP Developers”
- Chapter 10, “Database Connections for JSP Developers”
- Chapter 11, “Database Connections for PHP Developers”

CHAPTER 6

Setting Up a Web Application

This chapter describes how to configure your system to build web applications in Macromedia Dreamweaver MX.

Note: If you want to build the sample web application installed with Dreamweaver, see Dreamweaver Help (Help > Using Dreamweaver).

This chapter contains the following sections:

- “A note for Dreamweaver UltraDev 4 users” on page 127
- “What you need to build web applications” on page 128
- “Setting up a web server” on page 129
- “Setting up an application server” on page 129
- “Creating a root folder for the application” on page 132
- “Defining a Dreamweaver site” on page 132
- “Connecting to a database” on page 135
- “Troubleshooting application server errors” on page 136

A note for Dreamweaver UltraDev 4 users

If you're a Macromedia Dreamweaver UltraDev 4 user, your system is probably already configured to build web applications. If you installed Dreamweaver MX alongside UltraDev 4, then all your UltraDev sites were automatically imported into Dreamweaver. You don't need to redefine the sites in Dreamweaver MX.

However, if an imported site contains Macromedia ColdFusion pages built using UltraDev 4 server behaviors, you must tell Dreamweaver that the site contains UltraDev-generated pages. The reason is that Dreamweaver MX looks for code patterns on the page to identify and display server behaviors in the Server Behaviors panel. For Microsoft Active Server Pages (ASP) and JavaServer Pages (JSP), Dreamweaver MX generates the same code as UltraDev, so it recognizes and displays UltraDev-generated ASP and JSP server behaviors. For ColdFusion pages, however, Dreamweaver MX generates improved and easier-to-understand code. Because Dreamweaver looks for these new code patterns by default, it does not recognize UltraDev-generated ColdFusion server behaviors. You must tell Dreamweaver to look for the UltraDev code patterns if you want to keep working with the old server behaviors.

To tell Dreamweaver to look for ColdFusion code generated by UltraDev 4:

- 1 Open the Site Definition dialog box by selecting Site > Edit Sites, selecting your site, and clicking Edit.
- 2 If the wizard is showing, click Advanced then select Testing Server from the Category list. The Testing Server dialog box appears.
- 3 In the This Site Contains pop-up menu, choose one of the following options:
 - If you want Dreamweaver to continue generating UltraDev code for new pages, choose UltraDev 4 Pages Only.
 - If you want Dreamweaver to start using the improved code for new pages, choose Both Versions. This will incrementally upgrade the UltraDev 4 site; over time, the site will consist mostly of pages with the new server behaviors.
- 4 Click OK, then click Done.

Databases connections are handled differently if you specify that your site contains UltraDev 4 pages only. For more information, see “Connecting using UltraDev 4 connectivity” on page 141.

What you need to build web applications

To build web applications in Dreamweaver, you need the following software:

- A web server
- An application server that runs on your web server, or a web server that doubles as an application server, such as Microsoft PWS or IIS

Note: In the context of web applications, the terms “web server” and “application server” refer to software, not hardware.

If you want to use a database with your application, you need the following additional software:

- A database or database system
- A database driver that supports your database

For information on setting up a database for your web application, see “Connecting to a database” on page 135.

Several web hosting companies offer plans that let you use their software to test and deploy web applications. If you're a Windows user, you can install the required software on the same computer as Dreamweaver for development purposes. You can also install the software on a network computer (typically a Windows NT or Windows 2000 computer) so that other developers on your team can work on a project.

If you're a Macintosh user, you can use a web hosting service or install the required software on a remote computer. If you're a Mac OS 10.1 user, you can develop PHP sites locally using the Apache web server and PHP application server installed with your operating system. For setup information, see the following websites:

- <http://developer.apple.com/internet/macosx/php.html>
- <http://www.entropych.com/software/macosx/>
- <http://www.stepwise.com/Articles/Workbench/2001-10-11.01.html>

Setting up a web server

To run web applications, you need a web server. A web server is software that serves files in response to requests from web browsers. A web server is sometimes called an HTTP server. Common web servers include IIS, Netscape Enterprise Server, iPlanet Web Server, and Apache HTTP Server.

If you're not using a web hosting service, choose a web server and install it on your local computer or on a remote computer. For more information, see the server vendor's documentation or your system administrator.

Windows users can get a web server up and running quickly on their local computer by installing either PWS or IIS. The web server may already be installed. Check your folder structure to see if it contains a C:\Inetpub or D:\Inetpub folder. PWS and IIS create this folder during installation. If you want to install PWS or IIS, see Dreamweaver Help (Help > Using Dreamweaver).

ASP.NET pages only work with one web server: Microsoft IIS 5 or higher. PWS is not supported. Also, because IIS 5 is a service of the Windows 2000 and Windows XP Professional operating systems, you can only use these two versions of Windows to run ASP.NET applications. Windows 98, ME, or NT are not supported. Note, however, that you can develop (as opposed to run) ASP.NET applications on any computer running Dreamweaver MX, including the Macintosh.

Setting up an application server

To run web applications, your web server needs to work with an application server. An application server is software that helps a web server process specially marked web pages. When such a page is requested, the web server sends the page to the application server for processing before sending the page to the browser.

Common application servers include Macromedia ColdFusion MX, Macromedia JRun, Microsoft .NET Framework, PHP, IBM WebSphere, and Jakarta Tomcat. The Microsoft IIS and PWS web servers also double as ASP application servers. The application server is typically installed on the same system that runs the web server.

Here are the topics covered in this section:

- “Choosing your application server” on page 129
- “Installing a ColdFusion application server” on page 130
- “Installing an ASP.NET application server” on page 130
- “Installing an ASP application server” on page 131
- “Installing a JSP application server” on page 131
- “Installing a PHP application server” on page 131

Choosing your application server

Your choice of application server depends on several factors, including your budget, the server technology you want to use (ColdFusion, ASP.NET, ASP, JSP, or PHP), and your choice of web server.

Budget: Some vendors sell high-end application servers that are expensive to buy and administer. Others vendors provide easier, more cost-effective solutions (examples include Macromedia ColdFusion and JRun servers). Some application servers are built into web servers (such as Microsoft IIS and PWS) and others can be downloaded for free from the Internet (such as Jakarta Tomcat and PHP).

Server technology: Application servers use different technologies. Dreamweaver supports five server technologies: ColdFusion, ASP.NET, ASP, JSP, and PHP. For more information, see Dreamweaver Help (Help > Using Dreamweaver). The following table shows common application servers available for the five server technologies supported by Dreamweaver:

Server technology	Application server
ColdFusion	Macromedia ColdFusion MX
ASP.NET	Microsoft IIS 5 with .NET Framework
ASP	Microsoft IIS or PWS Sun Chili!Soft ASP
JSP	Macromedia JRun IBM WebSphere Apache Tomcat BEA WebLogic
PHP	PHP server

Web server: Your choice of application server can also depend on the web server you want to use. Make sure the application works with your web server. For example, the .NET Framework only works with IIS 5 or higher.

Installing a ColdFusion application server

To run ColdFusion pages, you need the ColdFusion application server. This server is available for Windows, Linux, Solaris, and HP-UX systems.

You can download and install a fully functional, developer edition of ColdFusion MX from the Macromedia website at <http://www.macromedia.com/software/coldfusion/>. A copy of ColdFusion MX Server Developer Edition is also available on the Dreamweaver CD (Windows version only).

Macintosh users can use a web hosting service with a ColdFusion plan or install ColdFusion on a remote Windows, Linux, Solaris, or HP-UX computer running a web server.

After installing the application server, create a root folder for your web application. See “Creating a root folder for the application” on page 132.

Installing an ASP.NET application server

To develop ASP.NET pages, you need the following software:

- A Windows 2000 or Windows XP Professional computer running IIS 5 or later
- The Microsoft .NET Framework, which you can download from the Microsoft website

Download the .NET Framework from the Microsoft website at <http://asp.net/download.aspx> and follow the installation instructions on the website.

Macintosh users can use a web hosting service with an ASP.NET plan or install the .NET Framework on a remote Windows 2000 or Windows XP Professional computer running IIS 5 or later.

After installing the .NET Framework, create a root folder for your web application. See “Creating a root folder for the application” on page 132.

Installing an ASP application server

To develop ASP pages, you need an application server that supports Microsoft Active Server Pages 2.0. Here are some popular choices:

- Microsoft IIS, which comes with Windows NT Server, Windows 2000, and Windows XP Professional
- Microsoft PWS, a scaled-down version of IIS that runs in Windows 98 and NT Workstation
- Sun Chili!Soft ASP, versions of which run on Windows, Linux, Solaris, and other platforms. For more information, see the Chili!Soft website at <http://www.chilisoft.com/chiliasp/default.asp>

Windows users can install and run IIS or PWS on their local computer. For instructions, see Dreamweaver Help (Help > Using Dreamweaver).

Macintosh users can use a web hosting service with an ASP plan or install IIS or PWS on a remote computer.

After installing IIS or PWS, create a root folder for your web application. See “Creating a root folder for the application” on page 132.

Installing a JSP application server

To develop JSP pages, you need an application server that supports JavaServer Pages. Here are some popular choices:

- Macromedia JRun for Windows, Linux, Solaris, or UNIX. You can download a trial version from the Macromedia website at <http://www.macromedia.com/software/jrun/>.
- IBM WebSphere for various operating systems. You can download a trial version from the IBM website at <http://www-4.ibm.com/software/webservers/appserv/download.html>.
- Tomcat for Windows and UNIX. You can download a copy of Tomcat from the Jakarta Project website at <http://jakarta.apache.org/tomcat/>.

Macintosh users can use a web hosting service with a JSP plan or install a JSP application server on a remote computer running a web server.

After installing a JSP application server, create a root folder for your web application. See “Creating a root folder for the application” on page 132.

Installing a PHP application server

To run PHP pages, you need the PHP application server, which is open-source software available on the web. Editions of the application server exist for Windows, Linux, UNIX, HP-UX, Solaris, and Mac OS X systems. The application server works with the following web servers: Apache, Microsoft IIS or PWS, Netscape and iPlanet servers, and almost all web servers that support the CGI interface.

You can download the PHP application server from the PHP website at <http://www.php.net/downloads.php>. For information on installing the server, see the PHP documentation, which you can also download from the PHP website at <http://www.php.net/download-docs.php>.

Macintosh users can use a web hosting service with a PHP plan or install PHP on a remote computer running a web server. If you're a Mac OS 10.1 user, you can use the PHP application server installed with your operating system. For more information, see the following websites:

- <http://developer.apple.com/internet/macosx/php.html>
- <http://www.entropy.ch/software/macosx/>
- <http://www.stepwise.com/Articles/Workbench/2001-10-11.01.html>

After installing the PHP application server, create a root folder for your web application.

Creating a root folder for the application

After the server software is installed, create a root folder for your web application on the computer running the web server.

Make sure the folder is published by the web server—in other words, the web server can serve any file in this folder or in any of its subfolders in response to an HTTP request from a web browser. For example, on a computer running PWS or IIS, any file in the `Inetpub\wwwroot` folder or in any of its subfolders can be served to a web browser.

Defining a Dreamweaver site

After configuring your system to develop web applications, define a Dreamweaver site to manage your files.

Note: Macromedia HomeSite and ColdFusion Studio users can think of a Dreamweaver site as a HomeSite or Studio project.

Before you start, make sure you meet the following requirements:

- You have access to a web server. The web server can be running on your local computer, on a remote computer such as a development server, or on a server maintained by a web hosting company. See “Setting up a web server” on page 129.
- An application server is installed and running on the system running your web server. See “Setting up an application server” on page 129.
- You created a root folder for your web application on the system running your web server. For more information, see “Creating a root folder for the application” on page 132.

Defining a Dreamweaver site for your web application consists of three steps:

- 1 Define a folder located on your hard disk as a Dreamweaver local folder to store working copies of your site files (see “Defining a local folder” on page 133).
- 2 Define a folder located on the computer running your web server as a Dreamweaver remote folder (see “Defining a remote folder” on page 133).
- 3 Specify where Dreamweaver should send dynamic pages to be processed while you work (see “Specifying where dynamic pages can be processed” on page 134).

After the Dreamweaver site is defined, you can start building your web application.

Defining a local folder

You can define a Dreamweaver local folder for each new web application you create. The local folder is the folder you use to store working copies of site files on your hard disk. If you don't define a local folder, Dreamweaver will not work properly.

Defining a local folder also gives you the ability to manage your files and to transfer files to and from your web server at the click of a button.

To define a Dreamweaver local folder:

- 1 Create a folder on your local disk to store working copies of your files.
You may want to create subfolders to store image files and other assets.
- 2 In Dreamweaver, choose Site > New Site.
The Site Definition dialog box appears.
- 3 If the wizard is showing, click Advanced then select Local Info from the Category list (it should be the default).
- 4 In the Site Name box, enter a descriptive name for your Dreamweaver site.
- 5 In the Local Root Folder box, specify the folder you created in Step 1.
You can enter a path or click the folder icon to browse to and select the folder.
- 6 If you want, complete the other options in the Local Info category (they are not required to make the site work).

For more information on these options, click the Help button in the dialog box.

Leave the Site Definition dialog box open. You must specify a remote folder next.

Defining a remote folder

After defining a local folder, you can define a remote folder for your Dreamweaver site. The remote folder is the folder you created for your web application on the web server (see "Creating a root folder for the application" on page 132).

You don't need to define a remote folder if the folder you defined in "Defining a local folder" on page 133 can double as the root folder for your web application. (This implies that the web server is running on your local computer.)

To define a Dreamweaver remote folder:

- 1 If the Site Definition dialog box is not open, open it by choosing Site > Edit Sites, selecting your site, and clicking Edit.
The Site Definition dialog box appears.
- 2 If the wizard is showing, click Advanced then select Remote Info from the Category list.
The Remote Info dialog box appears.
- 3 In the Access pop-up menu, choose one of the following options: Local/Network, FTP, or RDS.
Your choice tells Dreamweaver how you want to transfer files between your local folder and remote folder.

Note: To use RDS, the remote folder must be on a computer running ColdFusion.

You can also send your files to a SourceSafe application by choosing SourceSafe Database. (SourceSafe is used by developers for file version control.) If you choose this option, you need to define a separate folder. For instructions, see “Specifying where dynamic pages can be processed” on page 134.

- 4 After choosing an access method, set the access options as appropriate.

For more information on these options, click the Help button in the dialog box.

Leave the Site Definition dialog box open. You need to define a folder to process dynamic pages next.

Specifying where dynamic pages can be processed

After defining the remote folder in Dreamweaver, specify a folder where dynamic pages can be processed. Dreamweaver uses this folder to generate dynamic content and connect to databases while you work.

Typically, you specify the root folder you created on the web server (see “Creating a root folder for the application” on page 132) because a web server and application server work together.

Note: The root folder can be local or remote, depending on where your web server is running.

To specify where Dreamweaver can get dynamic pages processed:

- 1 If the Site Definition dialog box is not open, open it by choosing Site > Edit Sites, selecting your site, and clicking Edit.

The Site Definition dialog box appears.

- 2 If the wizard is showing, click Advanced then select Testing Server from the Category list.

The Testing Server dialog box appears. Dreamweaver needs the services of a testing server to generate and display dynamic content while you work. The testing server can be your local computer, a development server, a staging server, or a production server. As long as it can process the kind of dynamic pages you plan to develop, the choice doesn't matter.

- 3 Complete the dialog box and click OK.

For more information, click the Help button in the dialog box.

About the URL prefix

You must specify a URL prefix so Dreamweaver MX can use the services of a testing server to display data and connect to databases while you work. Dreamweaver uses the testing server to generate the dynamic content displayed in the Live Data window and in your browser when you use the Preview in Browser command.

Dreamweaver also uses the testing server to establish connections to a database at design time. Dreamweaver uses the design-time connection to provide you with useful information about the database, such as the names of the tables in your database and the names of the columns in your tables.

A URL prefix comprises the domain name and any of your website's home directory's subdirectories or virtual directories.

Note: This section uses the terminology used in Microsoft IIS. The terminology may vary from server to server, but the same concepts apply to most web servers.

The home directory is the folder on the server mapped to your site's domain name. Suppose the folder you want to use to process dynamic pages is `c:\sites\company\`, and this folder is your home directory (that is, this folder is mapped to your site's domain name—for example, `www.mystartup.com`). In that case, the URL prefix is `http://www.mystartup.com/`.

If the folder you want to use to process dynamic pages is a subfolder of your home directory, simply add the subfolder to the URL. Suppose your home directory is `c:\sites\company\`, your site's domain name is `www.mystartup.com`, and the folder you want to use to process dynamic pages is `c:\sites\company\inventory`. Enter the following URL prefix:

```
http://www.mystartup.com/inventory/
```

If the folder you want to use to process dynamic pages is not your home directory or any of its subdirectories, you must create a virtual directory.

A virtual directory is a folder not physically contained in the home directory of the server even though it appears to be in the URL. To create a virtual directory, specify an alias for the folder's path in the URL. Suppose your home directory is `c:\sites\company`, your processing folder is `d:\apps\inventory`, and you define an alias for this folder called `warehouse`. Enter the following URL prefix:

```
http://www.mystartup.com/warehouse/
```

Localhost is a term that refers to the home directory in your URLs when the client (usually a browser, but in this case Dreamweaver) runs on the same system as your web server. Suppose Dreamweaver is running on the same system as the web server, your home directory is `c:\sites\company`, and you defined a virtual directory called `warehouse` to refer to the folder you want to use to process dynamic pages. Enter the following URL prefix:

```
http://localhost/warehouse/
```

To determine your domain name and home directory in PWS and IIS 4.0, click the Main icon in Personal Web Manager and note the home page specified in the Publishing area.

Connecting to a database

If you want to use a database with your web application, you must first connect to it. For a general discussion of database connections, see “Understanding database connections” on page 657.

Dreamweaver MX handles database connections differently depending on your choice of server technology. See the following chapters:

- Chapter 7, “Database Connections for ColdFusion Developers,” on page 139
- Chapter 8, “Database Connections for ASP.NET Developers,” on page 145
- Chapter 9, “Database Connections for ASP Developers,” on page 151
- Chapter 10, “Database Connections for JSP Developers,” on page 165
- Chapter 11, “Database Connections for PHP Developers,” on page 171

Troubleshooting application server errors

This section describes some common application server errors and ways to troubleshoot them.

For more information on Macromedia application servers, visit the ColdFusion and JRun Support Centers on the Macromedia website at <http://www.macromedia.com/support/>.

For more information on other application servers, see the server's documentation, visit the vendor's website, or contact the vendor's technical support department.

This section covers the following server errors:

- ColdFusion - 405 Method Not Allowed
- JRun - Syntax Error: Identifier expected instead of this token
- JRun - Error 2140
- WebSphere - Error 403 (forbidden by rule)
- WebSphere - Error 404 (File Not Found)

ColdFusion - 405 Method Not Allowed

This error occurs with ColdFusion 4 or 5 when you verify your installation. It might be caused by having the FrontPage version of PWS. Make sure you have the full version of PWS, not the FrontPage version.

If you have the correct version of PWS, check the virtual directory name for `cfdocs` and `cfide`. In PWS, click the Advanced icon, choose the virtual directory from the list, and click Edit Properties. In the Alias box, remove any commas from the virtual directory name.

You might also want to store your database files outside the web server's root folder. Web servers can sometimes cache files and place exclusive locks on them, causing write-permission problems. Storing your databases outside the web server's root folder also provides an additional measure of security.

JRun - Syntax Error: Identifier expected instead of this token

This error occurs when trying to process a JSP page named `default.jsp`.

The word "default" is a reserved keyword in JRun. To fix the problem, rename the page `index.jsp` or `home.jsp`.

JRun - Error 2140

This error may occur for several reasons:

- The license key was not entered correctly. Cut and paste the license key from your purchase confirmation e-mail to avoid any typing errors. Also make sure there is no space following the license key.
- The `jsm-default` was not properly installed as an NT service.

For more information, see article 12015 at the JRun support center at http://www.macromedia.com/go/error_2140.

WebSphere - Error 403 (forbidden by rule)

This error occurs when a servlet or JSP is invoked. Here are possible causes and solutions:

- You used an incorrect filename when invoking the JSP file. Verify the filename of the JSP file. Correct any errors in the spelling, capitalization, or extension of the filename. Make sure that your JSP filename ends with .jsp.
- If you're not using the default port number for your IBM HTTP Server for AS/400 instance (port 80), then you didn't add the host:port as an alias for your virtual host. To view your virtual hosts aliases, click the Advanced properties tab of your virtual host in the Administrative Console.
- There is no JSP-enabling servlet for the web application in which the JSP file resides. Verify that the JSP-enabling servlet has been added to the web application in which the JSP resides. Create the JSP-enabling servlet if necessary.
- If your servlet was created within a web application, your servlet needs to have a uniform resource indicator (URI) path. Verify that the URI path is correct.
- The Pass or Redirect directives in the active HTTP server configuration are stopping the JSP or servlet from invoking. Verify that the Pass and Redirect directives in the active IBM HTTP Server for AS/400 configuration are correct.

For more information, see the FAQ on the IBM website at <http://www.ibm.com/support/techdocs/atmstr.nsf/PubAllNum/FQ101338>.

For other problems with WebSphere, see the IBM WebSphere Support Center at <http://www.ibm.com/software/webservers/appserv/support.html>.

WebSphere - Error 404 (File Not Found)

This error occurs when invoking a servlet or JSP. Here are possible causes and solutions:

- Verify that the JSP or servlet URL is not misspelled.
- The application server process does not have sufficient authority to access the folder containing the JSP-generated Java code and class file. Verify that QEJBSVR has authority to the /QIBM/UserData/WebASAdv/default/temp or /QIBM/UserData/WebASAdv/<username>/temp folder structure.
- This error may be occurring because a class used by your servlet cannot be found. Verify all the classes that your servlet requires are in the web application classpath variable.
- This error might be caused by a misplaced Pass directive in the HTTP configuration file. If you have the (PASS /*) Pass directive before the Service directives, the Service directive will never be referenced. To correct this problem, either move the Pass directive to the end of your HTTP configuration entries or make it more specific (for example, PASS /*.html). For more information about HTTP configuration directives, see "Editing the HTTP configuration file to add server directives" in the WebSphere documentation.

For more information, see the IBM WebSphere Support Center at <http://www.ibm.com/software/webservers/appserv/support.html>.

CHAPTER 7

Database Connections for ColdFusion Developers

This chapter describes how to connect to databases when developing Macromedia ColdFusion applications with Macromedia Dreamweaver MX. For ColdFusion applications, you connect in Dreamweaver by choosing a ColdFusion data source defined in ColdFusion Administrator, the server's management console.

The chapter assumes you have set up a ColdFusion web application (see “Setting Up a Web Application” on page 127). It also assumes a database is set up on your local computer or on a system to which you have network or FTP access.

This chapter contains the following topics:

- “Connecting to a database” on page 139
- “Editing or deleting a database connection” on page 140
- “Connecting using UltraDev 4 connectivity” on page 141

To learn more about databases and database connections, see “Beginner's Guide to Databases” on page 651.

Connecting to a database

When developing a ColdFusion web application in Dreamweaver, you connect to a database by choosing a ColdFusion data source defined in ColdFusion Administrator, the server's management console.

Note: If you're running ColdFusion 5 on a Windows computer, you can also set up a data source name (DSN) on the computer. System DSNs are automatically treated as data sources by ColdFusion 4 or 5.

Before you can connect to a database, make sure Dreamweaver knows where to find the ColdFusion data sources. To retrieve the ColdFusion data sources at design time, Dreamweaver places scripts in a folder on the computer running ColdFusion. You must specify this folder in the Testing Server category of the Site Definition dialog box. For more information, see “Specifying where dynamic pages can be processed” on page 134.

Next, you must create a ColdFusion data source in ColdFusion Administrator (if one doesn't already exist). For more information, see “Creating a ColdFusion data source” on page 140.

After creating a ColdFusion data source, you can use it in Dreamweaver to connect to the database. For more information, see “Connecting to the database in Dreamweaver” on page 140.

The section assumes you're using Dreamweaver MX connectivity, not Dreamweaver UltraDev 4 connectivity. Database connections are handled differently in UltraDev 4. For more information, see "Connecting using UltraDev 4 connectivity" on page 141.

Creating a ColdFusion data source

Before you can connect to a database, you must create a ColdFusion data source in ColdFusion Administrator, the server's management console.

Note: If you're running ColdFusion 5 on a Windows computer, you can also set up a DSN on the computer. System DSNs are automatically treated as data sources by ColdFusion 5. For more information, see "Setting Up a DSN in Windows" on page 671.

To create a ColdFusion data source:

- 1 In the Databases panel (Window > Databases) in Dreamweaver, click the Modify Data Sources icon (the second icon from the right on the panel toolbar).

ColdFusion Administrator opens in a browser.

- 2 Log in to ColdFusion Administrator and create the data source.

For instructions, see the ColdFusion documentation.

You must provide certain parameter values to create the ColdFusion data source. For the parameter values specific to your database driver, see the driver vendor's documentation or consult your system administrator.

Once you create a ColdFusion data source, you can use it in Dreamweaver.

Connecting to the database in Dreamweaver

After creating a ColdFusion data source in ColdFusion Administrator, you can use it to connect to the database in Dreamweaver.

Open any ColdFusion page in Dreamweaver, then open the Databases panel (Window > Databases). Your ColdFusion data sources appear in the panel.

If the data sources don't appear, make sure Dreamweaver knows where to find the ColdFusion data sources. In the Testing Server category of the Site Definition dialog box, specify the site's root folder on the computer running ColdFusion. For more information, see "Specifying where dynamic pages can be processed" on page 134.

Editing or deleting a database connection

You edit or delete ColdFusion data sources in ColdFusion Administrator. In the Databases panel (Window > Databases) in Dreamweaver, click the Modify Data Sources icon (the second icon from the right on the panel toolbar). ColdFusion Administrator opens in a browser. Log in and modify the data source. For instructions, see the ColdFusion documentation.

To avoid getting errors after deleting or renaming a ColdFusion data source, update every recordset that uses the old data source in Dreamweaver by double-clicking the name of the recordset in the Bindings panel and choosing a new data source.

Connecting using UltraDev 4 connectivity

This section describes how to connect to a database if you built your ColdFusion application using Dreamweaver UltraDev 4 server behaviors, or if you're building it in Dreamweaver MX using UltraDev 4 server behaviors. For more information, see "A note for Dreamweaver UltraDev 4 users" on page 127.

The section uses the term "UD4 ColdFusion" to refer to UltraDev 4 connectivity.

This section covers the following topics:

- "Connection requirements" on page 141
- "Creating a regular UD4 ColdFusion connection" on page 141
- "Creating an advanced UD4 ColdFusion connection (Macintosh users)" on page 142
- "Editing or deleting a UD4 ColdFusion connection" on page 142

Connection requirements

Before creating a UD4 ColdFusion database connection, you must have the following:

- A Dreamweaver site that specifies that the site contains only UltraDev 4 files. For more information, see "Specifying where dynamic pages can be processed" on page 134
- A database set up on your local computer or on a system to which you have network or FTP access
- An appropriate database driver installed on the computer running ColdFusion
- A ColdFusion data source defined for the database. For more information, see "Creating a ColdFusion data source" on page 140

Creating a regular UD4 ColdFusion connection

This section describes how to create a regular database connection if you're using UltraDev 4 connectivity.

UltraDev 4's ColdFusion connectivity on the Macintosh does not support stored procedures in databases other than SQL Server 7.0. If you're a Macintosh user and want to use a stored procedure in a database other than SQL Server 7.0, create an advanced ColdFusion connection using JDBC to connect to the database at design time. For procedures, see "Creating an advanced UD4 ColdFusion connection (Macintosh users)" on page 142.

To create a regular UD4 ColdFusion connection:

- 1 Open a ColdFusion page in Dreamweaver, then open the Databases panel (Window > Databases).
- 2 Click the plus (+) button on the panel and select Data Source Name from the pop-up menu.
- 3 If this is the first connection you create for the site, Dreamweaver prompts you for your ColdFusion RDS (Remote Development Services) user name and password.

After you log in, Dreamweaver connects to the server, retrieves the ColdFusion data sources, and displays the Data Source Name dialog box.

- 4 Complete the dialog box and click OK.

For instructions, click the Help button in the dialog box.

The new connection appears in the Databases panel.

Creating an advanced UD4 ColdFusion connection (Macintosh users)

UltraDev ColdFusion connectivity on the Macintosh does not support stored procedures in databases other than SQL Server 7.0. If you're a Macintosh user and want to use a stored procedure in a database other than SQL Server 7.0, you must create an advanced ColdFusion connection.

To create an advanced ColdFusion database connection in Dreamweaver:

- 1 Open a ColdFusion page in Dreamweaver, then open the Databases panel (Window > Databases).
- 2 Click the plus (+) button on the panel and select Data Source Name - Advanced from the pop-up menu.
- 3 If this is the first connection you create for the site, Dreamweaver prompts you for your ColdFusion RDS user name and password.

After you log in, Dreamweaver connects to the server, retrieves the ColdFusion DSNs, and displays the Data Source Name - Advanced dialog box.

- 4 Complete the dialog box and click OK.

For instructions, click the Help button in the dialog box.

The new connection appears in the Databases panel.

Editing or deleting a UD4 ColdFusion connection

When you create a database connection, Dreamweaver stores the connection information in a file in the Connections subfolder in the site's local root folder. Dreamweaver does not actually create a database connection for your ColdFusion application until you define a recordset for a page in the application (see "Defining a recordset" on page 502). At that point, Dreamweaver writes code in the file to establish the connection, and inserts an include directive in your page. At runtime, the server inserts the connection code in your document.

To update a connection:

- 1 Open a ColdFusion page in Dreamweaver MX, then open the Databases panel (Window > Databases).

A list of connections appears in the panel.

- 2 Right-click (Windows) or Control-click (Macintosh) the connection and choose Edit Connection from the pop-up menu.

The dialog box you used to create the connection appears.

- 3 Make the changes and click OK.

Dreamweaver automatically updates the include file, which updates all the pages in the site that use the connection.

To delete a connection:

- 1 Open a ColdFusion page in Dreamweaver, then open the Databases panel (Window > Databases).

A list of connections appears in the panel.

- 2 Right-click (Windows) or Control-click (Macintosh) the connection and choose Delete Connection from the pop-up menu.

The dialog box you used to create the connection appears.

3 Confirm that you want to delete the connection.

To avoid getting errors after deleting a connection, update every recordset that uses the old connection by double-clicking the name of the recordset in the Bindings panel and choosing a new connection.

CHAPTER 8

Database Connections for ASP.NET Developers

To use a database with an ASP.NET application, you need to create a database connection in Macromedia Dreamweaver MX. This chapter describes how to create the database connection.

Note: If you're developing Microsoft Active Server Pages (ASP) applications, see "Database Connections for ASP Developers" on page 151.

The chapter assumes you have set up an ASP.NET application (see "Setting Up a Web Application" on page 127). It also assumes a database is set up on your local computer or on a system to which you have network or FTP access.

This chapter contains the following sections:

- "Connecting to a database" on page 145
- "Editing or deleting a database connection" on page 148

To connect to the sample database installed with Dreamweaver, see Dreamweaver Help (Help > Using Dreamweaver). To learn more about databases and database connections, see "Beginner's Guide to Databases" on page 651.

Connecting to a database

This section describes how to connect to a database when developing an ASP.NET application in Dreamweaver MX.

Before you can connect to a database, you must obtain an OLE DB provider for your database. If you want to connect to a Microsoft SQL Server database, you can use the Managed Data Provider for SQL Server that is supplied by the .NET Framework. For more information, see "Obtaining an OLE DB provider for your database" on page 146.

Once the database provider is installed, you can use it to connect to the database. For instructions, see the following sections:

- "Creating a database connection in Dreamweaver" on page 146
- "Creating a connection using Data Link Properties" on page 147

Obtaining an OLE DB provider for your database

An ASP.NET application must connect to a database through an OLE DB provider. The provider acts as an interpreter that lets an ASP.NET application communicate with a database. For more information on OLE DB and the role of database providers, see “Interfacing with the database” on page 658.

If you want to connect to a Microsoft SQL Server database, you can use the Managed Data Provider for SQL Server that comes with the .NET Framework. This provider, which is optimized for SQL Server and is very fast, is installed when you install the .NET Framework.

If you want to connect to a database other than SQL Server, make sure an OLE DB provider for your database is installed on the computer running the .NET Framework. You automatically installed an OLE DB provider for Microsoft Access when you downloaded and installed the Microsoft Data Access Components (MDAC) 2.7 package on the computer.

Note: Installing MDAC 2.7 is highly recommended when you install the .NET Framework. For more information, see “Installing an ASP.NET application server” on page 130.

You can download OLE DB providers for Oracle9i and Oracle8i databases from the Oracle website at http://otn.oracle.com/software/tech/windows/ole_db/content.html (registration is required). You can also purchase OLE DB providers from third-party vendors.

Once you have a provider for your database, you can use it to create database connections in Dreamweaver.

Creating a database connection in Dreamweaver

After obtaining an OLE DB provider for your database, you can use it to create a database connection in Dreamweaver.

Another option is to use the Microsoft Data Link Properties dialog box to help you create the connection. For instructions, see “Creating a connection using Data Link Properties” on page 147.

To create a database connection for ASP.NET:

- 1 Open an ASP.NET page in Dreamweaver, then open the Databases panel (Window > Databases). The panel displays the connections defined for the site.
- 2 Click the plus (+) button on the panel and select OLE DB Connection or SQL Server Connection from the pop-up menu.

Note: Select SQL Server Connection only if you want to connect to a Microsoft SQL Server database.

The OLE DB Connection or SQL Server Connection dialog box appears.

- 3 Complete the dialog box and click OK.

For more information, click the Help button in the dialog box.

The new connection appears in the Databases panel.

Creating a connection using Data Link Properties

After obtaining an OLE DB provider for your database (see “Obtaining an OLE DB provider for your database” on page 146), you can create a database connection by using the Data Link Properties dialog box in Windows.

Important: You can use this method only if the OLE DB provider you want to use is installed on the same Windows computer as Dreamweaver.

To create a database connection using Data Link Properties:

- 1 Open an ASP.NET page in Dreamweaver, then open the Databases panel (Window > Databases).
The panel displays the connections defined for the site.
- 2 Click the plus (+) button on the panel and select OLE DB Connection from the pop-up menu.
The OLE DB Connection dialog box appears.
- 3 Click the Build button.
The Data Link Properties dialog box appears. This Windows dialog box displays the OLE DB providers currently on the Windows computer running Dreamweaver.
- 4 Complete the Data Link Properties dialog box and click OK.
Dreamweaver inserts a connection string in the OLE DB Connection dialog box.
- 5 Click Test.
Dreamweaver attempts to connect to the database. If the connection fails, double-check the connection string. If the connection still fails, check the settings for the folder Dreamweaver uses to process dynamic pages (see “Specifying where dynamic pages can be processed” on page 134).
- 6 Click OK.
The new connection appears in the Databases panel.

Sample OLE DB connection parameters for ASP.NET

An OLE DB connection string combines all the information your ASP.NET application needs to connect to a database. Dreamweaver inserts this string in your page’s server-side scripts for later processing by your application server.

Dreamweaver provides you with string templates to create OLE DB connection strings for ASP.NET applications (see “Creating a database connection in Dreamweaver” on page 146). To create a connection string, you replace placeholders in the template with the requested parameter values. This section gives sample parameters for Microsoft Access and SQL Server databases.

Note: For the parameter values specific to other databases, see the database vendor’s documentation or consult your system administrator.

Case 1: You have the .NET Framework on your local computer and you want to connect to a Microsoft Access database called `sdSchool.mdb` located in the following folder on your hard disk: `c:\inetpub\wwwroot\SkyDiveSchool\data\sdSchool.mdb`. Here are the parameters to create this connection string:

Provider=Microsoft.Jet.OLEDB.4.0;

Data Source=c:\inetpub\wwwroot\SkyDiveSchool\data\sdSchool.mdb;

Case 2: You use the .NET Framework on a remote development server and you want to connect to a Microsoft Access database called `mtnSchool.mdb` located on the server in the following folder: `d:\users\tara\projects\MtnDrivingSchool\data\mtnSchool.mdb`. Here are the parameters to create the connection string:

```
Provider=Microsoft.Jet.OLEDB.4.0;
```

```
Data Source=d:\users\tara\projects\MtnDrivingSchool\data\mtnSchool.mdb;
```

Case 3: Suppose you use the .NET Framework on a networked development server called Savant and you want to connect to a Microsoft SQL Server database called `pubs` on the server. Your SQL Server user name is “sa” and there is no password. If you use the Managed Data Provider for SQL Server (that is, if you chose SQL Connection in the Databases panel), here are the parameters to create the connection string:

```
Data Source=Savant;
```

```
Initial Catalog=pubs;
```

```
User ID=sa;
```

```
Password=;
```

Editing or deleting a database connection

When you create a database connection, Dreamweaver stores the connection information in a file in the Connections subfolder in the site’s local root folder. Dreamweaver does not create a database connection for your ASP.NET application until you define a DataSet for a page in the application (see “Defining a recordset” on page 502). At that point, Dreamweaver writes code in the file to establish the connection, and inserts an include directive in your page. At runtime, the server inserts the connection code in your document.

To update a connection:

- 1 Open an ASP.NET page in Dreamweaver, then open the Databases panel (Window > Databases).

A list of connections appears in the panel.

- 2 Right-click (Windows) or Control-click (Macintosh) the connection and choose Edit Connection from the pop-up menu.

The dialog box you used to create the connection appears.

- 3 Make the changes and click OK.

Dreamweaver automatically updates the include file, which automatically updates all the pages in the site that use the connection.

If you rename a connection, update every recordset that uses the old connection name by double-clicking the recordset in the Bindings panel and choosing the new connection name in the DataSet dialog box.

To delete a connection:

- 1** Open an ASP.NET page in Dreamweaver, then open the Databases panel (Window > Databases).
A list of connections appears in the panel.
- 2** Right-click (Windows) or Control-click (Macintosh) the connection and choose Delete Connection from the pop-up menu.
The dialog box you used to create the connection appears.
- 3** Confirm that you want to delete the connection.

Note: To avoid getting errors after deleting a connection, update every recordset that uses the old connection by double-clicking the name of the recordset in the Bindings panel and choosing a new connection in the DataSet dialog box.

CHAPTER 9

Database Connections for ASP Developers

To use a database with a Microsoft Active Server Pages (ASP) application, you need to create a database connection in Macromedia Dreamweaver MX. This chapter describes how to create the database connection.

Note: If you're developing ASP.NET applications, see "Database Connections for ASP.NET Developers" on page 145.

The chapter assumes you have set up an ASP application (see "Setting Up a Web Application" on page 127). It also assumes a database is set up on your local computer or on a system to which you have network or FTP access.

This chapter contains the following sections:

- "Understanding ASP database connections" on page 151
- "Creating a DSN connection" on page 152
- "Creating a DSN-less connection" on page 155
- "Connecting to a database on an ISP" on page 158
- "Editing or deleting a database connection" on page 160

To connect to the sample database installed with Dreamweaver, see Dreamweaver Help (Help > Using Dreamweaver). To learn more about databases and database connections, see Appendix A, "Beginner's Guide to Databases," on page 651.

Understanding ASP database connections

An ASP application must connect to a database through an open database connectivity (ODBC) driver or an object linking and embedding database (OLE DB) provider. The driver or provider acts as an interpreter that lets the web application communicate with the database. For more information on the role of database drivers, see "Interfacing with the database" on page 658. The following table shows some drivers you can use with Microsoft Access, Microsoft SQL Server, and Oracle databases:

Database	Database driver
Microsoft Access	Microsoft Access Driver (ODBC)
Microsoft SQL Server	Microsoft SQL Server Driver (ODBC) Microsoft SQL Server Provider (OLE DB)
Oracle	Microsoft Oracle Driver (ODBC) Oracle Provider for OLE DB

You can use a data source name (DSN) or a connection string to connect to the database.

A DSN is a one-word identifier, such as Acme, that points to the database and contains all the information needed to connect to it. You define a DSN in Windows. You can use a DSN if you're connecting through an ODBC driver installed on a Windows system. For detailed instructions, see "Creating a DSN connection" on page 152.

A connection string is a hand-coded expression that identifies the database and lists the information needed to connect to it. Example:

```
Driver={SQL Server};Server=Socrates;Database=AcmeMktg;  
UID=wiley;PWD=roadrunner
```

You must use a connection string if you're connecting through one of the following:

- An OLE DB provider
- An ODBC driver not installed on a Windows system

For detailed instructions, see the following sections:

- "Creating a DSN-less connection" on page 155
- "Connecting to a database on an ISP" on page 158

Note: You can also use a connection string if you're connecting through an ODBC driver installed on a Windows system, but using a DSN is easier.

Creating a DSN connection

You can use a DSN to create an ODBC connection between your web application and your database. A DSN is a name containing all the parameters needed to connect to a specific database using an ODBC driver. For more information, see "Understanding DSNs" on page 671.

Note: Because you can only specify an ODBC driver in a DSN, you must use a connection string if you want to use an OLE DB provider. For more information, see "Creating an OLE DB connection" on page 157.

You can define the DSN on a local or remote Windows computer. The following topics describes methods for creating both types of connections:

- "Creating a connection using a local DSN" on page 152
- "Creating a connection using a remote DSN" on page 154

Creating a connection using a local DSN

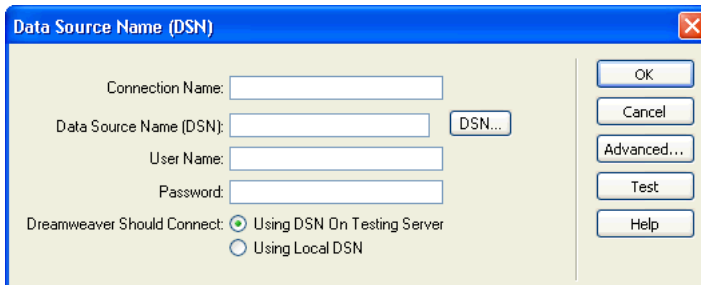
You can use a locally defined DSN to create a database connection in Dreamweaver. If you want to use a local DSN, the DSN must be defined on the Windows computer running Dreamweaver.

To create a database connection with a locally defined DSN:

- 1 Define a DSN on the Windows computer running Dreamweaver.
For instructions, see "Creating a DSN" on page 671.
- 2 Open an ASP page in Dreamweaver, then open the Databases panel (Window > Databases).
Dreamweaver displays all the connections defined for the site.

- 3 Click the plus (+) button on the panel and select Data Source Name (DSN) from the pop-up menu.

The Data Source Name (DSN) dialog box appears.



- 4 Enter a name for the new connection.

Note: Do not use any spaces or special characters in the name.

- 5 Select the Using Local DSN option at the bottom of the dialog box.
- 6 Select the DSN you want to use from the Data Source Name (DSN) pop-up menu.

If you want to use a local DSN but haven't defined one yet, click Define to open the Windows ODBC Data Source Administrator. For instructions, see "Creating a DSN connection" on page 152.

- 7 If required, complete the User Name and Password boxes.
- 8 If you want, restrict the number of database items Dreamweaver retrieves at design time by clicking Advanced and entering a schema or catalog name.

For more information, see "Restricting database information displayed in Dreamweaver" on page 481.

Note: You cannot create a schema or catalog in Microsoft Access.

- 9 Click Test.

Dreamweaver attempts to connect to the database. If the connection fails, double-check the DSN. If the connection still fails, check the settings for the folder Dreamweaver uses to process dynamic pages (see "Specifying where dynamic pages can be processed" on page 134).

- 10 Click OK.

The new connection appears in the Databases panel.

Creating a connection using a remote DSN

You can use a DSN defined on a remote computer to create a database connection in Dreamweaver. If you want to use a remote DSN, the DSN must be defined on the Windows computer running your application server (probably IIS).

To create a database connection with a remotely defined DSN:

- 1 Define a DSN on the remote system running your application server.

For instructions, see “Creating a DSN” on page 671.

- 2 Open an ASP page in Dreamweaver, then open the Databases panel (Window > Databases).

Dreamweaver displays all the connections defined for the site.

- 3 Click the plus (+) button on the panel and select Data Source Name (DSN) from the pop-up menu.

The Data Source Name (DSN) dialog box appears.

- 4 Enter a name for the new connection.

Note: Do not use any spaces or special characters in the name.

- 5 Select the Using DSN on Testing Server option at the bottom of the dialog box.

Macintosh users can ignore this step because all database connections use DSNs on the application server.

- 6 Enter the DSN.

You can click the DSN button to connect to the server and choose from the DSNs defined on it.

- 7 If required, complete the User Name and Password boxes.

- 8 If you want, restrict the number of database items Dreamweaver retrieves at design time by clicking Advanced and entering a schema or catalog name.

For more information, see “Restricting database information displayed in Dreamweaver” on page 481.

Note: You cannot create a schema or catalog in Microsoft Access.

- 9 Click Test.

Dreamweaver attempts to connect to the database. If the connection fails, double-check the DSN. If the connection still fails, check the settings for the folder Dreamweaver uses to process dynamic pages (see “Specifying where dynamic pages can be processed” on page 134).

- 10 Click OK.

The new connection appears in the Databases panel.

Creating a DSN-less connection

You can use a DSN-less connection to create an ODBC or OLE DB connection between your web application and your database.

You use a connection string to create the connection. A connection string combines all the information your web application needs on the server to connect to a database. Dreamweaver inserts this string in your page's server-side scripts for later processing by your application server.

Here's an example of a connection string:

```
Driver={Microsoft Access Driver (*.mdb)};  
DBQ=C:\Inetpub\wwwroot\Academy\curriculum.mdb
```

Here's a second example:

```
Driver={SQL Server};Server=Socrates;Database=MedCenter;  
UID=mwelby;PWD=realme
```

This section contains the following topics:

- “Creating a database connection with a connection string” on page 155
- “Writing a connection string” on page 156
- “Creating an OLE DB connection” on page 157

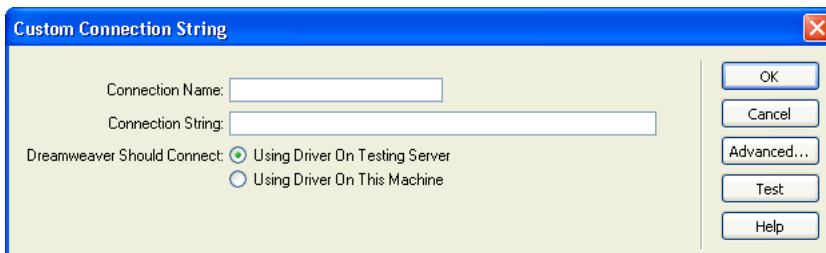
Creating a database connection with a connection string

You can use a connection string to create a database connection between your web application and your database.

To create a DSN-less connection:

- 1 Open an ASP page in Dreamweaver, then open the Databases panel (Window > Databases). Dreamweaver displays all the connections defined for that site, if any.
- 2 Click the plus (+) button on the panel and select Custom Connection String from the pop-up menu.

The Custom Connection String dialog box appears as follows.



- 3 Complete the dialog box and click OK.

For more information, do one of the following:

- Press the Help button in the dialog box.
- See “Writing a connection string” on page 156.

Writing a connection string

A connection string combines all the information your web application needs on the server to connect to a database. Dreamweaver inserts this string in your page's server-side scripts for later processing by your application server.

A connection string for Microsoft Access and SQL Server databases consists of a combination of the following parameters separated by semicolons:

Provider specifies the OLE DB provider for your database. For example, here are parameters for common OLE DB providers for Access, SQL Server, and Oracle databases, respectively:

```
Provider=Microsoft.Jet.OLEDB.4.0;...  
Provider=SQLOLEDB;...  
Provider=OraOLEDB;...
```

For the parameter value of your OLE DB provider, see your provider vendor's documentation, or consult your system administrator.

If you don't include a Provider parameter, then the default OLE DB provider for ODBC is used and you must specify an appropriate ODBC driver for your database.

Driver specifies the ODBC driver to use if you don't specify an OLE DB provider for your database.

Server specifies the server hosting the SQL Server database if your web application runs on a different server.

Database is the name of a SQL Server database.

DBQ is the path to a file-based database such as one created in Microsoft Access. The path is the one on the server hosting the database file.

UID specifies the user name.

PWD specifies the user password.

DSN is the data source name, if you use one. Depending on how you define the DSN on your server, you can omit the connection string's other parameters. For example, `DSN=Results` can be a valid connection string if you define the other parameters when you create the DSN (see "Creating a DSN" on page 671).

Connection strings for other kinds of databases may not use the parameters listed above, or will have different names or uses for the parameters. For more information, see your database vendor's documentation, or consult your system administrator.

Here's an example of a connection string that will create an ODBC connection to an Access database called `trees.mdb`:

```
Driver={Microsoft Access Driver (*.mdb)};  
DBQ=C:\Inetpub\wwwroot\Research\trees.mdb
```

Here's an example of a connection string that will create an OLE DB connection to a SQL Server database called `Mothra` located on a server called `Gojira`:

```
Provider=SQLOLEDB;Server=Gojira;Database=Mothra;UID=jsmith;  
PWD=orlando8
```

Creating an OLE DB connection

You can use an OLE DB provider to communicate with your database. Creating a direct OLE DB connection can improve the speed of your connection by eliminating the ODBC layer between your web application and the database. By using a database-specific OLE DB provider, you eliminate the ODBC middleman.

If you don't specify an OLE DB provider for your database, ASP uses the default OLE DB provider for ODBC drivers to communicate with an ODBC driver, which in turn communicates with the database.

OLE DB is available only on Windows NT, 2000, or XP.

Different OLE DB providers exist for different databases. You can obtain OLE DB providers for Microsoft Access and SQL Server by downloading and installing the Microsoft Data Access Components (MDAC) 2.5 and 2.6 packages on the Windows computer running IIS or PWS (if applicable). You can download the MDAC packages for free from the Microsoft website at <http://www.microsoft.com/data/download.htm>.

Note: Make sure you install MDAC 2.5 before installing MDAC 2.6.

You can download OLE DB providers for Oracle9i and Oracle8i databases from the Oracle website at http://otn.oracle.com/software/tech/windows/ole_db/content.html (registration is required).

In Dreamweaver MX, you create an OLE DB connection by including a `Provider` parameter in a connection string. For example, here are parameters for common OLE DB providers for Access, SQL Server, and Oracle databases, respectively:

```
Provider=Microsoft.Jet.OLEDB.4.0;...  
Provider=SQLOLEDB;...  
Provider=OraOLEDB;...
```

For the parameter value of your OLE DB provider, see your provider vendor's documentation, or consult your system administrator.

Connecting to a database on an ISP

If you're an ASP developer working with a commercial Internet service provider (ISP), you often don't know the physical path of the files you upload, including your database file or files. If your ISP doesn't define a DSN for you or is slow to do so, you must find another way to create the connections to your database files. One alternative is to create a DSN-less connection to a database file, but you can define such a connection only if you know the physical path of the database file on the ISP's server.

This section describes how you can obtain the physical path of a database file on a server by using the `MapPath` method of the ASP server object. The section covers the following topics:

- “Understanding physical and virtual paths” on page 158
- “Finding a file's physical path with the virtual path” on page 158
- “Using a virtual path to connect to a database” on page 159

Note: The techniques discussed in this chapter apply only if your database is file-based, such as a Microsoft Access database where data is stored in an `.mdb` file.

Understanding physical and virtual paths

After using Dreamweaver to upload your files to a remote server, the files reside in a folder in the server's local directory tree. For example, on a server running Microsoft IIS, the path to your home page could be as follows:

```
c:\inetpub\wwwroot\accounts\users\jsmith\index.htm
```

This path is known as the physical path to your file.

The URL to open your file, however, does not use the physical path. It uses the name of the server or domain followed by a virtual path, as in the following example:

```
www.plutoserve.com/jsmith/index.htm
```

The virtual path, /jsmith/index.htm, stands in for the physical path, c:\inetpub\wwwroot\accounts\users\jsmith\index.htm.

Finding a file's physical path with the virtual path

If you work with an ISP, you don't always know the physical path to the files you upload. ISPs typically provide you with an FTP host, possibly a host directory, and a login name and password. ISPs also specify a URL to view your pages on the Internet, such as www.plutoserve.com/jsmith/.

If you know the URL, then you can get the file's virtual path—it's the path that follows the server or domain name in a URL. Once you know the virtual path, you can get the file's physical path on the server using the MapPath method.

Among other things, the MapPath method takes the virtual path as an argument and returns the file's physical path and filename. Here's the method's syntax:

```
Server.MapPath("/virtualpath")
```

Suppose a file's virtual path is /jsmith/index.htm, then the following expression will return its physical path:

```
Server.MapPath("/jsmith/index.htm")
```

You can experiment with the MapPath method as follows.

- 1 Open an ASP page in Dreamweaver and switch to Code view (View > Code).
- 2 Enter the following expression in the page's HTML code.

```
<%Response.Write(stringvariable)%>
```

- 3 Use the MapPath method to obtain a value for the *stringvariable* argument.

Here's an example:

```
<% Response.Write(Server.MapPath("/jsmith/index.htm")) %>
```

- 4 Switch to Design view (View > Design) and turn on Live Data (View > Live Data) to view the page.

The page displays the physical path of the file on the application server. Using the example discussed in this section, the page displays the following physical path:

```
c:\inetpub\wwwroot\accounts\users\jsmith\index.htm
```

For more information on the MapPath method, consult the online documentation that comes with Microsoft IIS or PWS.

Using a virtual path to connect to a database

To write a DSN-less connection string to a database file located on a remote server, you must know the physical path to the file. For example, here is a typical DSN-less connection string for a Microsoft Access database:

```
Driver={Microsoft Access Driver (*.mdb)};  
DBQ=c:\inetpub\wwwroot\accounts\users\jsmith\data\statistics.mdb
```

If you don't know the physical path of your files on the remote server, you can get the path by using the `MapPath` method in your connection string.

To create a DSN-less connection with the `MapPath` method:

- 1 Upload the database file to the remote server.

Make a note of its virtual path—for example, `/jsmith/data/statistics.mdb`.

- 2 Open an ASP page in Dreamweaver, then open the Databases panel (Window > Databases). Dreamweaver displays all the connections defined for the site.

- 3 Click the plus (+) button on the panel and select Custom Connection String from the pop-up menu.

- 4 Enter a name for the new connection.

Note: Do not use any spaces or special characters in the name.

- 5 Enter the connection string and use the `MapPath` method to supply the DBQ parameter.

Suppose the virtual path to your Microsoft Access database is `/jsmith/data/statistics.mdb`. The connection string can be expressed as follows if you use VBScript as your scripting language:

```
"Driver={Microsoft Access Driver (*.mdb)};DBQ=" & Server.MapPath(  
"/jsmith/data/statistics.mdb")
```

The ampersand (&) is used to concatenate (combine) two strings. The first string is enclosed in quotation marks and the second is returned by the `Server.MapPath` expression. When the two strings are combined, the following string is created:

```
Driver={Microsoft Access Driver (*.mdb)}; -  
DBQ=C:\inetpub\wwwroot\accounts\users\jsmith\data\statistics.mdb
```

If you use JavaScript, the expression is identical except that you use a plus (+) sign instead of an ampersand (&) to concatenate the two strings:

```
"Driver={Microsoft Access Driver (*.mdb)};DBQ=" + Server.MapPath(  
"/jsmith/data/statistics.mdb")
```

- 6 Select the Using Driver On Testing Server option.

Macintosh users can ignore this step because all database connections use the application server.

- 7 Click Test.

Dreamweaver attempts to connect to the database. If the connection fails, double-check the connection string.

If the connection still fails, contact your ISP to make sure the database driver you specified in the connection string is installed on the remote server. Also check that the ISP has the most recent version of the driver. For example, a database created in Microsoft Access 2000 will not work with Microsoft Access Driver 3.5. You need Microsoft Access Driver 4.0 or later.

- 8 Click OK.

The new connection appears in the Databases panel.

- 9 Update the database connection of existing dynamic pages, and use the new connection with any new page you build.

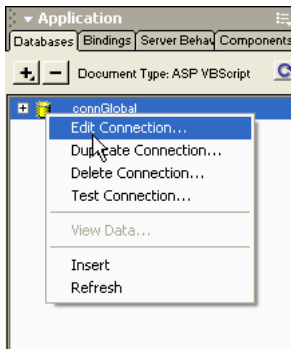
To update the connection of a dynamic page, open the page in Dreamweaver, double-click the recordset name in the Bindings panel or Server Behaviors panel, and select the connection you just created from the Connection pop-up menu.

Editing or deleting a database connection

When you create a database connection, Dreamweaver stores the connection information in a file in the Connections subfolder in the site's local root folder. Dreamweaver does not actually create a database connection for your ASP application until you define a recordset for a page in the application (see “Defining a recordset” on page 502). At that point, Dreamweaver writes code in the file to establish the connection, and inserts an include directive in your page. At runtime, the server inserts the connection code in your document.

To update a connection:

- 1 Open an ASP page in Dreamweaver, then open the Databases panel (Window > Databases).
A list of connections appears in the panel.
- 2 Right-click (Windows) or Control-click (Macintosh) the connection and choose Edit Connection from the pop-up menu, as follows.



The dialog box you used to create the connection appears.

- 3 Make the changes and click OK.

Dreamweaver automatically updates the include file, which updates all the pages in the site that use the connection.

To delete a connection:

- 1 Open an ASP page in Dreamweaver, then open the Databases panel (Window > Databases).

A list of connections appears in the panel.

- 2 Right-click (Windows) or Control-click (Macintosh) the connection and choose Delete Connection from the pop-up menu.

The dialog box you used to create the connection appears.

- 3 Confirm that you want to delete the connection.

Note: To avoid getting errors after deleting a connection, update every recordset that uses the old connection by double-clicking the name of the recordset in the Bindings panel and choosing a new connection.

CHAPTER 10

Database Connections for JSP Developers

To use a database with a JavaServer Pages (JSP) application, you need to create a database connection in Macromedia Dreamweaver MX. This chapter describes how to create the connection.

The chapter assumes you have set up a JSP application (see “Setting Up a Web Application” on page 127). It also assumes you have a database set up on your local computer or on a system to which you have network or FTP access.

This chapter contains the following sections:

- “Understanding JSP connections” on page 165
- “Connecting to a database” on page 165
- “Connecting through an ODBC driver” on page 168
- “Editing or deleting a database connection” on page 170

To connect to the sample database installed with Dreamweaver, see Dreamweaver Help (Help > Using Dreamweaver). To learn more about databases and database connections, see “Beginner’s Guide to Databases” on page 651.

Understanding JSP connections

A JSP application must connect to a database through a JDBC driver. The driver acts as an interpreter that lets a JSP application communicate with a database. For more information on JDBC and the role of database drivers, see “Interfacing with the database” on page 658.

You must specify certain parameter values to connect through your JDBC driver. For more information, see “About JDBC connection parameters” on page 167. For the parameter values specific to your driver, see the driver vendor’s documentation or consult your system administrator.

You can also use an ODBC driver (and so a Windows DSN) if you have a JDBC-ODBC Bridge driver. For more information, see “Connecting through an ODBC driver” on page 168.

Connecting to a database

This section describes how to connect to a database when developing a JSP application in Dreamweaver MX.

The section assumes a JSP application server is running on a local or remote computer. (For most Macintosh users and many development teams, the JSP application server runs on a separate development server.) For more information, see “Setting Up a Web Application” on page 127.

Before you can connect to a database, you must obtain a JDBC driver for your database.

After you install the database driver, you can connect to the database through it. For instructions, see “Creating a database connection for JSP” on page 166.

Obtaining a JDBC driver for your database

Make sure you have a JDBC driver for your database before you try to create a database connection. Some common JDBC drivers include the Oracle Thin JDBC driver, the Oracle Java Driver, and the i-net JDBC drivers for Microsoft SQL Server.

Database system vendors such as Oracle often include drivers with their systems. You can also purchase drivers from third-party vendors. For example, you can obtain a JDBC driver for Microsoft SQL Server from i-net software at http://www.inetsoftware.de/English/Produkte/JDBC_Overview/default.htm.

Sun also provides a searchable database of JDBC drivers and their vendors on its website at <http://industry.java.sun.com/products/jdbc/drivers>.

Once you have a JDBC driver for your database, you can create a database connection.

Creating a database connection for JSP

After you install an appropriate JDBC driver for your database on the computer running your application server, you can create a database connection in Dreamweaver.

To create a database connection for JSP:

- 1 Open a JSP page in Dreamweaver, then open the Databases panel (Window > Databases).
Dreamweaver displays the connections defined for the site.
- 2 Click the plus (+) button and choose your driver from the pop-up menu.
If your driver is not listed, choose Custom JDBC Connection.
A connection dialog box appears.
- 3 Enter the connection parameters in the connection dialog box.
For more information, see “About JDBC connection parameters” on page 167.
- 4 Specify the location of the JDBC driver you want to use.
 - If your JDBC driver is installed on the same computer as Dreamweaver, select the Using Driver On This Machine option.
 - If your JDBC driver is not installed on the same computer as Dreamweaver, select the Using Driver On Testing Server option.
Macintosh users can ignore this step because all database connections use the application server.
- 5 Click Test.
Dreamweaver attempts to connect to the database. If the connection fails, double-check the the connection parameters. If the connection still fails, check the settings for the folder Dreamweaver uses to process dynamic pages (see “Specifying where dynamic pages can be processed” on page 134).
- 6 Click OK.

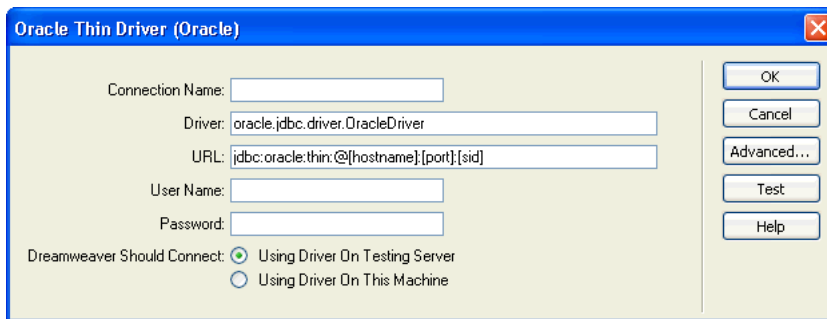
The new connection appears in the Databases panel.

About JDBC connection parameters

JDBC connections usually consist of four parameters: the driver, user name, password, and URL (which specifies the location of the database). Generally, the values of the driver parameter and the URL parameter depend on the driver.

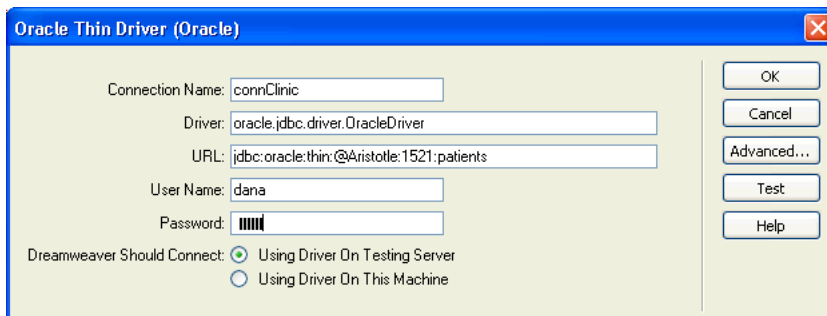
This section demonstrate how to define connection parameters in Dreamweaver using the Oracle Thin JDBC driver as an example. For the connection parameters of other drivers, consult the driver vendor's documentation.

The Oracle Thin JDBC driver supports Oracle databases. If you want to use this driver to connect to your Oracle database, click the plus (+) button on the Databases panel and choose the Oracle Thin Driver (Oracle) driver from the pop-up menu. The following, partially-complete dialog box appears:



Enter a connection name and replace the placeholders (in square brackets) with valid connection parameters. For the `[hostname]` placeholder, enter the IP address or the name assigned to the database server by the system administrator. For the `[sid]` placeholder, enter the database system identifier. If you have more than one Oracle database running on the same system, you use the SID to tell them apart.

For example, if your server is called Aristotle, the database port is 1521, and you defined a database SID called `patients` on that server, you would enter the following parameter values in Dreamweaver:



Connecting through an ODBC driver

Although JSP applications must communicate with databases through JDBC drivers, they can communicate through ODBC drivers if you have a JDBC-ODBC bridge driver. Using an ODBC driver lets you use a Windows DSN, which simplifies the task of creating the connection. The bridge driver acts as an interpreter between your JSP application, which talks JDBC, and your ODBC driver, which talks ODBC. This channel of communication lets your JSP application talk to the database.

This kind of connection offers two main advantages. First, you can use the free ODBC drivers from Microsoft. Second, you can use a DSN to simplify the task of creating the connection.

You must meet the following requirements to connect through an ODBC driver:

- Your JSP application server must be running on a Windows computer.
- An ODBC driver for your database must be installed on the Windows computer running the application server. For more information, see “Checking for the ODBC driver” on page 168.
- A JDBC-ODBC bridge driver must be installed on the Windows computer running the application server. For more information, see “Installing the Sun JDBC-ODBC Bridge driver” on page 168.

If you meet these requirements, you can connect through an ODBC driver. For instructions, see “Creating an ODBC connection” on page 169.

Checking for the ODBC driver

Make sure an ODBC driver for your database is installed on the Windows computer running the JSP application server. To find out whether or not an ODBC driver is installed, see “Viewing the ODBC drivers installed on a Windows system” on page 660. If an appropriate driver is not installed, you can download and install the Microsoft Data Access Components (MDAC) 2.5 and 2.6 packages on the computer running the JSP application server. You can download MDAC for free from the Microsoft website at <http://www.microsoft.com/data/download.htm>. These packages contain the latest ODBC drivers from Microsoft.

Note: Install MDAC 2.5 before installing MDAC 2.6.

If you have an ODBC driver for your database, you can install a JDBC-ODBC driver next.

Installing the Sun JDBC-ODBC Bridge driver

To connect through an ODBC driver, you must install the Sun JDBC-ODBC Bridge driver on the Windows computer running the JSP application server. The driver comes with the Sun Java 2 SDK, Standard Edition, for Windows.

To find out if you already have the Java 2 SDK with the driver, check your hard disk for either the `c:\jdk1.2.2` or the `c:\jdk1.3` folder.

Note: Java 1.2.2 and 1.3 are the same as Java 2.

If you don't have the SDK, you can download it from the Sun website at <http://java.sun.com/j2se/> and install it. The driver installs automatically when you install the SDK.

Although it is adequate for development use with lower-end database systems such as Microsoft Access, the Sun JDBC-ODBC Bridge driver is not intended for production use. For example, it lets only one JSP page connect to the database at a time (in other words, it does not support concurrent use by multiple threads). For more information on the driver's limitations, see article 12409 on the Macromedia support center at http://www.macromedia.com/go/jdbc-odbc_problems.

After you install the bridge driver, you can create the database connection next.

Creating an ODBC connection

Before connecting through an ODBC driver, make sure the appropriate ODBC driver and the Sun JDBC-ODBC Bridge driver are installed on the Windows computer running the JSP application server .

To connect through an ODBC driver in JSP:

- 1 Define a DSN on the Windows system hosting your application server.

For instructions, see “Creating a DSN” on page 671.

- 2 Open a JSP page in Dreamweaver, then open the Databases panel (Window > Databases).

The panel displays the connections defined for that site.

- 3 Click the plus (+) button on the panel and choose “Sun JDBC-ODBC Driver (ODBC Database)” from the pop-up menu.

The Sun JDBC-ODBC Driver (ODBC Database) dialog box appears.

- 4 Enter a name for the new connection.

Note: Do not use any spaces or special characters in the name.

- 5 Replace the `[odbc dsn]` placeholder in the URL box with the DSN you defined in step 1.

The URL box should look like this:

```
jdbc:odbc:myDSN
```

- 6 Specify the user name and password to access the database.

If you don't need a user name or password, leave the boxes blank. For example, if your DSN is called Acme and you don't need a user name or password to access the database, enter the following parameter values:

Driver: `sun.jdbc.odbc.JdbcOdbcDriver`

URL: `jdbc:odbc:Acme`

Username:

Password:

- 7 Specify the location of the JDBC-ODBC Bridge driver.

- If the driver is installed on the same computer as Dreamweaver, select the Using Driver On This Machine option.
- If the driver is not installed on the same computer as Dreamweaver, select the Using Driver On Testing Server option.

Macintosh users can ignore this step because all database connections use the application server.

8 Click Test.

Dreamweaver attempts to connect to the database. If the connection fails, double-check the DSN and the other connection parameters. If the connection still fails, check the settings for the folder Dreamweaver uses to process dynamic pages (see “Specifying where dynamic pages can be processed” on page 134).

9 Click OK.

The new connection appears in the Databases panel.

Editing or deleting a database connection

When you create a database connection, Dreamweaver stores the connection information in a file in the Connections subfolder in the site’s local root folder. Dreamweaver does not actually create a database connection for your JSP application until you define a recordset for a page in the application (see “Defining a recordset” on page 502). At that point, Dreamweaver writes code in the file to establish the connection, and inserts an include directive in your page. At runtime, the server inserts the connection code in your document.

To update a connection:

1 Open a JSP page in Dreamweaver, then open the Databases panel (Window > Databases).

A list of connections appears in the panel.

2 Right-click (Windows) or Control-click (Macintosh) the connection and choose Edit Connection from the pop-up menu.

The dialog box you used to create the connection appears.

3 Make the changes and click OK.

Dreamweaver automatically updates the include file, which updates all the pages in the site that use the connection.

If you rename a connection, update every recordset that uses the old connection name by double-clicking the recordset in the Bindings panel and choosing the new connection name in the Recordset dialog box.

To delete a connection:

1 Open a JSP page in Dreamweaver, then open the Databases panel (Window > Databases).

A list of connections appears in the panel.

2 Right-click (Windows) or Control-click (Macintosh) the connection and choose Delete Connection from the pop-up menu.

The dialog box you used to create the connection appears.

3 Confirm that you want to delete the connection.

To avoid getting errors after deleting a connection, update every recordset that uses the old connection by double-clicking the name of the recordset in the Bindings panel and choosing a new connection in the Recordset dialog box.

CHAPTER 11

Database Connections for PHP Developers

To use a database with a PHP application, you need to create a database connection in Macromedia Dreamweaver MX. This chapter describes how to create the database connection.

For PHP development, Dreamweaver only supports the MySQL database system. Other database systems such as Microsoft Access or Oracle are not supported. MySQL is open-source software you can download for free from the Internet for non-commercial use. For more information, see the MySQL website at <http://www.mysql.com/downloads/mysql.html>.

The chapter assumes you have set up a PHP application (see “Setting Up a Web Application” on page 127). It also assumes a MySQL database is set up on your local computer or on a system to which you have network or FTP access.

This chapter contains the following sections:

- “Connecting to a database” on page 171
- “Editing or deleting a database connection” on page 172

To connect to the sample database installed with Dreamweaver, see Dreamweaver Help (Help > Using Dreamweaver). To learn more about databases and database connections, see “Beginner’s Guide to Databases” on page 651.

Connecting to a database

This section describes how to connect to a database when developing a PHP application in Dreamweaver. It assumes you have one or more MySQL databases and that the MySQL server is started.

To create a database connection to your MySQL database:

- 1 Open a PHP page in Dreamweaver, then open the Databases panel (Window > Databases).
- 2 Click the plus (+) button on the panel and choose MySQL Connection from the pop-up menu. The MySQL Connection dialog box appears.
- 3 Complete the dialog box and click OK.

For more information, click the Help button in the dialog box.

The new connection appears in the Databases panel.

Editing or deleting a database connection

When you create a database connection, Dreamweaver stores the connection information in a file in the Connections subfolder in the site's local root folder. Dreamweaver does not actually create a database connection for your PHP application until you define a recordset for a page in the application (see “Defining a recordset” on page 502). At that point, Dreamweaver writes code in the file to establish the connection, and inserts an include directive in your page. At runtime, the server inserts the connection code in your document.

To update a connection:

- 1 Open a PHP page in Dreamweaver, then open the Databases panel (Window > Databases).

A list of connections appears in the panel.

- 2 Right-click (Windows) or Control-click (Macintosh) the connection and choose Edit Connection from the pop-up menu.

The dialog box you used to create the connection appears.

- 3 Make the changes and click OK.

Dreamweaver automatically updates the include file, which updates all the pages in the site that use the connection.

To delete a connection:

- 1 Open a PHP page in Dreamweaver, then open the Databases panel (Window > Databases).

A list of connections appears in the panel.

- 2 Right-click (Windows) or Control-click (Macintosh) the connection and choose Delete Connection from the pop-up menu.

The dialog box you used to create the connection appears.

- 3 Confirm that you want to delete the connection.

To avoid getting errors after deleting a connection, update every recordset that uses the old connection by double-clicking the name of the recordset in the Bindings panel and choosing a new connection in the Recordset dialog box.

Part III

Working with Page Code

Use the advanced coding tools in Dreamweaver to create or modify pages.

This part contains the following chapters:

- Chapter 12, “Setting Up Your Coding Environment”
- Chapter 13, “Coding in Dreamweaver”
- Chapter 14, “Optimizing and Debugging Your Code”
- Chapter 15, “Editing Code in Design View”

CHAPTER 12

Setting Up Your Coding Environment

You can adapt the coding environment in Macromedia Dreamweaver MX so it fits the way you work. For example, you can change the way you view code, set up different keyboard shortcuts, or import and use your favorite tag library.

This chapter covers the following topics:

- “Viewing your code” on page 175
- “Setting viewing preferences” on page 176
- “Setting coding preferences” on page 177
- “Customizing keyboard shortcuts” on page 180
- “Setting coding preferences” on page 177
- “Setting Validator preferences” on page 181
- “Managing tag libraries” on page 181
- “Importing custom tags into Dreamweaver” on page 184
- “Using an external HTML editor with Dreamweaver” on page 187

Viewing your code

You can view the source code for the current document in several ways: you can display it in the Document window by turning on Code view, you can split the Document window to display both the page and its associated code, or you can work in the Code inspector, a separate coding window.

Note: Code view and Code inspector share the same functionality.

This section contains instructions for changing the way you view your code.

To view code in the Document window:

Choose View > Code.

To view code in a separate window:

Choose Window > Others > Code Inspector.

To code and visually edit a page in the Document window at the same time:

1 Choose View > Code and Design.

The code appears in the top pane and the page appears in the bottom pane.

2 To display the page on top, choose View > Design View on Top.

3 To adjust the size of the panes in the Document window, drag the splitter bar to the desired position.

The splitter bar is located between the two panes.

Code view is updated automatically when you make changes in Design view. However, after making changes in Code view, you must manually update the document in Design view by clicking in Design view and pressing F5.

Setting viewing preferences

You can set word wrapping, display line numbers for the code, highlight invalid HTML code, set syntax coloring for code elements, and set indenting from the View > Code View Options menu.

To set options for Code view and the Code inspector:

1 Display at least one page in Code view or the Code inspector.

2 Choose View > Code View Options.

A pop-up menu appears with a check mark beside each option that is already set.

3 Select any of the following options from the menu:

- **Word Wrap** wraps the code so that you can view it without scrolling horizontally. This option doesn't insert line breaks; it just makes the code easier to view.
- **Line Numbers** displays line numbers along the side of the code.
- **Highlight Invalid HTML** causes Dreamweaver to highlight in yellow any invalid HTML that browsers don't support. When you select an invalid tag, the Property inspector displays information on how to correct the error.
- **Syntax Coloring** turns code coloring on and off. To change the coloring scheme, see "Setting code coloring preferences" on page 180.
- **Auto Indent** makes your code indent automatically when you press the Enter key while writing code. To change the indent spacing or tags that automatically indent, see "Setting code formatting preferences" on page 177.

If you selected a checked option, the option is turned off. Otherwise, the option is set.

Setting coding preferences

You can set the following preferences for coding in Dreamweaver:

- Code formatting preferences determine common code formatting options, such as line length and indentation. See “Setting code formatting preferences” on page 177. You can also apply your new formatting preferences to existing HTML documents. See “Applying new formatting preferences to existing documents” on page 178.
- Code Hints preferences let you enable or disable Code Hints, and let you customize the hints you see. Code Hints give you inline assistance while you type in Code view. For example, if you type an opening tag bracket (<), a menu appears listing possible tags. Instead of typing the rest of the tag by hand, you can select one of the hints listed in the menu and press Enter to enter it automatically. See “Setting Code Hints preferences” on page 179.
- Code rewriting preferences determine what changes, if any, Dreamweaver makes to your code when you open an HTML or script document. See “Setting code rewriting preferences” on page 179.
- Code coloring preferences lets you set special formatting and syntax coloring for code elements such as tags, comments, and scripts. See “Setting code coloring preferences” on page 180.

To set advanced preferences, use the Tag Library Editor (see “Managing tag libraries” on page 181).

Setting code formatting preferences

You can change the look of your code by specifying formatting preferences such as indentation, line length, and the case of tag and attribute names.

Note that all the preferences except “Override Case Of” affect only new documents and new additions to existing documents. That is, when you open a previously created HTML document, these formatting options are not applied to it; to reformat existing HTML documents, use the Apply Source Formatting command. For more information, see “Applying new formatting preferences to existing documents” on page 178.

To set code formatting preferences:

- 1 Select Edit > Preferences > Code Format or Dreamweaver > Preferences > Code Format (Mac OS X).

The Code Format dialog box appears.

2 Adjust any of the following settings in the dialog box:

Use specifies whether to indent using spaces or tabs.

Indent Size determines the size of indents. The size is measured in spaces if Use is set to spaces, or in tabs if Use is set to tabs.

Tab Size determines the size of tabs (measured in character spaces).

Automatic Wrapping adds “hard” returns once a line reaches the specified column width. (Dreamweaver inserts hard returns only in places where they don't change the appearance of the document in browsers, so some lines may remain longer than the Automatic Wrapping option specifies.) By contrast, the Wrap option in Code view adds a “soft” return for lines that extend beyond the width of the window.

Line Break Type specifies the type of remote server (Windows, Macintosh, or UNIX) that hosts your remote site. Choosing the correct type of line break characters ensures that your HTML source code appears correctly when viewed on the remote server. (For FTP, this setting applies only to binary transfer mode; Dreamweaver ASCII transfer mode ignores this setting. If you download files using ASCII mode, Dreamweaver sets line breaks based on the operating system of your computer; if you upload files using ASCII mode, the line breaks are all set to CR LF.) This setting is also useful when you are working with an external text editor that recognizes only certain kinds of line breaks. For example, use CR LF (Windows) if Notepad is your external editor, and CR (Macintosh) if SimpleText is your external editor.

Default Tag Case and **Default Attribute Case** control the capitalization of tag and attribute names. These options are applied to tags and attributes that you insert or edit in the Document window, but they are not applied to the tags and attributes that you enter directly in Code view, or to the tags and attributes in a document when you open it (unless you also select one or both of the Override Case Of options).

Override Case Of: Tags and Attributes specify whether to enforce your specified case options at all times, including when you open an existing HTML document. When you select one of these options, all tags or attributes in open documents are immediately converted to the specified case, as are all tags or attributes in each document you open from then on (until you deselect this option again). Tags or attributes you type in Code view and the Quick Tag Editor are also converted to the specified case, as are tags or attributes you insert using the Insert bar. For example, if you want tag names always to be converted to lowercase, specify lowercase in Case for Tags, and then select the Override Case Of: Tags option. Then when you open a document that contains uppercase tag names, Dreamweaver converts them all to lowercase.

Centering specifies whether elements should be centered using `div align="center"` or `center`. Both are part of the HTML 4.0 Transitional specification, but `center` is supported by a wider range of browsers.

Applying new formatting preferences to existing documents

The code formatting options that you specify in Code Format preferences only apply to any new documents that you create with Dreamweaver. To apply new formatting preferences to existing documents, use the Apply Source Formatting command.

To apply new formatting preferences to an existing document:

- 1 Open the document in Dreamweaver.
- 2 Choose Commands > Apply Source Formatting.

Setting Code Hints preferences

Code Hints lets you quickly insert tag names, attributes, and values as you enter code. To set the preferences, choose Edit > Preferences > Code Hints or Dreamweaver > Preferences > Code Hints (Mac OS X).

Tip: Even if Code Hints is disabled, you can display a pop-up hint in Code view or the Code inspector by pressing Control-space in Windows.

To set Code Hints preferences:

- 1 Select Edit > Preferences > Code Hints or Dreamweaver > Preferences > Code Hints (Mac OS X).

The Code Hints dialog box appears.

- 2 Adjust any of the following settings in the dialog box:

Enable Auto Tag Completion enables automatic tag completion.

Enable Code Hints enables Code Hints while entering code in Code view. Drag the Delay slider to set the time in seconds before the program displays appropriate hints.

Menus lets you set exactly which type of Code Hints you want displayed while typing. You can use all or some of the menus.

Setting code rewriting preferences

Code rewriting preferences determine what Dreamweaver does while opening documents. These preferences have no effect when you edit HTML or scripts in Code view.

If you turn off the rewriting options, Dreamweaver displays invalid markup items in the Document window for HTML that it would have rewritten.

If you are importing an HTML document from Microsoft Word, you can use the Clean Up Word HTML command to remove any unnecessary HTML code. For more information, see “Cleaning up Microsoft Word HTML” in Dreamweaver help (Help > Using Dreamweaver).

To set code rewriting preferences:

- 1 Select Edit > Preferences > Code Rewriting or Dreamweaver > Preferences > Code Rewriting (Mac OS X).

The Code Rewriting dialog box appears.

2 Adjust any of the following settings in the dialog box:

Fix Invalidly Nested and Unclosed Tags rewrites overlapping tags. For example, `<i>text</i>` is rewritten as `<i>text</i>`. This option also inserts closing quotation marks and closing brackets if they are missing.

Rename Form Items When Pasting ensures you don't have duplicate names for form objects.

Remove Extra Closing Tags deletes closing tags that have no corresponding opening tag.

Warn When Fixing or Removing Tags displays a summary of technically invalid HTML that Dreamweaver attempted to correct. The summary notes the location of the problem (using line and column numbers) so that you can find the correction and ensure that it's rendering as intended.

Never Rewrite Code: In Files with Extensions allows you to prevent Dreamweaver from rewriting code in files with the specified filename extensions. This option is particularly useful for files that contain third-party tags.

Encode Special Characters in URL Using % ensures the URL contains only legal characters.

Encode <, >, &, and " in Attribute Values Using & ensures the URL contains only legal characters.

Setting code coloring preferences

Syntax coloring helps you to identify code blocks in large documents. You can set syntax coloring preferences for code elements such as comments and scripts and for tags in general. To set color preferences for specific tags, edit the tag definition in the Tag Library Editor. For more information, see “Editing libraries, tags, and attributes” on page 183.

To set code coloring preferences:

- 1 Choose Edit > Preferences > Code Coloring or Dreamweaver > Preferences > Code Coloring (Mac OS X).
- 2 Choose a document type and click Edit Coloring Scheme.
The Edit Color Scheme dialog box appears.
- 3 Select an item and set its text color, background color, and style (bold, italic, or underline).
You can preview the color scheme in the Preview pane.
- 4 When done, click OK.

Customizing keyboard shortcuts

You can use your favorite keyboard shortcuts in Dreamweaver. If you're accustomed to using specific keyboard shortcuts—for example, Control-Enter to add a line break, Control-G to go to a specific position in the code, or Shift+F6 to validate a file—you can add them to Dreamweaver using the Keyboard Shortcut Editor. For more information, see “Using the Keyboard Shortcut Editor” on page 51.

Opening files in Code view by default

You can configure Dreamweaver to open files with certain extensions automatically in Code view instead of Design view.

When you open a file type that normally doesn't contain any HTML (for example, a JavaScript file), the file opens in Code view (or Code inspector) instead of Design view. You can edit the file, and save it as an HTML file or a non-HTML file.

To set the default view for non-HTML files:

- 1 Choose Edit > Preferences > File Types/Editors or Dreamweaver > Preferences > File Types/Editors (Mac OS X).
- 2 In the Open in Code View text box, type the filename extensions you want to open automatically in Code view. Put a space between each file type.

The default file extensions listed are .js, .txt, and .asa. You can add as many as you like.

Setting Validator preferences

You can use the Validator in Dreamweaver to quickly locate tag or syntax errors in your code (see “Validating your tags” on page 205). You can specify the tag-based languages against which the Validator should check, the specific problems that the Validator should check, and the types of errors that the Validator should report.

To set preferences for the Validator:

- 1 Choose Edit > Preferences > Validator or Dreamweaver > Preferences > Validator (Mac OS X).
- 2 Select the tag libraries against which to validate.

Some tag libraries are built on top of others. For example, if you select HTML 4.0, HTML 3.2 and HTML 2.0 are automatically selected, because the definition for HTML 4.0 is incomplete without them.

Note: If you validate both CFML and HTML in a single document, the Validator can't assess the number sign (#). This is because, in CFML, # is an error and ## is correct; while in HTML, ## is an error and # is correct.

- 3 Click Options.

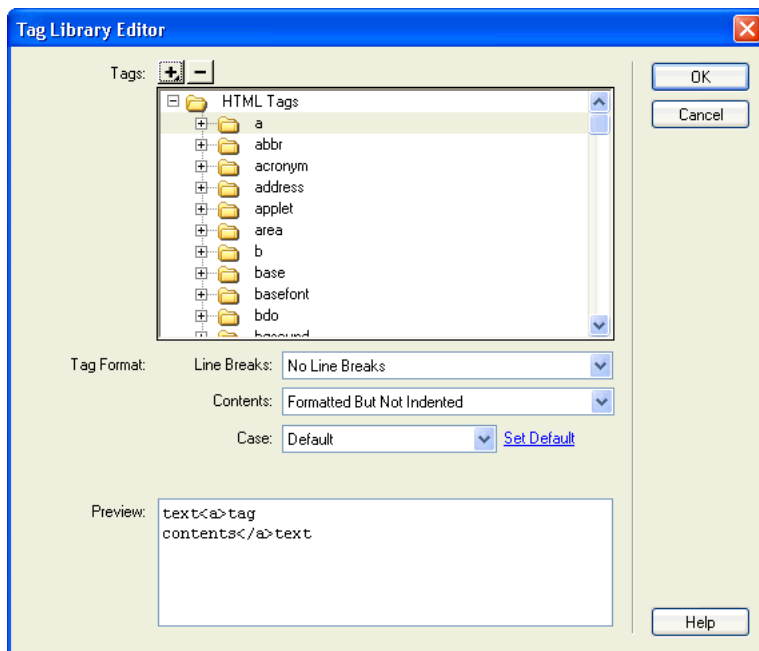
The Validator Options dialog box appears.

- 4 Select the Display option for every type of error that you want the Validator report to include.
- 5 Select items the Validator should look for.
- 6 Click OK to close the Validator Options dialog box, then click OK to close the Preferences dialog box.

Managing tag libraries

You can use the Tag Library Editor to manage the tag libraries in Dreamweaver. The Tag Library Editor lets you add and delete tag libraries, tags, and attributes; set properties for a tag library; and edit tags and attributes.

To open the Tag Library Editor, choose Edit > Tag Libraries. The Tag Library Editor appears as follows. (The contents of this dialog box changes depending on the selected tag.)



This section covers the following topics:

- “Adding libraries, tags, and attributes” on page 182
- “Editing libraries, tags, and attributes” on page 183
- “Deleting libraries, tags, and attributes” on page 184

See also “Importing custom tags into Dreamweaver” on page 184.

Adding libraries, tags, and attributes

You can use the Tag Library Editor to add tag libraries, tags, and attributes to the tag libraries in Dreamweaver.

Note: To import a tag, see “Importing custom tags into Dreamweaver” on page 184.

To add a tag library:

- 1 In the Tag Library Editor (Edit > Tag Libraries), click the plus (+) button and choose New Tag Library.

The New Tag Library dialog box appears.

- 2 In the Library Name text box, enter a name (for example, **PHP Tags**).
- 3 Click OK.

To add one or more tags to a tag library:

- 1 In the Tag Library Editor (Edit > Tag Libraries), click the plus (+) button and choose New Tags.
The New Tags dialog box appears.
- 2 In the Tag Library pop-up menu, choose a tag library (for example, CFML Tags).
By default, the tag library selected in the tag editor appears in the Tag Library box.
- 3 In the Tag Names text box, enter a name for the tag.
If you are adding more than one tag, enter the names for the tags, separated by a comma and space (for example, `cfgraph, cfgraphdata`).
- 4 To add every tag with a matching end tag (`</ . . >`), select Have Matching End Tags.
- 5 Click OK.

To add one or more attributes to a tag:

- 1 In the Tag Library Editor (Edit > Tag Libraries), click the plus (+) button and choose New Attributes.
The New Attributes dialog box appears.
- 2 In the Tag Library pop-up menu, choose a tag library.
By default, the tag library selected in the tag editor appears in the Tag Library text box.
- 3 In the Tag pop-up menu, choose a tag (for example, `cfgraph`).
By default, the tag selected in the tag editor appears in the Tag box.
- 4 In the Attribute Names text box, enter a name for the attribute (for example, `type`).
To add more than one attribute, enter the attributes' names, separated by a comma and space.
- 5 Click OK.

Editing libraries, tags, and attributes

You can use the Tag Library Editor to set properties for a tag library and edit tags and attributes in a library—whether it be their attributes and values, or their format (for easy identification in the code).

To set properties for a tag library:

- 1 In the Tag Library Editor (Edit > Tag Libraries), select a tag library in the Tags pop-up menu.
- 2 In the Used In box, select every document type that should use the tag library.
The tags in the tag library are only available in the types of documents that you select here.
- 3 If the tags in the tag library need a prefix, then enter the prefix in the Tag Prefix text box.
- 4 If you are finished making changes in the Tag Library Editor, click OK.

To edit a tag in a tag library:

- 1 In the Tag Library Editor (Edit > Tag Libraries), open a tag library in the Tags pop-up menu and select a tag.
- 2 Set any of the following Tag Format options:
 - **Line Breaks** specifies where Dreamweaver inserts line breaks for a tag.

- **Contents** specifies how Dreamweaver inserts the contents of a tag; that is, if it applies line break and formatting rules to the content.
- **Case** specifies the case for a specific tag. Choose from Default, Lowercase, Uppercase, or Mixed Case. If you choose Mixed Case, the Tag Name Mixed Case dialog box appears. Enter how Dreamweaver should insert the tag (for example, **getProperty**) and click OK.
- **Set Default** sets the default case for all tags. In the Default Tag Case dialog box that appears, choose <UPPERCASE> or <lowercase>, and click OK.

Tip: You might want to set your default case to be lowercase to comply with XML and XHTML standards.

To edit an attribute for a tag:

- 1 In the Tag Library Editor (Edit > Tag Libraries), open a tag library in the Tags pop-up menu, open a tag, and select a tag attribute.
- 2 In the Attribute Case pop-up menu, choose Default, Lowercase, Uppercase, or Mixed Case. If you choose Mixed Case, in the Attribute Name Mixed Case dialog box, enter how Dreamweaver should insert the attribute (for example, **onClick**). Click OK. Click the Set Default link to set the default case for all attributes.
- 3 In the Attribute Type pop-up menu, select the type of attribute. If you choose Enumerated, enter every allowed value for the attribute in the Values text box. Separate each value by a comma and no space. For example, for the type attribute of the `cfgraph` tag, enter **bar,horizontalBar,line,pie**.

Deleting libraries, tags, and attributes

You can use the Tag Library Editor to delete tag libraries, tags, and attributes.

To delete a library, tag, or attribute:

- 1 In the Tag Library Editor (Edit > Tag Libraries), select a tag library, tag, or attribute in the Tags pop-up menu.
- 2 Click the minus (-) button.
- 3 If you are asked to confirm the deletion, click OK. The item is removed from the Tags pop-up menu.
- 4 If you don't want to permanently delete the item, click Cancel. The Tag Library Editor closes without deleting the item, and without saving any changes that you have made in the Tag Library Editor since opening it.
- 5 To permanently delete the item, click OK.

Importing custom tags into Dreamweaver

You can import custom tags into Dreamweaver so that they become an integral part of the authoring environment. For example, when you start typing a custom tag in Code view, the Code Hints feature automatically lists the tag's attributes and lets you choose one.

This section covers the following topics:

- “Importing tags from XML files” on page 185

- “Importing custom ASP.NET tags” on page 185
- “Importing JSP tags from a file” on page 185
- “Importing JSP tags from a server (web.xml)” on page 186
- “Importing JRun tags” on page 186

Importing tags from XML files

You can import tags from an XML Document Type Definition (DTD) file or a schema.

To import tags from an XML DTD or schema:

- 1 Open the Tag Library Editor (Edit > Tag Libraries).
- 2 Click the plus (+) button and choose DTD Schema > Import XML DTD or Schema File.
- 3 Enter the file or URL of the DTD or schema file.
- 4 Enter the prefix to be used with the tags.
The prefix is used to identify a tag on a page as part of a particular tag library.
- 5 Click OK.

Importing custom ASP.NET tags

You can import custom ASP.NET tags into Dreamweaver.

Before you begin, make sure that the custom tag is installed on the testing server defined in the Site Definition dialog box (see “Specifying where dynamic pages can be processed” on page 134). Compiled tags (.dll files) must be placed in the site root’s bin folder. Non-compiled tags (.ascx files) can reside in any virtual directory or subdirectory on the server. For more information, see the Microsoft ASP.NET documentation.

To import ASP.NET custom tags into Dreamweaver:

- 1 Open an ASP.NET page in Dreamweaver.
- 2 Open the Tag Library Editor (Edit > Tag Libraries).
- 3 Click the plus (+) button and choose one of the following:
 - To import all the ASP.NET custom tags from the application server, choose ASP.NET > Import All ASP.NET Custom Tags.
 - To import only certain custom tags from the application server, choose ASP.NET > Import Selected ASP.NET Custom Tags.

The Import Selected ASP.NET Custom Tags dialog box appears, listing every ASP.NET custom tag installed on the application server. Control-click (Windows) or Command-click (Macintosh) tags from the list and click OK.

Importing JSP tags from a file

You can import a JSP tag library into Dreamweaver from a variety of file types.

To import a JSP tag library into Dreamweaver:

- 1 Open a JSP page in Dreamweaver.

- 2 Open the Tag Library Editor (Edit > Tag Libraries).
- 3 Click the plus (+) button and choose JSP > Import From File (*.tld, *.jar, *.zip).
- 4 Specify the .tld, .jar, or .zip file containing the tag library.
- 5 Enter a URI to identify the tag library.

The URI (Uniform Resource Identifier) often consists of the URL of the organization maintaining the tag library. The URL is not used to access the organization; it is used to uniquely identify the tag library.

- 6 Enter the prefix to be used with the tags.
The prefix is used to identify a tag on a page as part of a particular tag library.
- 7 Click OK.

Importing JSP tags from a server (web.xml)

You can import JSP tags from a server.

To import JSP tags from a server:

- 1 Open a JSP page in Dreamweaver.
- 2 Open the Tag Library Editor (Edit > Tag Libraries).
- 3 Click the plus (+) button and choose JSP > Import From Server (web.xml).

The Import from Server dialog box appears.

- 4 Select a tag library.
- 5 Enter a URI to identify the tag library.

The URI (Uniform Resource Identifier) often consists of the URL of the organization maintaining the tag library. The URL is not used to access the organization; it is used to uniquely identify the tag library.

- 6 Click OK.

Importing JRun tags

If you use Macromedia JRun, you can import your JRun tags into Dreamweaver.

To import JRun tags into Dreamweaver:

- 1 Open a JSP page in Dreamweaver.
- 2 Open the Tag Library Editor (Edit > Tag Libraries).
- 3 In the Tag Library Editor, click the plus (+) button and choose JSP > Import JRun Server Tags From Folder.
- 4 Specify a folder containing the JRun tags.
- 5 Enter a URI to identify the tag library.
- 6 Enter the prefix to be used with the tags.

The prefix is used to identify a tag on a page as part of a particular tag library.

- 7 If you are finished making changes, click OK to close the Tag Library Editor.

Using an external HTML editor with Dreamweaver

You can launch an external HTML or text editor from Dreamweaver to edit the source code for the current document and then switch back to Dreamweaver to continue editing graphically. Dreamweaver detects any changes that have been saved to the document externally and prompts you to reload the document upon returning.

You can use the following integrated HTML editors: Macromedia HomeSite (Windows only) or BBEdit (Macintosh only). You can also use any other text editor, such as Notepad, SimpleText, vi, emacs, and TextPad.

Using an integrated HTML editor

When you install Dreamweaver, you can install HomeSite in Windows or a trial version of BBEdit on the Macintosh. Dreamweaver is tightly integrated with both of these products.

Because of this integration, you can edit a document in both Dreamweaver and HomeSite/BBEdit, switching from one application to the other, and the document is kept in sync automatically in both applications. In addition, both programs track the current selection; for example, if you select text in Dreamweaver and switch to BBEdit, the same element is selected in BBEdit.

Using HomeSite (Windows only)

You don't need to enable integration for HomeSite; it's integrated automatically when the program is installed.

To use HomeSite:

- 1 Choose Edit > Edit with HomeSite.
- 2 Edit the document in HomeSite and save your changes.
- 3 To return to Dreamweaver, click Dreamweaver in the Editor toolbar.

Using BBEdit (Macintosh only)

You can disable BBEdit integration if you prefer to work with an older version of BBEdit or a different HTML text editor. Selections are not tracked if BBEdit integration is turned off.

To use BBEdit with Dreamweaver:

- 1 Choose Edit > Edit with BBEdit.
- 2 Edit the document in BBEdit.
- 3 Click the Dreamweaver button on the HTML Tools palette in BBEdit to return to Dreamweaver.

To disable BBEdit integration:

- 1 Choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X), and select File Types/Editors.
- 2 Deselect Enable BBEdit Integration and click OK.

Using external editors

You can open other external editors from Dreamweaver, but the document is not kept in sync in both applications like it is with HomeSite or BBEdit. When you are finished making changes in the external editor, you must manually refresh the document in Dreamweaver.

To choose an external HTML editor:

- 1 Choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X), and select File Types/Editors.
- 2 (Macintosh only) To use an HTML editor other than BBEdit, deselect the Enable BBEdit Integration option. To use BBEdit, leave Enable BBEdit Integration selected and skip the rest of these steps.
- 3 Click the Browse button beside the External Code Editor text box and choose a text editor.
- 4 In the Reload Modified Files option, specify what you want Dreamweaver to do when it detects that changes have been made externally to a document that is open in Dreamweaver.
- 5 In the Save on Launch option, specify whether Dreamweaver should always save the current document before launching the editor, never save the document, or prompt you to ask whether to save or not each time you launch the external editor.

To launch an external HTML editor:

Choose Edit > Edit with [editor name].

CHAPTER 13

Coding in Dreamweaver

Macromedia Dreamweaver MX offers a full-featured coding environment designed for any type of web development, from writing simple HTML pages to designing, testing, and deploying complex web applications.

This chapter contains the following topics:

- “About the Dreamweaver coding environment” on page 189
- “Writing and editing code” on page 191
- “Copying and pasting code” on page 195
- “Searching and replacing tags and attributes” on page 196
- “Accessing language references” on page 200

For information on migrating from Macromedia HomeSite to Dreamweaver, visit the Dreamweaver MX Support Center at http://www.macromedia.com/go/migrate_from_homesite.

Related topics

- “Setting Up Your Coding Environment” on page 175
- “Optimizing and Debugging Your Code” on page 203
- “Editing Code in Design View” on page 217

About the Dreamweaver coding environment

The coding environment in Dreamweaver gives you full access to the code in your pages. This section covers the following topics:

- “Supported file types” on page 190
- “How Dreamweaver respects your code” on page 190
- “About Roundtrip HTML” on page 190
- “About the code inserted by server behaviors” on page 190

Related topic

- “Viewing your code” on page 175

Supported file types

For client-side coding, you can work on numerous file types in Dreamweaver, including HTML, XML, Cascading Style Sheets (CSS), JavaScript, VBScript, Wireless Markup Language (WML), Extension Data Markup Language (EDML), Dreamweaver template (.dwt), and Text.

For server-side coding, you can work with the following types of files: ColdFusion, ColdFusion component (.cfc), ASP.NET Visual Basic, ASP.NET C#, ASP VBScript, ASP JavaScript, JSP, and PHP. You can also create stand-alone C#, Visual Basic, and Java files.

How Dreamweaver respects your code

Dreamweaver generates valid, clean code in both its coding and visual environments. Dreamweaver also respects the code that you write by hand or that another editor generates. Dreamweaver never rewrites your code. For example, Dreamweaver does not alter your white space or change the case of attributes. If you choose the View > Code View Options > Highlight Invalid HTML option, Dreamweaver highlights any errors in yellow. Optionally, you can clean up your code according to criteria that you specify. You can also customize how Dreamweaver formats code. For more information, see “Setting coding preferences” on page 177.

About Roundtrip HTML

Roundtrip HTML lets you move your documents back and forth between a text-based HTML editor and Dreamweaver with little or no effect on the content and structure of the document’s original HTML source code. Its key features include the following:

- By default, when you switch back to Dreamweaver from an external editor, or when you open an HTML document that was not created with Dreamweaver, Dreamweaver rewrites overlapping tags, adds required closing tags, and removes extra closing tags. However, Dreamweaver does not make any other changes in the code, even if the code is invalid, provided that browsers can properly render the invalid code.

You can disable code rewriting (see “Setting code rewriting preferences” on page 179).

- Dreamweaver does not change tags it doesn’t recognize—including XML tags—because it has no criteria by which to judge them. If an unrecognized tag overlaps another tag (for example, `<MyNewTag>text</MyNewTag></code>), Dreamweaver marks it as an error but doesn’t rewrite the code.`
- Optionally, you can set Dreamweaver to highlight in yellow invalid HTML that browsers don’t support. When you select a highlighted section, the Property inspector displays information on how to correct the error.
- Dreamweaver lets you launch a third-party text editor to edit the current document. For more information, see “Using an external HTML editor with Dreamweaver” on page 187.

About the code inserted by server behaviors

When you develop a dynamic page and choose a server behavior from the Server Behaviors panel, Dreamweaver inserts one or more code blocks (or “participants”) into your page to make the server behavior work.

If you change the code within a participant, you can no longer use a visual tool to edit or delete the server behavior. Dreamweaver looks for specific patterns in the page code to detect server behaviors and display them in the Server Behaviors panel. If you change a participant's code in any way, including changing word spacing and line breaks, Dreamweaver can no longer detect the server behavior and display it in the Server Behaviors panel. However, the server behavior still exists on the page, and you can edit it in the coding environment in Dreamweaver.

Writing and editing code

Dreamweaver offers several features to help you write and edit code efficiently. This section covers these features, as follows:

- “Using Code Hints” on page 191
- “Working with code snippets” on page 192
- “Inserting code quickly with the Insert bar” on page 193
- “Inserting tags with the Tag Chooser” on page 193
- “Editing tags with tag editors” on page 194
- “Editing tags with the Tag inspector” on page 194
- “Indenting code blocks” on page 194
- “Copying and pasting code” on page 195
- “Inserting HTML comments” on page 195
- “Listing JavaScript and VBScript functions in your page” on page 196

Using Code Hints

Code Hints lets you insert and edit code as you type in Code view (or Code inspector). When you type certain characters, a list appears, suggesting options to complete your entry. You can use this feature to insert or edit code, or just to see the available attributes for a tag, the available parameters for a function, or the available methods for an object.

You can set how long Code Hints waits before displaying a list of options, specify the tags to include in the list of tags, or turn off Code Hints altogether. For more information, see “Setting Code Hints preferences” on page 179.

To insert a tag using Code Hints:

- 1 Type a start bracket (<) to display a list of tags.
- 2 Press the Escape key to close the list.
- 3 Scroll down the list and double-click a tag to insert it.
- 4 If the tag supports attributes, press the spacebar to display a list of allowed attributes for the tag. Select an attribute and press Enter.
- 5 Type the value for the attribute or, if the attribute takes only certain values, select a value from the list of allowed values for the attribute.
- 6 For every attribute that you want to add for the tag, repeat the previous two steps, making sure that you don't press the spacebar between a value and its end quote (").
- 7 Type the end bracket for the tag (>) after the last attribute-value pair.

To edit a tag, do any of the following:

- To add an attribute, place the insertion point to the left of a tag end bracket (>) and press the spacebar to display a list of any supported attributes for the tag. Insert the attribute and its value as previously described.
- To change an attribute, delete the attribute and add an attribute, as previously described.
- To change a value, delete the value and add a value, as previously described.

To insert or edit code for a function or object:

- 1 Choose Edit > Preferences > Code Hints or Dreamweaver > Preferences > Code Hints (Mac OS X), then set the preferences to show without any delay.
- 2 In Code view (or Code inspector), type your functions or object code as usual.

Whenever you type a keyword or character for which Code Hints is available, a pop-up menu of completion options appears. When desired, select from the list and press Enter.

To configure or disable Code Hints, see “Setting Code Hints preferences” on page 179.

Working with code snippets

Code snippets let you store content for quick reuse. You can create and insert snippets of HTML, JavaScript, CFML, ASP, JSP, and more. Dreamweaver also contains some pre-defined snippets that you can use as a starting point.

You can determine if you want the code snippet to wrap around a selection or to exist only as a block of code. For example, you might link to a page that appears many times in your site, but it's attached to different objects and text. You can select an object, right-click (Windows) or Control-click a snippet in the Snippets panel, and choose Insert from the pop-up menu. The code for the link will wrap around the selection. You can also comment your snippets to help others to use them properly.

To insert a code snippet:

- 1 Place the insertion point in the desired position in the document; to insert a code snippet around a selection, make the selection in the document.
- 2 In the Snippets panel (Window > Snippets), double-click the snippet.

You can also right-click (Windows) or Control-click (Macintosh) the snippet, then choose Insert from the pop-up menu.

To create a code snippet:

- 1 In the Snippets panel, click the New Snippet icon at the bottom of the panel.

The Snippet dialog box appears.

- 2 Complete the dialog box and click OK.

For more information, click the Help button in the dialog box.

To edit or delete a code snippet:

- 1 In the Snippets panel, select a snippet and click the Edit Snippet or Remove icon at the bottom of the panel.

You can identify the Edit Snippet or Remove icon by its tooltip.

To create code snippet folders and manage code snippets:

- 1 In the Snippets panel, click the New Snippet Folder icon at the bottom of the panel.
- 2 Drag snippets in the new folder or other folders, as desired.

To share a snippet with other members of your team:

- 1 Find the snippet that you want to share in the Configuration\Snippets folder in the Dreamweaver MX application folder.
 - 2 Copy the snippet to a shared folder on your computer or a network computer.
 - 3 Have the other members of the team copy the snippet to their Configuration\Snippets folder.
- Anyone with access to the shared folder can now add, edit, and delete snippets from the folder.

Inserting code quickly with the Insert bar

You can use the Insert bar to quickly add code to your page. Simply position the insertion point in the code, then click an icon in the Insert bar. When you click an icon, the code may appear in your page immediately, or a dialog box may appear requesting more information to complete the code.

To find out what each icon does, place the mouse over the icon and wait for a tooltip to appear. The number and type of icons available on the Insert bar varies depending on the document type opened in Dreamweaver. It also depends on whether you're using Code view or Design view.

Though the Insert bar provides a collection of frequently used tags, it is not comprehensive. To choose from a more comprehensive selection of tags, use the Tag Chooser.

Inserting tags with the Tag Chooser

You can use the Tag Chooser to insert in your page any tag in the Dreamweaver tag libraries (which include ColdFusion and ASP.NET tag libraries). For more information on the tag libraries, see "Managing tag libraries" on page 181.

To insert a tag using Tag Chooser:

- 1 Position the insertion point in the code, then right-click (Windows) or Control-click (Macintosh) and choose Insert Tag.
The Tag Chooser appears. The left pane contains a list of supported tag libraries, and the right pane shows the individual tags in the selected tag library folder.
- 2 Select a tag from the list.
- 3 If you want to view syntax and usage information for the tag, click the Tag Info button.
If available, help for the tag displays in the dialog box.
- 4 If you want reference information about the tag, click the <?> icon.
If available, reference information about the tag appears in the Reference panel.
- 5 Click Insert to insert the selected tag into your code.
Tags that don't require additional information, which appear in the right pane with brackets (for example, <HTML></HTML>), are inserted into the document at the insertion point. All other tags display their own tag editor.
- 6 If a tag editor opens, enter the additional information and click OK.
See "Editing tags with tag editors" on page 194.

Editing tags with tag editors

Tag editors let you specify or edit the attributes for a tag using a dialog box. If you use the Tag Chooser to insert a tag, and the tag requires additional information, its tag editor appears automatically. Otherwise, the following procedure explains how to open the tag editor for a tag. For more information, see “Inserting tags with the Tag Chooser” on page 193.

To access the tag editor of a tag:

- 1 Place the insertion point within a tag.
- 2 Right-click (Windows) or Control-click (Macintosh) and select Edit Tag from the pop-up menu.
- 3 Make your changes and click OK.

Tip: To get more information about the tag within the tag editor, click Tag Info.

Editing tags with the Tag inspector

You can use the Tag inspector to edit tags in a property sheet similar to the ones found in other integrated development environments (IDEs).

To edit a tag using the Tag inspector:

- 1 Make sure the Tag inspector is open (Window > Tag Inspector).
- 2 In Code view (or Code inspector), click anywhere inside a tag.

If the tag supports attributes, Dreamweaver displays the attributes and their current values in the Tag inspector’s property sheet.

- 3 Make the changes to the tag in the property sheet.

Click an attribute name to type a value or, if the attribute takes pre-defined values, to select from a list.

- 4 If the attribute takes a value from a source of dynamic content (like a database), click the lightning bolt icon at the end of the selected attribute’s row and select the source.

For information on defining sources of dynamic content, see “Defining Sources of Dynamic Content” on page 501.

- 5 Click elsewhere in the panel to update the tag in your document.

Indenting code blocks

As you write and edit your code in Code view or Code inspector, you can indent code blocks to visually separate the block from the rest, making it easier to find.

To indent a code block:

- 1 Select the line or lines of the code block that you want to indent. (You must select the entire line.)
- 2 Choose Edit > Indent Code.

The code is indented. Choose Edit > Outdent Code to move the indented code back.

Copying and pasting code

You can copy and paste source code from another application or from Code view itself. You can copy and paste the code as text or as code with tags intact.

To copy and paste source code as text:

- 1 Copy the code from Dreamweaver or another application.
- 2 Place the insertion point directly in Design view, and choose Edit > Paste.

To copy and paste source code with tags intact:

- 1 Copy the code from Dreamweaver or another application.
To copy from Design view, choose Edit > Copy as HTML.
- 2 Do one of the following:
 - In Code view, position the insertion point, and choose Edit > Paste. If you are using Design view, press F5 to update its display of the document.
 - In Design view, position the insertion point, and choose Edit > Paste as HTML.

Inserting HTML comments

A comment is descriptive text that you insert in the HTML code to explain the code or provide other information. The text of the comment appears only in Code view and will not display in your web page in the browser.

To insert a comment:

- 1 In Code view or Design view, place the insertion point where you want the comment.
- 2 Click the Comment icon on the Common tab of the Insert bar.

Use the tooltips to identify the Comment icon.

- In Code view, Dreamweaver inserts a `<!-- -->` tag in the page. Enter your comment inside the tag.
- In Design view, Dreamweaver displays the Comment dialog box. Enter your comment and click OK.

To display comment markers in Design view, choose View > Visual Aids > Invisible Elements. Make sure that comments are selected in the Invisible Elements preferences, or the comment marker will not appear.

To add to an existing comment, do one of the following:

- Select the Comment marker in Design view and enter text in the Property inspector.
- Find the comment in the code and add text directly to it.

Listing JavaScript and VBScript functions in your page

You can view all of the JavaScript or VBScript functions in a page open in Code view.

To view the script functions in a page:

- 1 Open the document in Code view
- 2 Make sure the Document toolbar is displayed (View > Toolbars > Document).
- 3 Click the Code Navigation icon in the toolbar.

The icon consists of a pair of braces ({}).

Note: The icon is disabled in Design view.

If your code contains JavaScript or VBScript functions, they will appear in the pop-up menu. To see the functions listed in alphabetical order, hold down Control (Windows) or Option (Macintosh) while clicking the Code Navigation button.

To go to a function in your code:

Select a function from the Code Navigation pop-up menu. The function will be highlighted in Code view (or the Code inspector).

Searching and replacing tags and attributes

You can use Dreamweaver to search and replace tags and attributes in your code. This section covers the following topics:

- “About HTML source code searches” on page 196
- “Searching for tags and attributes” on page 197
- “Searching for text contained in specific tags” on page 197
- “Saving search patterns” on page 198
- “About regular expressions” on page 199

About HTML source code searches

Use the Source Code option in the Find or Replace dialog box to search for specific text strings in the HTML source code. For example, a search for **black dog** in the following code would produce two matches (in the alt attribute and in the first sentence):

```
<br>
We saw several black dogs in the park yesterday. The black
<a href="barnaby.html">dog</a> we liked best was called Barnaby.
```

The phrase `black dog` also appears in the second sentence, but it doesn't match because it's interrupted by a link.

For more information, see “Searching and replacing text” on page 294.

Searching for tags and attributes

Use the Specific Tag option in the Find or Replace dialog box to search for specific tags, attributes, and attribute values. For example, you can search for all `img` tags with no `alt` attribute.

To perform a tag search:

- 1 Choose Edit > Find and Replace, and specify which files to search.
- 2 Choose Specific Tag from the Search For pop-up menu.
- 3 Choose a specific tag from the pop-up menu next to Search For pop-up menu, enter a tag, or choose [any tag].

If you only want to search for all occurrences of the specified tag, press the minus (-) button and skip to step 6. Otherwise, proceed with step 4.

- 4 Limit the search with one of the following tag modifiers:
 - **With Attribute** chooses an attribute that must be in the tag for it to match. You can specify a particular value for the attribute or choose [any value].
 - **Without Attribute** chooses an attribute that can't be in the tag for it to match. For example, choose this option to search for all `IMG` tags with no `ALT` attribute.
 - **Containing** specifies text or a tag that must be contained within the original tag for it to match. For example, in the code `heading 1`, the `font` tag is contained within the `b` tag.
 - **Not Containing** specifies text or a tag that can't be contained within the original tag for it to match.
 - **Inside Tag** specifies a tag that the target tag must be inside of for it to match.
 - **Not Inside Tag** specifies a tag that the target tag can't be inside of for it to match.
- 5 Click the plus (+) button and repeat step 4 to limit the search further.
- 6 Initiate the search:
 - Click Find Next to highlight the next instance of the tag in the current document, then in any subsequent document if searching in more than one document.
 - Click Find All to generate a list of all the instances of the tag in the current document or, if you are searching a directory or site, generate a list of documents that contain the tag. The list appears in the Search tab of the Results panel.

Searching for text contained in specific tags

Use the Text (Advanced) option in the Find or Replace dialog box to search for specific text strings that are either within or not within a set of container tags. For example, you can search for the word **Untitled** between `<title>` tags to find all the untitled pages on your site.

To perform an advanced text search:

- 1 Choose Edit > Find and Replace, and specify the files to search.
- 2 Choose Text (Advanced) from the Search For pop-up menu.
- 3 Enter the text you want to search for in the text field adjacent to the Search For pop-up menu. For example, type the word **Untitled**.

- 4 Choose Inside Tag or Not Inside Tag, and then choose a tag from the adjacent pop-up menu. For example, choose Inside Tag and then `title`.
- 5 Click the plus (+) button to limit the search to tags with a specific attribute or attributes. Since the `<title>` tag has no attributes, you won't need to use this option to find all the untitled pages on your site.
- 6 Initiate the search:
 - Click Find Next to highlight the next instance of the text in the current document, then in any subsequent document if searching in more than one document.
 - Click Find All to generate a list of all the instances of the text in the current document or, if you are searching a directory or site, generate a list of documents that contain the tag. The list appears in the Search tab of the Results panel.

Saving search patterns

You can save search patterns for later use by clicking the Save Query icon (the disk icon) in the Find or Replace dialog box. Saving a query is a good idea if you regularly perform the same search and don't want to reconstruct the search pattern every time. For example, you might want to strip out nonstandard tags from documents created with another visual HTML editor, or to confirm that all images in a file have `height`, `width`, and `alt` attributes before the document goes live on the web.

To save a search pattern:

- 1 In the Find and Replace dialog box (Edit > Find and Replace), set the parameters for the search. If you are performing a tag or advanced text search, see “Searching for tags and attributes” on page 197 or “Searching for text contained in specific tags” on page 197 for information about setting additional search parameters.
- 2 Click the Save Query icon (the disk icon).

The default location where queries are saved is the Configuration/Queries folder inside the Dreamweaver application folder.
- 3 In the dialog box that appears, give the file a name that identifies the query and click Save.

For example, if the search pattern involves looking for `img` tags with no `alt` attribute, you might name the query `img_no_alt.dwr`. Find queries that end in the extension `.dwq`; replace queries that end in the extension `.dwr`.

To recall a search pattern:

- 1 Choose Edit > Find and Replace.
- 2 Click the Load Query button (the folder icon).

The Load Query dialog box automatically opens in the Configuration/Queries folder. You can navigate to another folder if you have saved queries elsewhere.
- 3 Select a query file and click Open.

If you are in the Find dialog box, only Find queries (`.dwq` files) are available. If you are in the Replace dialog box, both Find queries (`.dwq` files) and Replace queries (`.dwr` files) are available.
- 4 Click Find Next, Find All, Replace, or Replace All to initiate the search.

About regular expressions

Regular expressions are patterns that describe character combinations in text. Use them in your searches to help describe concepts such as “sentences that begin with ‘The’” and “attribute values that contain a number.” The following table lists the special characters in regular expressions, their meanings, and usage examples.

To search for text containing one of the special characters in the table, “escape” the special character by preceding it with a backslash. For example, to search for the actual asterisk in the phrase `some conditions apply*`, your search pattern might look like this: `apply*`. If you don’t escape the asterisk, you’ll find all the occurrences of “apply” (as well as any of “appl”, “applyy”, and “applyyy”), not just the ones followed by an asterisk.

Character	Matches	Example
<code>^</code>	Beginning of input or line.	<code>^T</code> matches “T” in “This good earth” but not in “Uncle Tom’s Cabin”
<code>\$</code>	End of input or line.	<code>h\$</code> matches “h” in “teach” but not in “teacher”
<code>*</code>	The preceding character 0 or more times.	<code>um*</code> matches “um” in “rum”, “umm” in “yummy”, and “u” in “huge”
<code>+</code>	The preceding character 1 or more times.	<code>um+</code> matches “um” in “rum” and “umm” in “yummy” but nothing in “huge”
<code>?</code>	The preceding character at most once (that is, indicates that the preceding character is optional).	<code>st?on</code> matches “son” in “Johnson” and “ston” in “Johnston” but nothing in “Appleton” or “tension”
<code>.</code>	Any single character except newline.	<code>.an</code> matches “ran” and “can” in the phrase “bran muffins can be tasty”
<code>x y</code>	Either x or y.	<code>FF0000 0000FF</code> matches “FF0000” in <code>bgcolor="#FF0000"</code> and “0000FF” in <code>font color="#0000FF"</code>
<code>{n}</code>	Exactly n occurrences of the preceding character.	<code>o{2}</code> matches “oo” in “loom” and the first two o’s in “mooooo” but nothing in “money”
<code>{n,m}</code>	At least n, and at most m, occurrences of the preceding character.	<code>F{2,4}</code> matches “FF” in “#FF0000” and the first four F’s in #FFFFFF
<code>[abc]</code>	Any one of the characters enclosed in the brackets. Specify a range of characters with a hyphen (for example, [a-f] is equivalent to [abcdef]).	<code>[e-g]</code> matches “e” in “bed”, “f” in “folly”, and “g” in “guard”
<code>[^abc]</code>	Any character not enclosed in the brackets. Specify a range of characters with a hyphen (for example, [^a-f] is equivalent to [^abcdef]).	<code>[^aeiou]</code> initially matches “r” in “orange”, “b” in “book”, and “k” in “eek!”
<code>\b</code>	A word boundary (such as a space or carriage return).	<code>\bb</code> matches “b” in “book” but nothing in “goober” or “snob”
<code>\B</code>	A nonword boundary.	<code>\Bb</code> matches “b” in “goober” but nothing in “book”
<code>\d</code>	Any digit character. Equivalent to [0-9].	<code>\d</code> matches “3” in “C3PO” and “2” in “apartment 2G”
<code>\D</code>	Any nondigit character. Equivalent to [^0-9].	<code>\D</code> matches “S” in “900S” and “Q” in “Q45”
<code>\f</code>	Form feed.	
<code>\n</code>	Line feed.	

Character	Matches	Example
\r	Carriage return.	
\s	Any single white-space character, including space, tab, form feed, or line feed.	\sbook matches "book" in "blue book" but nothing in "notebook"
\S	Any single non-white-space character.	\Sbook matches "book" in "notebook" but nothing in "blue book"
\t	A tab.	
\w	Any alphanumeric character, including underscore. Equivalent to [A-Za-z0-9_].	b\w* matches "barking" in "the barking dog" and both "big" and "black" in "the big black dog"
\W	Any non-alphanumeric character. Equivalent to [^A-Za-z0-9_].	\W matches "&" in "Jake & Mattie" and "%" in "100%"
Control+Enter or Shift+Enter (Windows), or Control+ Return or Shift+Return or Command+ Return (Macintosh)	Return character. Make sure that you deselect the Ignore Whitespace Differences option when searching for this, if not using regular expressions. Note that this matches a particular character, not the general notion of a line break; for instance, it doesn't match a tag or a <p> tag. Return characters appear as spaces in the Document window, not as line breaks.	

Use parentheses to set off groupings within the regular expression to be referred to later; use \$1, \$2, \$3, and so on (use (\$) in the Find field and use the backslash (\) in the Replace field) to refer to the first, second, third, and later parenthetical groupings. For example, searching for (\d+)\s+(\d+)\s+(\d+) and replacing it with \$2/\$1/\$3 swaps the day and month in a date separated by slashes (to convert between American-style dates and European-style dates).

Accessing language references

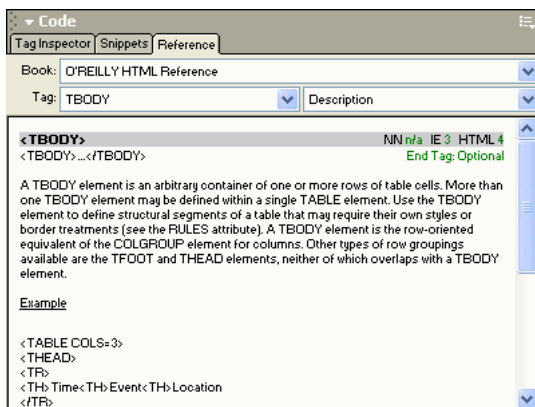
The Reference panel provides you with a quick reference tool for markup languages, JavaScript objects, and CSS styles with their attributes. The Reference panel provides you with information on the specific tags, objects, or styles that you are working with in Code view (or Code inspector).

To open the Reference panel:

- 1 In Code view, right-click (Windows) or Control-click (Macintosh) a tag, attribute, or keyword.
- 2 Choose Reference from the pop-up menu.

The Reference panel opens and displays information about the tag, attribute, or keyword you clicked. To adjust the text size in the Reference panel, select Large Font, Medium Font, or Small Font from the options menu (small arrow at top-right of panel).

The Book pop-up menu displays the name of the book that the reference material comes from. To display tags, objects, or styles from another book, choose a different book from the Book pop-up menu.



The Tag, Object, or Style (depending on which book you have selected) pop-up menu displays the tag, object, or style that you selected in the Code view (or Code inspector). To view information about another tag, object, or style, select a new one from the menu.

Next to the Tag, Object, or Style pop-up menu is a menu that contains the list of attributes for the tag you choose. The default selection is Description, which displays a description of the chosen tag. You can also select an attribute from the menu to view more information about it.

CHAPTER 14

Optimizing and Debugging Your Code

You can use Macromedia Dreamweaver MX to optimize and debug your code. For example, you can validate your tags, make your document XHTML compliant, or debug your JavaScripts.

This chapter covers the following topics:

- “Cleaning up your code” on page 203
- “Verifying that tags and braces are balanced” on page 204
- “Validating your tags” on page 205
- “Making pages XHTML compliant” on page 205
- “Using the JavaScript debugger” on page 209
- “Using the ColdFusion debugger” on page 215

Cleaning up your code

Use the Clean Up HTML or Clean Up XHTML command to remove empty tags, combine nested `font` tags, and otherwise improve messy or unreadable HTML/XHTML code.

For an XHTML document, the Clean Up XHTML command fixes syntax errors, sets the case of tag attributes to lowercase, and adds or reports the missing required attributes for a tag. (After executing the Clean Up XHTML command, a dialog box appears, listing the number of `img` and `area` tags that don't have an `alt` attribute.)

To clean up HTML generated from a Microsoft Word document, see “Cleaning up Microsoft Word HTML” in Dreamweaver help (Help > Using Dreamweaver).

To clean up the code:

1 Open a document.

- If the document is in HTML, choose Commands > Clean Up HTML.
- If the document is in XHTML, choose Commands > Clean Up XHTML.

2 In the dialog box that appears, select from the following options:

- **Remove Empty Container Tags** removes any tags that have no content between them. For example, `` and `` are considered empty tags, but the `` tag in `some text` is not.
- **Remove Redundant Nested Tags** removes all redundant instances of a tag. For example, in the code `This is what I really wanted to say `, the `b` tags surrounding the word “really” are redundant and would be removed.

- **Remove Non-Dreamweaver HTML Comments** removes all comments that were not inserted by Dreamweaver. For example, `<!--begin body text-->` would be removed, but `<!--#BeginEditable "doctitle" -->` wouldn't, because it's a Dreamweaver comment that marks the beginning of an editable region in a template.
 - **Remove Dreamweaver Special Markup**
 - **Remove Specific Tag(s)** removes the tags specified in the adjacent text field. Use this option to remove custom tags inserted by other visual editors and other tags that you don't want to appear on your site (for example, `blink`). Separate multiple tags with commas (for example, `font, blink`).
 - **Combine Nested `` Tags When Possible** consolidates two or more `font` tags when they control the same range of text. For example, `big red` would be changed to `big red`.
 - **Show Log on Completion** displays an alert box with details about the changes made to the document as soon as the cleanup is finished.
- 3 Click OK.

Depending on the size of your document and the number of options selected, it may take several seconds to complete the cleanup.

Verifying that tags and braces are balanced

As you write and edit your code in Code view or the Code inspector, you can check to make sure the tags or braces ({}) in the page are balanced. Checking to see if tags are balanced ensures that all tags have matching opening tags and closing tags, and is especially useful when you use multiple nesting levels in your code—for example, if you have nested tables within other tables.

To check for balanced tags:

- 1 Open the document in Code view.
- 2 Place the insertion point in the nested code you want to check.
- 3 Choose Edit > Select Parent Tag.

The enclosing matching tags are highlighted in your code. If you keep choosing Edit > Select Parent Tag, and your tags are balanced, eventually Dreamweaver will highlight the outermost `<html>` and `</html>` tags.

To check for balanced braces:

- 1 Open the document in Code view.
- 2 Place the insertion point in the nested code you want to check.
- 3 Choose Edit > Balance Braces.

The enclosing matching tags are highlighted in your code. Choosing Edit > Balance Braces again will highlight the matching tags that enclosed the previous selection.

Validating your tags

You can use the Dreamweaver Validator to find out if your code has tag or syntax errors. The Validator supports many tag-based languages, including several versions of HTML, XHTML, ColdFusion Markup Language (CFML), JavaServer Pages (JSP), and Wireless Markup Language (WML). You can also validate an XML document.

Note: To validate your document for accessibility, see “Testing your website for accessibility” on page 343.

You can set preferences for the Validator, such as specifying the tag-based languages against which the Validator should check, the specific problems that the Validator should check, and the types of errors that the Validator should report. For more information, see “Setting Validator preferences” on page 181.

You can run the Validator for the current document or for a selected tag.

To run the validator:

1 Do one of the following:

- For an XML (or XHTML) file, choose File > Check Page > Validate as XML.
- Otherwise, choose File > Check Page > Validate Markup.

The Validation tab of the Results panel displays either a “No errors or warnings” message or lists the syntax errors it found.

2 Double-click an error message to highlight it in the document.

3 Right-click (Windows) or Control-click (Macintosh) in the report to save it as an XML file or to view it in a browser (which lets you print the report).

Making pages XHTML compliant

This section describes XML and XHTML, XHTML requirements, Cascading Style Sheets (CSS) requirements for XHTML, and how to create XHTML-compliant documents.

This section covers the following topics:

- “About XML and XHTML” on page 205
- “About the XHTML code generated by Dreamweaver” on page 206
- “Using Cascading Style Sheets (CSS) with XHTML” on page 208
- “Making documents XHTML compliant” on page 208

About XML and XHTML

Extensible Markup Language, or XML, defines the structure of information, or how the information could be stored in a database. For example, an XML language for cooking could have a <recipe> tag, and <recipe> could contain tags for <header>, <ingredients>, and <instructions>. The power of XML lies in the fact that the information can be stored and retrieved from a database, rendered in a variety of formats, and used for a variety of purposes (for example, on a web page, hand-held computer, or cell phone; or for a cookbook, recipe card club, or book about cilantro).

Extensible Hypertext Markup Language, or XHTML, is a reformulation of HTML as an XML application. It's almost identical to HTML 4.01, but it's more strict and clean.

Benefits of XHTML

Generally, using XHTML lets you realize the benefits of XML, while ensuring the backward and future compatibility of your web documents. Following are some specific reasons to use XHTML:

- XHTML is designed to replace HTML. If you want your Internet website to be visible and properly rendered in the future, it's a sound investment to begin coding in XHTML now.
- XHTML is designed to be operable across devices, not just on PCs.
- XHTML is an XML language, so it offers the potential for extensibility. You can also use standard XML tools to view, edit, and validate your XHTML documents.
- XHTML documents can use applications such as scripts and applets that rely on the HTML Document Object Model or the XML Document Object Model.
- XHTML requires quality code. This reduces the discrepancies between how different user agents, such as web browsers, render a web document, since many of these discrepancies are caused by incorrect or poorly formatted code.

Migrating from HTML to XHTML

XHTML works with browsers and other user agents that support HTML 4, as well as those that support XML. So you can switch to XHTML without excluding anyone or waiting for XML user agents to become more prevalent. Because XHTML is so similar to HTML, it's not difficult to switch from HTML to XHTML.

Web resources for XHTML

For more information about XHTML, see the World Wide Web Consortium (W3C) website, which contains the specification for XHTML 1.1 - Module-Based XHTML (<http://www.w3.org/TR/xhtml11/>) and XHTML 1.0 (<http://www.w3c.org/TR/xhtml1/>), as well as XHTML validator sites for web-based files (<http://validator.w3.org/>) and local files (<http://validator.w3.org/file-upload.html>).

About the XHTML code generated by Dreamweaver

Dreamweaver generates XHTML code, and cleans up existing XHTML code, in a way that meets many of the XHTML requirements automatically. For the few XHTML requirements that remain, Dreamweaver provides you with the tools that you need.

Note: Some of the requirements described in this section are also required in various versions of HTML.

The following table describes the XHTML requirements that Dreamweaver meets automatically.

XHTML requirement	To meet this requirement, Dreamweaver does this...
When the character encoding of a document is other than the default UTF-8, the document must have an XML declaration.	Adds an XML declaration to an XHTML document and specifies the encoding; for example: <code><?xml version="1.0" encoding="iso-8859-1" ?></code>
There must be a DOCTYPE declaration in the document prior to the root element, and the declaration must reference one of the three Document Type Definition (DTD) files for XHTML (strict, transitional or frameset).	Adds an XHTML DOCTYPE to an XHTML document: <code><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"></code> Or, if the XHTML document has a frameset: <code><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd"></code>
The root element of the document must be "html", and the html element must designate the XHTML namespace.	Adds the namespace attribute to the html element, as follows: <code><html xmlns="http://www.w3.org/1999/xhtml"></code>
A standard document must have the head, title, and body structural elements. A frameset document must have the head, title, and frameset structural elements.	Includes the head, title, and body elements in a standard document and, in a frameset document, includes the head, title, and frameset elements.
All elements in the document must nest properly: <code><p>This is a <i>bad example.</p></i></code> <code><p>This is a <i>good example.</i></p></code>	Generates correctly nested code and, when cleaning up XHTML, corrects nesting in code that was not generated by Dreamweaver.
All element and attribute names must be lowercase.	Forces HTML element and attribute names to be lowercase in the XHTML code that it generates, and when cleaning up XHTML, regardless of your tag and attribute case preferences.
Every element must have an end tag, unless it is declared in the DTD as EMPTY.	Inserts end tags in the code that it generates, and when cleaning up XHTML.
Empty elements must have an end tag, or the start tag must end with <code></></code> . For example, <code>
</code> is not valid; the correct form is <code>
</br></code> or <code>
</code> . Following are the empty elements: area, base, basefont, br, col, frame, hr, img, input, isindex, link, meta, and param. And for backwards-compatibility with browsers that are not XML-enabled, there must be a space before the <code>/></code> (for example, <code>
</code> , not <code>
</code>).	Inserts empty elements with a space before the closing slash in empty tags in the code that it generates, and when cleaning up XHTML.
Attributes can't be minimized; for example, <code><td nowrap></code> is not valid; the correct form is <code><td nowrap="nowrap"></code> . This affects the following attributes: checked, compact, declare, defer, disabled, ismap, multiple, noresize, noshade, nowrap, readonly, and selected.	Inserts full attribute-value pairs in the code that it generates, and when cleaning up XHTML. <i>Note:</i> If an HTML browser does not support HTML 4, it might fail to interpret these boolean attributes when they appear in their full form.
All attribute values must be quoted.	Quotes attribute values in the code that it generates, and when cleaning up XHTML.
The following elements must have an id attribute as well as a name attribute: a, applet, form, frame, iframe, img, and map. For example, <code>Introduction</code> is not valid; the correct form is <code>Introduction</code> or <code></code> <code>Introduction</code> .	Sets the name and id attributes to the same value, whenever the name attribute is set by a Property inspector, in the code that Dreamweaver generates, and when cleaning up XHTML.

XHTML requirement	To meet this requirement, Dreamweaver does this...
<p>For attributes with values of an enumerated type, the values must be lower case.</p> <p>An enumerated type value is a value from a specified list of allowed values; for example, the align attribute has the following allowed values: center, justify, left, and right.</p>	<p>Forces enumerated type values to be lowercase in the code that it generates, and when cleaning up XHTML.</p>
<p>All script and style elements must have a type attribute. (The type attribute of the script element has been required since HTML 4, when the language attribute was deprecated.)</p>	<p>Sets the type and language attributes in script elements, and the type attribute in style elements, in the code that it generates and when cleaning up XHTML.</p>
<p>All img and area elements must have an alt attribute.</p>	<p>Sets these attributes in the code that it generates and, when cleaning up XHTML, reports missing alt attributes.</p>

Using Cascading Style Sheets (CSS) with XHTML

For certain CSS styles, an HTML user agent (such as a web browser) may produce different visual or aural results than an XML user agent. Following are some rules to reduce these discrepancies:

- Use lowercase for element and attribute names in CSS style sheets.

Dreamweaver forces style element and attribute names to be lowercase in an XHTML document, regardless of your preferences, in the code that it generates and when cleaning up XHTML.
- HTML parsers infer the `tbody` element, but XML parsers do not. Therefore, if you define a `tbody` style, add the `tbody` element to your tables.
- CSS defines different conformance rules for HTML and XML documents; therefore, be aware that the HTML rules apply to an XHTML document delivered as HTML, and the XML rules apply to an XHTML document delivered as XML.

Note: In documents that specify the XHTML namespace, browsers and other user agents are expected to continue recognizing the class attribute, and the id attribute as an attribute of type ID. Therefore, you can continue to use the shorthand “.” syntax to reference a CSS class for a selector (for example, P.note); and use the shorthand “#” selector syntax, even if the user agent doesn’t read the DTD.

Making documents XHTML compliant

This section describes how to make a new document XHTML compliant, and how to make an existing HTML document XHTML compliant.

To create a new, XHTML-compliant document:

- 1 Choose File > New.

The New Document dialog box appears.
- 2 Choose a document type.
- 3 Select the Make Document XHTML Compliant option.
- 4 Click OK.

To create XHTML-compliant documents by default:

- 1 Choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X) and select the New Document category.
- 2 In the New Document category, choose a document type and select the Make Document XHTML Compliant option.
- 3 Click OK.

To make an existing HTML document XHTML compliant:

Open a document.

- For a document without frames, choose File > Convert > XHTML.
- For a document with frames, select a frame and choose File > Convert > XHTML. To convert the whole document, repeat this step for every frame and the frameset document.

Note: You can't convert an instance of a template, because it must be in the same language as the template on which it's based. For example, an instance of an XHTML template will always be XHTML; and an instance of an HTML template will always be HTML and can't be converted to XHTML or any other language.

Using the JavaScript debugger

The JavaScript debugger helps you isolate errors in your client-side JavaScript code. You can write code in Code view (or Code inspector), then run the debugger to check your code for syntax and logical errors. If the page has one or more syntax errors, the browser displays an error message; if your page contains a logical error, your page functions incorrectly but the browser doesn't display an error message.

You can use the JavaScript debugger with Microsoft Internet Explorer and Netscape Navigator 4.0 (not Netscape Navigator 6.0).

The debugger checks your code for syntax errors first, and then runs with the browser to help you identify any logical errors. If you have logical errors, you can use the JavaScript Debugger dialog box to examine variables and document properties while your program is running. You can set breakpoints (similar to alert statements) in your code to stop the execution of the program and display the values of JavaScript objects and properties in a variable list. You can also step to the next statement or step into a function call to see the variable values change.

This section covers the following topics:

- “Running the JavaScript debugger” on page 210
- “Finding syntax errors” on page 211
- “Finding and fixing logical errors” on page 212
- “Setting breakpoints” on page 212
- “Stepping through code” on page 213
- “Watching and editing variable values” on page 213

Running the JavaScript debugger

After you write your JavaScript code, you can start the JavaScript debugger to check for errors. The debugger checks for syntax errors first, then opens your page in the browser so you can check for logical errors.

To start debugging:

- 1 Choose File > Debug in Browser, then select a browser from the list.

In Windows, choose Internet Explorer or Netscape Navigator. In Macintosh, choose Netscape.

If the debugger finds a syntax error, it stops and lists it in the JavaScript Syntax Errors window. For more information, see “Finding syntax errors” on page 211.

- 2 If you are using Netscape Navigator, click OK in the debugger warning box that appears, then click Grant in the Java Security dialog box.

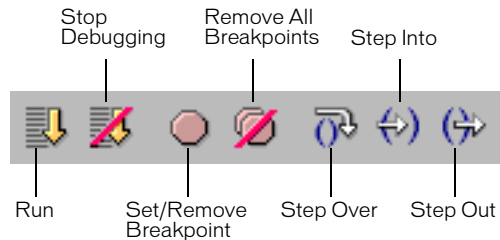
Note: If you have already accepted a Macromedia Security Certificate, the Java Security dialog box may not appear.

- 3 (Windows only) If you are using Internet Explorer, click Yes in the Java Security dialog box, and click OK in the debugger warning box that appears.

Note: The security dialog box is necessary because the debugger uses an Internet protocol to connect with the browser—it doesn't actually make a network connection or connect to any Internet servers.

The JavaScript Debugger dialog box appears with the browser window. The debugger stops automatically at the first line of code.

The following figure shows the toolbar in the JavaScript Debugger dialog box:



To run the debugger:

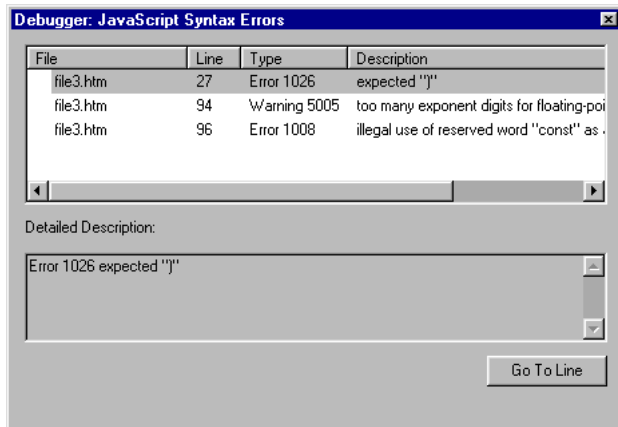
At the top of the JavaScript Debugger dialog box, click the Run button.

To stop the debugger:

At the top of the JavaScript Debugger dialog box, click the Stop Debugging button. The JavaScript Debugger dialog box closes.

Finding syntax errors

If the debugger finds syntax errors in a page, it stops and lists the errors in the JavaScript Syntax Errors dialog box, as shown in the following figure:



To view the error description:

Select an error in the JavaScript Syntax Errors dialog box. A description of the error appears in the Detailed Description box.

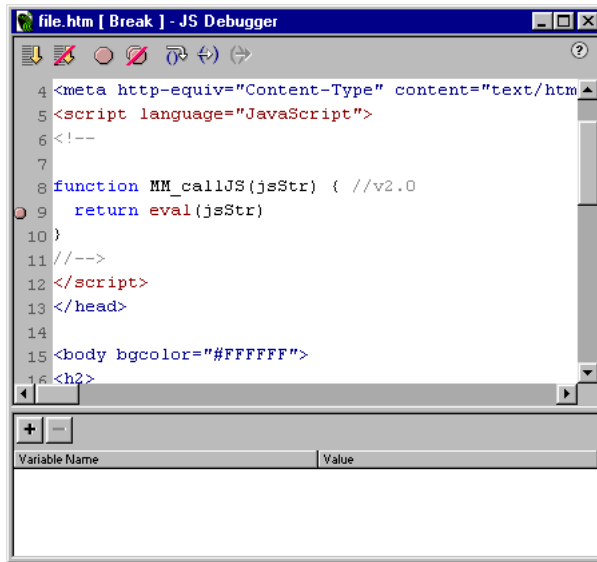
To go to the selected error in your code, do one of the following:

- Double-click the error.
- Click Go To Line.

The code is highlighted in Code view, or if Code view is not open, in the Code inspector.

Finding and fixing logical errors

When you start the debugger, the JavaScript Debugger dialog box opens and stops the execution of the browser in the first line of your code. You can set additional breakpoints at any line of code. The debugger stops execution of the code at each breakpoint, giving you the opportunity to view the values of JavaScript objects and properties in the variable list pane.



After the debugger stops at a breakpoint, you can step through your code (execute one statement at a time). This lets you see if the program executes as it should. The debugger can even step into linked code. For example, if your code contains a link to a source file, the debugger steps into the source file and displays it in the JavaScript Debugger dialog box. As you step through the code, you can watch as the values of your variables change through your program.

Setting breakpoints

A breakpoint marks a spot in the code where you want the program execution to stop. When you set a breakpoint, it's marked with a small red dot in the left margin of the JavaScript Debugger dialog box. When the program stops executing at that breakpoint, a small arrow appears over the dot, and you can examine the objects and properties that exist at that point. This lets you quickly pinpoint the source of the bug in the JavaScript code.

Breakpoints can only be set in the JavaScript code (between `script` tags) or on a line with an event handler. If you set a breakpoint elsewhere, Dreamweaver will automatically set the breakpoint on the next valid line of code (or place the insertion point there, if the line already has a breakpoint set on it). If there are no valid lines on which a breakpoint can be set, you will hear a beep.

To set a breakpoint:

- 1 In the JavaScript Debugger dialog box, place the insertion point in the line where you want the breakpoint.
- 2 In the toolbar, click the Set/Remove Breakpoint button.
- 3 To remove the breakpoint, click Set/Remove Breakpoint again.

To remove all breakpoints, do one of the following:

- At the top of the JavaScript Debugger dialog box, click the Remove All Breakpoints button.
- In the Code view (or Code inspector), choose Edit > Remove All Breakpoints.

Stepping through code

You can “step through” your code to execute your statements one at a time and see the results. For example, you can step over an `if` condition and see if the program stops at the first line within the conditional statement or at the next executable line after the `if` statement.

When the debugger stops at a statement with a function call, you can check the function to make sure it executes correctly. If the function is correct, you can step out of it to allow the debugger to run until the function returns. The program will stop again at the next statement after the function call location. If you attempt to step into a statement that contains a nonstandard JavaScript function, the JavaScript debugger steps over the statement instead.

To step over a statement:

At the top of the JavaScript Debugger dialog box, click the Step Over button.

When the program stops at any statement (including those with a function call), you can step over that statement to continue and pause before the next statement.

To step into a function:

At the top of the JavaScript Debugger dialog box, click the Step In button.

To step out of a function:

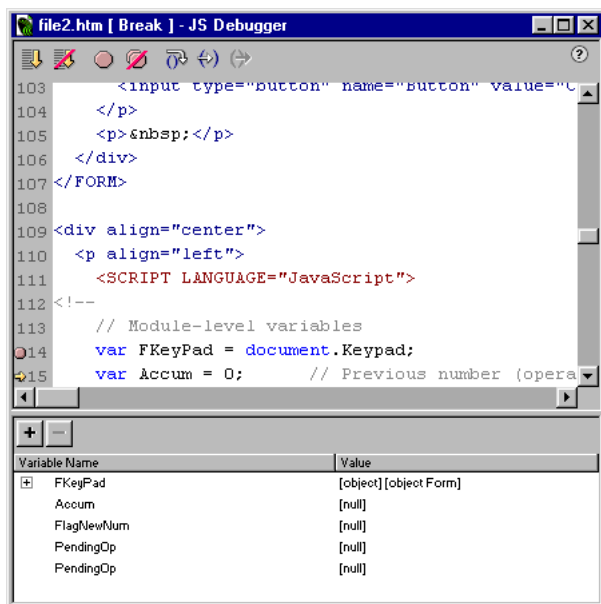
At the top of the JavaScript Debugger dialog box, click the Step Out button.

You can only use Step Out when the debugger is within a user-defined function. Stepping out causes the remaining statements in the function definition to be executed. The debugger pauses at the next statement.

Watching and editing variable values

To check the values of your variables as you step through the code, you use the variable list pane located at the bottom of the JavaScript Debugger dialog box. When you add a variable, its name appears in the left column; the right column displays its current value when the program stops execution at a breakpoint, or after you have stepped into the code.

The following figure shows the current value for several variables:



To add a variable to your variable list, do one of the following:

- Select the name of the variable in the code portion of the JavaScript Debugger dialog box. Click the plus (+) button and press Enter.
- Click the plus (+) button, type the name of the variable you want to watch, and press Enter.

The values appear next to each variable as you step through the code. If the variable is an object with properties, you can expand the variable to show its properties and values. To expand the variable, click the plus (+) button (Windows) or triangle button (Macintosh) next to it in the list. The expanded variable collapses automatically whenever you step through the code.

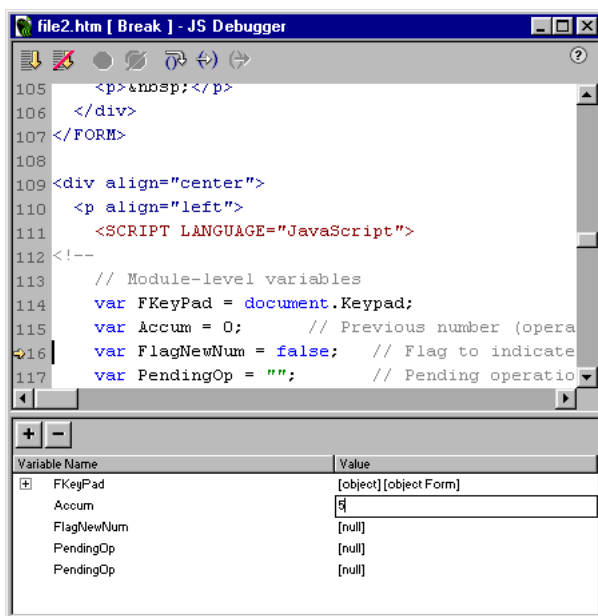
To remove a variable from the list:

- 1 Select the item in the variable list.
- 2 Click the minus (-) button.

To edit a value:

- 1 Select the item in the variable list.
- 2 Click the value in the value list.

3 In the text box that appears, type a new value.



Using the ColdFusion debugger

While developing Macromedia ColdFusion applications, you can configure ColdFusion to display information in a browser to help you debug the application. For example, if a page contains an error, ColdFusion displays possible causes for the error at the bottom of a ColdFusion page when you open the page in a browser. If you're a ColdFusion developer using Macromedia ColdFusion MX as your Dreamweaver testing server, you can view this information and fix the page without leaving Dreamweaver.

Note: This feature is not supported on the Macintosh. Macintosh developers can use Preview in Browser (F12) to open a ColdFusion page in a separate browser. If the page contains errors, information about the possible causes for the error appears at the bottom of the page.

Before starting, make sure debugging settings are enabled in ColdFusion Administrator. For more information, see ColdFusion help in Dreamweaver (Help > Using ColdFusion). Also, make sure your Dreamweaver testing server is running ColdFusion MX. For more information on the Dreamweaver testing server, see "Specifying a folder to process dynamic pages".

To debug a ColdFusion page:

- 1** Open the ColdFusion page in Dreamweaver.
- 2** Click the Server Debug icon on the document toolbar (the icon with the globe and lightning bolt), or select View > Server Debug.

Dreamweaver requests the file from the ColdFusion MX server and displays it in an internal Internet Explorer browser window. If the page contains errors, possible causes for the errors appear at the bottom of the page.

At the same time, a Server Debug panel opens. The panel provides a large amount of useful information, such as all the pages the server processed to render the page, all the SQL queries executed on the page, and all the server variables and their values, if any. The panel also provides a summary of execution times.

- 3** If an Exceptions category appears in the Server Debug panel, click the plus (+) icon to expand the category.

The Exceptions category appears if the server encountered a problem or problems with the page. Expand the category to find out more about the problem.

- 4** In the Location column of the Server Debug panel, click on the page's URL to open the page in Code view and fix it.

If Dreamweaver can locate the page, the page opens with the problem line or lines highlighted. If Dreamweaver cannot locate the page, it asks you for the location.

- 5** Fix the error, save the file to the server, and click browse.

Dreamweaver renders the page in the internal browser again and updates the Server Debug panel. If there are no more problems with the page, the Exceptions category does not reappear in the panel.

- 6** To leave debug mode, switch to Code view (View > Code) or Design view (View > Design).

To ensure the debug information is refreshed every time a page is displayed in the internal browser, make sure Internet Explorer checks for newer versions of the file every time the file is requested. In Internet Explorer, choose Tools > Internet Options, select the General tab, and click the Settings button in the Temporary Internet Files area. In the Settings dialog box, select the "Every visit to page" option.

CHAPTER 15

Editing Code in Design View

Although Macromedia Dreamweaver MX allows you to visually create and edit web pages without worrying about the underlying source code, there are times when you might need to edit the code for greater control or to troubleshoot your web page.

For example, suppose you select text on your document and apply a new font, but it only affects half the sentence. By looking at the code, you discover that the closing `` tag is in the middle of the sentence. Once you move `` to the end of your sentence, the problem is fixed.

This chapter is designed for people who prefer to work in Design view, but who also want quick access to the code. Dreamweaver lets you edit code while working in Design view.

The chapter contains the following topics:

- “Editing code with the Property inspector” on page 217
- “Editing code with a tag editor” on page 218
- “Editing code with the Quick Tag Editor” on page 218
- “Editing code with the tag selector” on page 221
- “Editing scripts” on page 221
- “Working with server-side includes” on page 223

Editing code with the Property inspector

You can use the Property inspector to inspect and edit the attributes of text or of objects on your page.

To use the Property inspector:

- 1 Click in text or select an object on the page.

The Property inspector for the text or object appears below the Document window. If the Property inspector is not visible, select Window > Properties.

- 2 Make changes to the attributes in the Property inspector.

Editing code with a tag editor

You can use a tag editor to inspect and edit the attributes of objects on your page.

To use a tag editor:

- 1 Right-click (Windows) or Control-click (Macintosh) the object on the page and choose Edit Tag from the pop-up menu.

The tag editor for that object appears.

- 2 Make changes to the object's attributes and click OK.

Editing code with the Quick Tag Editor

You can use the Quick Tag Editor to quickly inspect and edit HTML tags without leaving Design view. To open the Quick Tag Editor, press Control+T (Windows) or Command+T (Macintosh).

This section contains the following topics:

- “About the Quick Tag Editor” on page 218
- “Inserting an HTML tag with the Quick Tag Editor” on page 218
- “Editing an HTML tag with the Quick Tag Editor” on page 219
- “Applying HTML to a selection with the Quick Tag Editor” on page 219
- “Using the hints menu” on page 220

About the Quick Tag Editor

The Quick Tag Editor has three modes:

- **Insert HTML** mode to insert new HTML code
- **Edit Tag** mode to edit an existing tag
- **Wrap Tag** mode to wrap a new tag around the current selection

The mode in which the Quick Tag Editor opens depends on the current selection in Design view. In all three modes, the basic procedure for using the Quick Tag Editor is the same: open the editor, enter or edit tags and attributes, and then close the editor.

You can cycle through the various modes by pressing Control+T (Windows) or Command+T (Macintosh) while the Quick Tag Editor is active.

If you use any invalid HTML in the Quick Tag Editor, Dreamweaver attempts to correct it for you by inserting closing quotation marks and closing angle brackets where needed.

To perform more extensive HTML editing, use the Code view. For more information, see “Coding in Dreamweaver” on page 189.

Inserting an HTML tag with the Quick Tag Editor

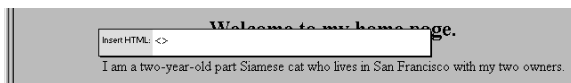
You can use the Quick Tag Editor to insert an HTML tag in your document.

To insert an HTML tag with the Quick Tag Editor:

- 1 In Design view, click in the page to place the insertion point where you want to insert code.

- 2 Press Control+T (Windows) or Command+T (Macintosh).

The Quick Tag Editor opens in Insert HTML mode.



- 3 Enter the HTML tag and press Enter.

The tag is inserted into your code.

- 4 Press Escape to exit without making any changes.

Editing an HTML tag with the Quick Tag Editor

You can use the Quick Tag Editor to edit an HTML tag in your document.

To edit an HTML tag with the Quick Tag Editor:

- 1 Select an object in Design view.

You can also select the tag you want to edit from the tag selector at the bottom of the Document window. For more information, see “Editing code with the tag selector” on page 221.

- 2 Press Control+T (Windows) or Command+T (Macintosh).

The Quick Tag Editor opens in Edit Tag mode.

- 3 Enter new attributes, edit existing attributes, or edit the tag’s name.

- 4 Press Tab to move forward from one attribute to the next; press Shift+Tab to move back.

By default, changes are applied to the document when you press Tab or Shift+Tab. To disable the automatic updates, select Edit > Preferences > Quick Tag Editor or Dreamweaver > Preferences > Quick Tag Editor (Mac OS X). The Quick Tag Editor Preferences dialog box appears. Deselect the Apply Changes Immediately While Editing option and click OK. For more information, click the Help button in the dialog box.

- 5 To close the Quick Tag Editor and apply all the changes, press Enter.

- 6 To exit without making any further changes, press Escape.

Applying HTML to a selection with the Quick Tag Editor

You can use the Quick Tag Editor to wrap a selection in your document with opening and closing HTML tags.

To apply HTML to a selection with the Quick Tag Editor:

- 1 Select unformatted text or an object in Design view.

If you select text or an object that includes an opening or closing HTML tag, the Quick Tag Editor will open in Edit Tag mode.

- 2 Press Control+T (Windows) or Command+T (Macintosh), or click the Quick Tag Editor button in the Property inspector.

The Quick Tag Editor opens in Wrap Tag mode.

- 3 Enter a single opening tag, such as `<font="verdana">`, and press Enter.

The tag is inserted at the beginning of the current selection, and a matching closing tag is inserted at the end.

- 4 Press Escape to exit without making any changes.

Using the hints menu

While in the Quick Tag Editor, you can access an attributes hint menu that lists all the valid attributes of the tag you are editing or inserting. If Dreamweaver doesn't recognize the tag you're editing, the hints menu contains all the attributes that Dreamweaver recognizes for any tag.

You can also disable the hints menu or adjust the delay before the menu pops up in the Quick Tag Editor.

To see a list of valid attributes for a tag, pause for a few seconds while editing an attribute name in the Quick Tag Editor. A hints menu appears, listing all the valid attributes for the tag you're editing.

Use the hints menu as follows:

- As you begin to type an attribute name, the hints menu scrolls to highlight the first attribute name that starts with the letters you typed.
- To move the highlighting up and down in the menu, use the Up and Down Arrow keys or the scroll bar.
- To choose the highlighted attribute name, press Enter. Alternatively, double-click an attribute name in the menu.
- To dismiss the hints menu without selecting an entry, press Escape or simply continue typing.

If you pause while entering or editing a tag name, a similar hints menu appears, listing tag names instead of attribute names.

To disable the hints menu or change the delay before it appears:

- 1 Choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X) and select Quick Tag Editor. The Quick Tag Editor Preferences dialog box appears.
- 2 To disable the hints menu, deselect the Enable Tag Hints option.
- 3 To change the delay before the menu pops up, adjust the Delay slider.
For more information, click the Help button in the dialog box.
- 4 Click OK.

Editing code with the tag selector

You can use the tag selector to select, edit, or remove tags without leaving Design view. The tag selector is located in the status bar at the bottom of the Document window and shows a series of tags, as follows.

```
<body> <form> <table> <tr>
```

To edit a tag:

- 1 Click in the document.

The tags that apply at the insertion point appear in the tag selector.

- 2 Right-click (Windows) or Control-click (Macintosh) a tag in the tag selector.

A pop-up menu appears.

- 3 Select Edit Tag from the menu.

The Quick Tag Editor appears. For more information, see “Editing an HTML tag with the Quick Tag Editor” on page 219.

To remove a tag:

- 1 Click in the document.

The tags that apply at the insertion point appear in the tag selector.

- 2 Right-click (Windows) or Control-click (Macintosh) a tag in the tag selector.

A pop-up menu appears.

- 3 Select Remove Tag from the menu.

To select an object represented by a tag:

- 1 Click in the document.

The tags that apply at the insertion point appear in the tag selector.

- 2 Click a tag in the tag selector.

The object represented by the tag is selected on the page.

Tip: Use this technique to select individual table rows (tr tags) or cells (td tags).

Editing scripts

You can work with client-side JavaScripts and VBScripts in both Code and Design view. This section covers the following topics:

- “Writing a client-side script in Design view” on page 222
- “Linking to an external script file” on page 222
- “Editing a script in Design view” on page 222

Writing a client-side script in Design view

You can write a JavaScript or VBScript for your page without leaving Design view. Before starting, make sure Dreamweaver displays script markers on the page. To display script markers, select View > Visual Aids > Invisible Elements.

To insert a client-side script in Design view:

- 1 Place the insertion point where you want the script.
- 2 Select Insert > Script Objects > Script.
The Script dialog box appears.
- 3 Complete the dialog box and click OK.

For more information, click the Help button in the dialog box.

Linking to an external script file

You can create a link in your document to an external script file without leaving Design view. Before starting, make sure Dreamweaver displays script markers on the page. To display script markers, select View > Visual Aids > Invisible Elements.

To link to an external script file:

- 1 Place the insertion point where you want the script.
- 2 Select Insert > Script Objects > Script.
The Script dialog box appears.
- 3 Click OK without typing anything in the Content box.
- 4 Select the script marker in Design view of the Document window.
- 5 In the Property inspector, click the folder icon to browse to and select the external script file, or type the filename in the Source box.

Editing a script in Design view

You can edit a script without leaving Design view. Before starting, make sure Dreamweaver displays script markers on the page. To display script markers, select View > Visual Aids > Invisible Elements.

To edit the script in Design view:

- 1 Select the script marker.
- 2 In the Property inspector, click the Edit button.
The script appears in the Script Properties dialog box.

If you linked to an external script file, the file automatically opens in Code view, where you can make your edits.

Note: If there is code between the script tags, the Script Properties dialog box will open even if there is also a link to an external script file.

- 3 Make edits to the script and click OK.

For more information, click the Help button on the Script Properties dialog box.

Working with server-side includes

Server-side includes are instructions to the web server to include a specified file in a web page before serving the page to the browser. You can use Dreamweaver to insert server-side includes in your pages, edit the includes, or preview pages containing includes.

This section covers the following topics:

- “About server-side includes” on page 223
- “Inserting a server-side include” on page 224
- “Editing the contents of a server-side include” on page 224

About server-side includes

When you open a document that’s on a web server, the server processes the include instructions and creates a new document in which the include instructions are replaced by the contents of the included file. The server then sends this new document to your browser. When you open a local document directly in a browser, however, there’s no server to process the include instructions in that document, so the browser opens the document without processing those instructions, and the file that’s supposed to be included doesn’t appear in the browser. It can thus be difficult, without using Dreamweaver, to look at local files and see them as they’ll appear to visitors after you’ve put them on the server.

With Dreamweaver, though, you can preview documents just as they’ll appear after they’re on the server, both in the Design view and when you preview in a browser.

Placing a server-side include in a document inserts a reference to an external file; it doesn’t insert the contents of the specified file in the current document. Dreamweaver displays the contents of the external file in Design view, making it easier to design pages.

You cannot edit the included file directly in a document. To edit the contents of a server-side include, you must directly edit the file that you’re including. Any changes to the external file are automatically reflected in every document that includes it.

There are two types of server-side includes: Virtual and File. Choose which to use depending on what type of web server you use:

- If your server is an Apache web server, choose Virtual. (This is the default choice in Dreamweaver.) In Apache, Virtual works in all cases, while File works only in some cases.
- If your server is a Microsoft Internet Information Server (IIS), choose File. (Virtual works with IIS only in certain specific circumstances.) Unfortunately, IIS won’t allow you to include a file in a folder above the current folder in the folder hierarchy, unless special software has been installed on the server. If you need to include a file from a folder higher in the folder hierarchy on an IIS server, ask your system administrator if the necessary software is installed.
- For other kinds of servers, or if you don’t know what kind of server you’re using, ask your system administrator which option to use.

Some servers are configured to examine all files to see if they contain server-side includes; other servers are configured to examine only files with a particular file extension, such as .shtml, .shtm, or .inc. If a server-side include isn't working for you, ask your system administrator if you need to use a special extension in the name of the file that uses the include. (For example, if the file is named canoe.html, you may have to rename it to canoe.shtml.) If you want your files to retain .html or .htm extensions, ask your system administrator to configure the server to examine all files (not just files with a certain extension) for server-side includes. Parsing a file for server-side includes takes a little extra time, though, so pages that the server parses are served a little more slowly than other pages; some system administrators therefore won't provide the option of parsing all files.

Inserting a server-side include

You can use Dreamweaver to insert server-side includes in your page.

To insert a server-side include:

- 1 Choose Insert > Script Objects > Server-Side Include.
- 2 In the dialog box that appears, browse to and select a file.

To change which file is included:

- 1 Select the server-side include in the Document window.
- 2 Open the Property inspector (Window > Properties).
- 3 Do one of the following:
 - Click the folder icon and browse to and select a new file to include.
 - In the text box, type the path and filename of the new file to include.

Editing the contents of a server-side include

You can use Dreamweaver to edit server-side includes. To edit the content associated with the included file, you must open the file.

To edit a server-side include:

- 1 Select the server-side include in either Design view or Code view, and click Edit in the Property inspector.

The included file opens in a new Document window.

- 2 Edit the file, then save it.

The changes are immediately reflected in the current document and in any subsequent document you open that includes the file.

Part IV

Designing the Page Layout

Use the visual design tools in Dreamweaver to create sophisticated page layouts.

This part contains the following chapters:

- Chapter 16, “Presenting Content with Tables”
- Chapter 17, “Laying Out Pages in Layout View”
- Chapter 18, “Using Frames”

CHAPTER 16

Presenting Content with Tables

Tables are a powerful tool for presenting tabular data and for laying out text and graphics on an HTML page. A table consists of one or more rows; each row consists of one or more cells. Although columns aren't usually explicitly specified in HTML code, Dreamweaver allows you to manipulate columns as well as rows and cells.

After you create a table you can easily modify both its appearance and structure. You can do any of the following:

- Add content
- Add, delete, split, and merge rows and columns
- Modify table, row, or cell properties to add color and alignment
- Copy and paste cells

Note: Many designers use tables to lay out web pages. Dreamweaver provides two ways to view and manipulate tables: Standard view, in which tables are presented as a grid of rows and columns, and Layout view, which allows you to draw, resize, and move boxes on the page while still using tables for the underlying structure (see Chapter 17, “Laying Out Pages in Layout View,” on page 241).

This chapter contains these sections:

- “Inserting a table” on page 228
- “Adding content to a table cell” on page 228
- “Importing tabular data” on page 229
- “Selecting table elements” on page 229
- “Formatting tables and cells” on page 231
- “Resizing tables” on page 233
- “Changing column widths and row heights” on page 234
- “Adding and removing rows and columns” on page 234
- “Nesting tables” on page 237
- “Cutting, copying, and pasting cells” on page 237
- “Sorting tables” on page 239
- “Exporting table data” on page 239

Inserting a table

Use the Insert bar or the Insert menu to create a new table. For information on creating accessible tables in Dreamweaver, see “Authoring for accessibility” on page 337.

To insert a table:

- 1 In the Design view of the Document window, place the insertion point where you want the table to appear.

Note: If your document contains no content, then the only place you can place the insertion point is at the beginning of the document.

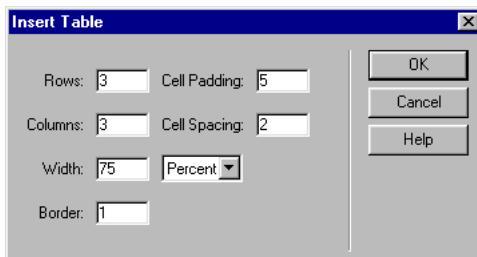
- 2 Do one of the following:

- Click the Table button in the Common category of the Insert bar.
- Choose Insert > Table.

The Insert Table dialog box appears.

- 3 Enter new values as desired.

For more information, click the Help button in the dialog box.



Adding content to a table cell

You can add text and images to table cells the same way that you add text and images outside of a table.

For more information, see Chapter 19, “Inserting and Formatting Text,” on page 271 and Chapter 20, “Inserting Images,” on page 297.

When you add or edit content in your table, you can save time by using the keyboard to navigate the table.

To move from one cell to another using the keyboard, do any of the following:

- Press Tab to move to the next cell.
Pressing Tab in the last cell of a table automatically adds another row to the table.
- Press Shift+Tab to move to the previous cell.
- Press the arrow keys to move up, down, left, or right.

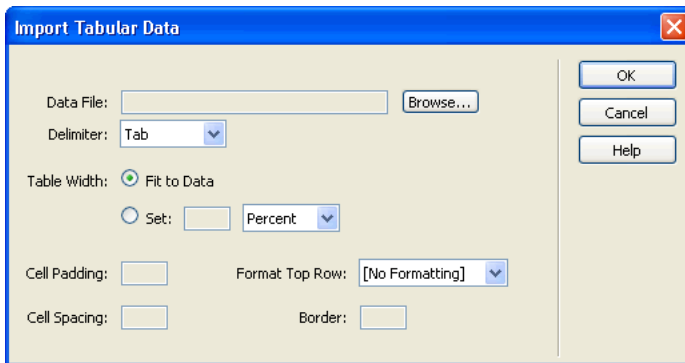
Importing tabular data

Tabular data that has been created in another application (such as Microsoft Excel) and saved in a delimited text format (with items separated by tabs, commas, colons, semicolons, or other delimiters) can be imported into Dreamweaver and formatted as a table.

To import table data:

- 1 Do one of the following:
 - Choose File > Import > Tabular Data.
 - Choose Insert > Table Objects > Import Tabular Data.

The Import Tabular Data dialog box appears.



- 2 In the dialog box, enter information about the file containing your data.

For more information, click the Help button in the dialog box.

Related topic

Chapter 34, “Displaying Database Records,” on page 527

Selecting table elements

You can select an entire table, row, or column at once. You can also select a contiguous block of cells within a table. After you’ve selected a table or cells, you can do the following:

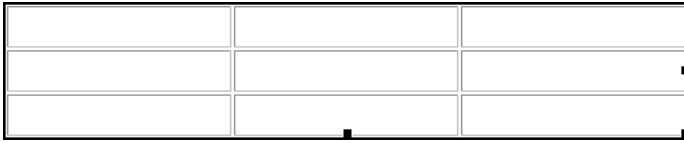
- Modify the appearance of selected cells or of the text contained in them. See “Formatting tables and cells” on page 231.
- Copy and paste cells. See “Cutting, copying, and pasting cells” on page 237.

You can also select multiple nonadjacent cells in a table and modify the properties of those cells. You cannot copy or paste sets of nonadjacent cells.

To select an entire table, do one of the following:

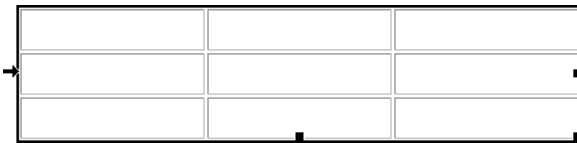
- Click the upper left corner of the table or click anywhere on the right or bottom edge.
- Click in a table cell, and then select the `table` tag in the tag selector at the lower left corner of the Document window.
- Click in a table cell, and then choose Modify > Table > Select Table.

- Click in a table cell, and then select the `table` tag in the Tag Inspector. Selection handles appear on the selected table's lower and right edges.



To select rows or columns:

- 1 Position the pointer to point to the left edge of a row or the top edge of a column.
- 2 When the pointer changes to a selection arrow, click to select a row or column, or drag to select multiple rows or columns.



To select a single cell, do one of the following:

- Click in the cell, and then select the `td` tag in the tag selector at the lower left corner of the Document window.
- Click in the cell, and then choose Edit > Select All.
Tip: Choose Edit > Select All again when a cell is selected to select the entire table.
- Click in a table cell, and then select the `td` tag in the Tag Inspector.

To select a line or a rectangular block of cells, do one of the following:

- Drag from a cell to another cell.
- Click in one cell, and then Shift-click another cell.

All of the cells within the linear or rectangular region defined by the two cells are selected.



To select nonadjacent cells:

Control-click (Windows) or Command-click (Macintosh) the cells, rows, or columns you want to select.

If each cell, row, or column you Control-click or Command-click isn't already selected, it's added to the selection. If it is already selected, it's removed from the selection.

Formatting tables and cells

You can change the appearance of tables by setting properties for the table and its cells or by applying a preset design to the table. See “Viewing and setting table properties” on page 231, “Viewing and setting cell, row, and column properties” on page 232, and “Using a design scheme to format a table” on page 232.

To format text in tables, you can apply formatting to selected text or use styles. See Chapter 19, “Inserting and Formatting Text,” on page 271.

About conflicts in table formatting

When formatting tables in Design view, you can set properties for the entire table or for selected rows, columns, or cells in the table. When a property, such as background color or alignment, is set to one value for the whole table and another value for individual cells, cell formatting takes precedence over row formatting, which in turn takes precedence over table formatting.

The order of precedence for table formatting is:

- 1 Cells
- 2 Rows
- 3 Table

For example, if you set the background color for a single cell to blue, and then set the background color of the entire table to yellow, the blue cell does not change to yellow, since cell formatting takes precedence over table formatting.

Note: When you set properties on a column, Dreamweaver changes the attributes of the `td` tag corresponding to each cell in the column.

Viewing and setting table properties

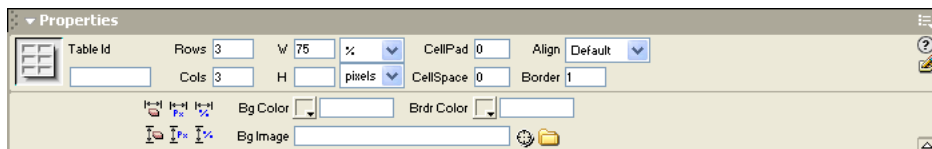
When a table is selected, the Property inspector lets you view and change table properties.

To view table properties:

- 1 Select the table.
- 2 Choose **Window > Properties** to open the Property inspector.

To view all table properties:

If the Property inspector isn't expanded, click the expander arrow in the lower right corner to see all properties.



To format a table in the Property inspector:

- 1 Select a table. For more information, see “Selecting table elements” on page 229.
- 2 Choose **Window > Properties** to open the Property inspector.

3 Change the table's formatting by setting properties.

For more information, click the Help button in the Property inspector.

Related topic

“Using a design scheme to format a table” on page 232

Viewing and setting cell, row, and column properties

When a cell or a set of cells is selected, the Property inspector allows you to view and change the cell properties.

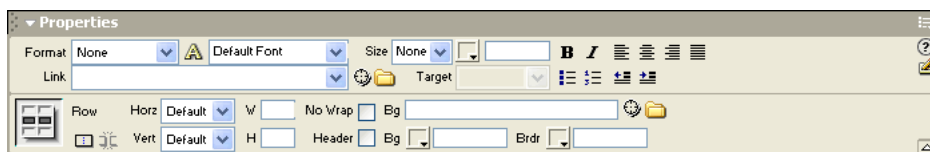
Note: When you set properties on a column, Dreamweaver changes the attributes of the `td` tag corresponding to each cell in the column. When you set certain properties for a row, however, Dreamweaver changes the attributes of the `tr` tag rather than changing the attributes of each `td` tag in the row. When you're applying the same format to all the cells in a row, applying the format to the `tr` tag produces cleaner, more concise HTML code.

To view table-element properties:

- 1 Select a cell or a set of cells in a row or column.
- 2 Choose Window > Properties to open the Property inspector.

To view all cell, row, or column properties:

If the Property inspector isn't expanded, click the expander arrow in the lower right corner to see all properties.



To format table elements in the Property inspector:

- 1 Select a cell, a row, or a column. For more information, see “Selecting table elements” on page 229.
- 2 Choose Window > Properties to open the Property inspector.
- 3 Change the table element's formatting by setting properties.

For more information, click the Help button in the Property inspector.

Related topic

“Selecting table elements” on page 229

Using a design scheme to format a table

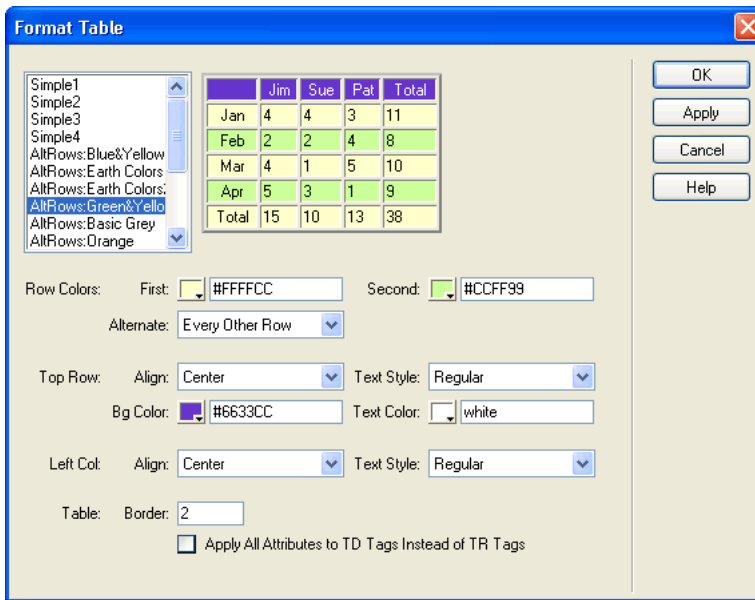
Use the Format Table command to quickly apply a preset design to a table. You can then select options to further customize the design.

Note: Only simple tables can be formatted using preset designs. You can't use these designs to format tables that contain merged cells (`colspan` or `rowspan`), column groups, or other unusual formatting that keeps the table from being a simple rectangular grid of cells.

To use a preset table design:

- 1 Select a table, and then choose Commands > Format Table.

The Format Table dialog box appears.



- 2 Customize options as desired.

For more information, click the Help button in the dialog box.

- 3 Click Apply or OK to format your table with the selected design.

Resizing tables

You can resize an entire table or individual rows and columns. When you resize an entire table, all of the cells in the table change size proportionately.

Note: If a table's cells have explicit widths or heights specified, resizing the table changes the visual size of the cells in the Document window but does not change the specified widths and heights of the cells.

To resize a table:

- 1 Select the table.
- 2 Do one of the following:
 - To resize the table horizontally, drag the selection handle on the right.
 - To resize the table vertically, drag the selection handle on the bottom.
 - To resize in both dimensions, drag the selection handle at the lower right corner.

Related topics

“Selecting table elements” on page 229

Changing column widths and row heights

You can change the width of a column or the height of a row by using the Property inspector or by dragging the borders of the column or row. You can also change cell widths and heights directly in the HTML code using Code view.

If you have trouble resizing, you can clear the column widths or row heights and start over.

Tip: You can specify widths and heights as either pixels or percentages, and you can convert from pixels to percentages and back. For more information, see “Viewing and setting table properties” on page 231.

To resize a column or a row visually, do one of the following:

- To change a column’s width, select the column, and then drag the right border of the column.
If the column is not the rightmost column of the table, the adjacent column’s width also changes, keeping the same overall table width. If the column is the rightmost column, the whole table’s width changes, and all of the columns grow wider or narrow proportionately.
- To change a row’s height, select the row, and then drag the lower border of the row.

To set a column’s width or a row’s height using the Property inspector:

- 1 Select the column or row.
- 2 Choose Window > Properties to open the Property inspector.
- 3 Change the table element’s formatting by setting properties.

For more information, click the Help button in the Property inspector.

To clear all set widths or heights:

- 1 Select the table.
- 2 Do one of the following:



- To clear all specified widths, choose Modify > Table > Clear Cell Widths or click the Clear Column Widths button in the Property inspector.



- To clear all specified heights, choose Modify > Table > Clear Cell Heights or click the Clear Row Heights button in the Property inspector.

Related topics

“Selecting table elements” on page 229

“Resizing tables” on page 233

“Coding in Dreamweaver” on page 189

Adding and removing rows and columns

To add and remove rows and columns, use the commands in the Modify > Table submenu.

Tip: Pressing Tab in the last cell of a table automatically adds another row to the table.

To add rows or columns:

- 1 Click in a cell.
- 2 Do one of the following:
 - To add a row above the current cell, choose Modify > Table > Insert Row.

- To add a column to the left of the current cell, choose **Modify > Table > Insert Column**.
- To add multiple rows or columns at once, or to add a row below the current cell or a column to the right of it, choose **Modify > Table > Insert Rows or Columns**.

The Insert Rows or Columns dialog box appears.

- 3** If you chose Insert Rows or Columns, enter the necessary information in the dialog box, then click **OK**.

For more information, click the Help button in the dialog box.

To delete a row or column:

- 1** Click in a cell within the row or column you want to delete.

- 2** Do one of the following:

- To delete a row, choose **Modify > Table > Delete Row**.
- To delete a column, choose **Modify > Table > Delete Column**.

Tip: Alternatively, you can select a complete row or column and then choose **Edit > Clear** or press **Delete**; the entire row or column is removed from the table.

To add or delete rows or columns using the Property inspector:

- 1** Select the entire table.

- 2** In the Property inspector, do one of the following:

- Increase or decrease the Rows value to add or delete rows.

Dreamweaver adds and removes rows at the bottom of the table.

- Increase or decrease the Cols value to add or delete columns.

Dreamweaver adds and removes columns at the right side of the table.

Note: Dreamweaver does not warn you if you are deleting rows and columns that contain data.

Related topics

“Selecting table elements” on page 229

“Inserting a repeating table” on page 444

“Displaying multiple behaviors” on page 533

Splitting and merging cells

Use the Property inspector or the commands in the **Modify > Table** submenu to split or merge cells. You can merge any number of adjacent cells—as long as the entire selection is a line or a rectangle of cells—to produce a single cell that spans several columns or rows. You can split a cell into any number of rows or columns, regardless of whether it was previously merged.

Dreamweaver automatically restructures the table (adding any necessary **COLSPAN** or **ROWSPAN** attributes) to create the specified arrangement.

In the following illustration, the cells in the middle of the first two rows have been merged so that they span two rows.

As an alternative approach to merging and splitting cells, Dreamweaver also provides tools for increasing and decreasing the number of rows or columns spanned by a cell.

To merge two or more cells in a table:

- 1 Select the cells. The selected cells must be contiguous and in the shape of a rectangle.

In the following illustration, the selection is a rectangle of cells, so the cells can be merged.

Location Name	City	State or Country
Baltimore-Washington International	Baltimore	MD
Cairo International	Cairo	Egypt
Canberra	Canberra	Australia
Cairns	Cairns	Queensland
Cape Town Airport	Cape Town	South Africa

In the following illustration, the selection is not a rectangle, so the cells can't be merged.

Location Name	City	State or Country
Baltimore-Washington International	Baltimore	MD
Cairo International	Cairo	Egypt
Canberra	Canberra	Australia
Cairns	Cairns	Queensland
Cape Town Airport	Cape Town	South Africa

- 2 Choose **Modify > Table > Merge Cells**, or click the **Merge Cells** button in the Property inspector.

The contents of the individual cells are placed in the resulting merged cell. The properties of the first cell selected are applied to the merged cell.

To split a cell:

- 1 Click in the cell.
- 2 Choose **Modify > Table > Split Cell**, or click the **Split Cell** button in the Property inspector.
- 3 In the **Split Cell** dialog box, specify how to split the cell.

For more information, click the **Help** button in the dialog box.

To increase or decrease the number of rows or columns spanned by a cell:

- 1 Select a cell.
- 2 Choose Modify > Table > Increase Row Span or Modify > Table > Increase Column Span or Modify > Table > Decrease Row Span or Modify > Table > Decrease Column Span.

Nesting tables

A nested table is a table inside a cell of another table. You can format a nested table as you would any other table; however, its width is limited by the width of the cell in which it appears.

<table border="1"><tr><td></td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr></table>									

To nest a table within a table cell:

- 1 Click in a cell of the existing table.
- 2 Choose Insert > Table.
- 3 In the Insert Table dialog box, specify the desired properties for the nested table, then click OK.

Cutting, copying, and pasting cells

You can cut, copy, or paste a single table cell or multiple cells at once, preserving the cells' formatting.

You can paste cells at the insertion point or in place of a selection in an existing table. To paste multiple table cells, the contents of the Clipboard must be compatible with the structure of the table or the selection in the table in which the cells will be pasted.

To cut or copy table cells:

- 1 Select one or more cells in the table. The selected cells must be contiguous and in the shape of a rectangle.

In the following illustration, the selection is a rectangle of cells, so the cells can be cut or copied.

Location Name	City	State or Country
Baltimore-Washington International	Baltimore	MD
Cairo International	Cairo	Egypt
Canberra	Canberra	Australia
Cairns	Cairns	Queensland
Cape Town Airport	Cape Town	South Africa

In the following illustration, the selection is not a rectangle, so the cells can't be cut or copied.

Location Name	City	State or Country
Baltimore-Washington International	Baltimore	MD
Cairo International	Cairo	Egypt
Canberra	Canberra	Australia
Cairns	Cairns	Queensland
Cape Town Airport	Cape Town	South Africa

- 2 Cut or copy the cells using Edit > Cut or Edit > Copy.

If you selected an entire row or column and you choose Edit > Cut, the entire row or column is removed from the table (not just the contents of the cells).

To paste table cells:

- 1 Choose where you want to paste the cells.

- To replace existing cells with the cells you are pasting, select a set of existing cells with the same layout as the cells on the clipboard. For example, if you've copied or cut a 3 x 2 block of cells, you can select another 3 x 2 block of cells to replace by pasting.
- To paste a full row of cells above a particular cell, click in that cell.
- To paste a full column of cells to the left of a particular cell, click in that cell.
- To create a new table with the pasted cells, place the insertion point outside of the table.

Note: If you have less than a full row or column of cells in the clipboard, and you click in a cell and then paste the cells from the clipboard, the cell you clicked in and its neighbors may (depending on its location in the table) be replaced with the cells you are pasting.

- 2 Choose Edit > Paste.

If you are pasting entire rows or columns into an existing table, the rows or columns are added to the table. If you are pasting an individual cell, the contents of the selected cell are replaced. If you are pasting outside a table, the rows, columns, or cells are used to define a new table.

To remove cell content but leave the cells intact:

- 1 Select one or more cells. (Make sure the selection does not consist entirely of complete rows or columns.)
- 2 Choose Edit > Clear or press Delete.

Note: If only complete rows or columns are selected when you choose Edit > Clear or press Delete, the entire rows or columns—not just their contents—are removed from the table.

To delete rows or columns that contain merged cells:

- 1 Select the row or column to be deleted.
- 2 Choose Modify > Table > Delete Row or Modify > Table > Delete Column.

Sorting tables

You can sort the rows of a table based on the contents of a single column. You can also perform a more complicated table sort based on the contents of two columns.

You cannot sort tables that contain `COLSPAN` or `ROWSPAN` attributes—that is, tables that contain merged cells. (For more information, see “Splitting and merging cells” on page 235.)

To sort a table:

- 1 Select the table (or click in any cell).
- 2 Choose Commands > Sort Table.
The Sort Table dialog box appears.
- 3 In the Sort Table dialog box, specify how to sort the table.

For more information, click the Help button in the dialog box.

Exporting table data

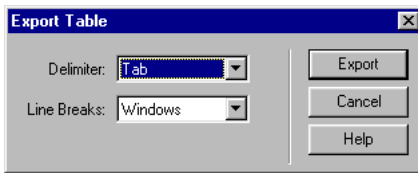
You can export table data from Dreamweaver into a text file, with the contents of adjacent cells separated by a delimiter. You can use commas, colons, semicolons, or spaces as delimiters.

When you export a table, the entire table is exported; you cannot select portions of a table to export. If you want only some of the data from a table—for example, the first six rows or the first six columns—then copy the cells containing that data, paste those cells outside of the table (to create a new table), and export the new table.

To export a table:

- 1 Place the insertion point in any cell of the table.
- 2 Choose File > Export > Table.

The Export Table dialog box appears.



- 3 In the Export Table dialog box, specify options for exporting the table.
For more information, click the Help button in the dialog box.
- 4 Click Export.
- 5 In the dialog box that appears, enter a name for the file, and then click Save.

CHAPTER 17

Laying Out Pages in Layout View

Dreamweaver gives you several different ways to lay out web pages.

One common method for creating a page layout is to use HTML tables to position elements. Tables can be difficult to use for layout, however, because they were originally created for displaying tabular data, not for laying out web pages.

To streamline the process of using tables for page layout, Dreamweaver provides Layout view. In Layout view, you can design your page using tables as the underlying structure, while avoiding some of the problems that often occur when creating table-based designs using traditional means. For example, in Layout view you can easily draw layout cells on your page, then move the cells where you want them. You can also easily create both fixed-width layouts and layouts that automatically stretch to the full width of the browser window (see “Setting column width” on page 250).

You can still lay out your pages using tables in the traditional way (see “Presenting Content with Tables” on page 227), or lay out pages using layers and then convert them to tables (see “Using tables and layers for layout” on page 386). However, Layout view is the easiest way to set up your page layout.

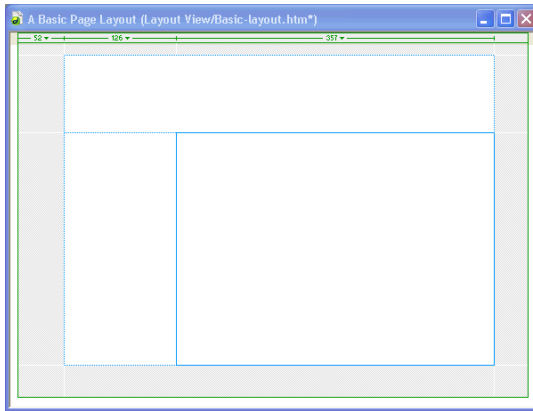
Note: In Layout view, you can’t use the Insert Table and Draw Layer tools that you can use in Standard view. To use those tools, you must first switch to Standard view.

This chapter contains these sections:

- “About layout cells and tables” on page 242
- “Switching into and out of Layout view” on page 242
- “Drawing layout cells and tables” on page 243
- “Adding content to a layout cell” on page 246
- “Moving and resizing layout cells and tables” on page 248
- “Formatting layout cells and tables” on page 250
- “Setting column width” on page 250
- “Setting Layout view preferences” on page 254

About layout cells and tables

In Layout view, you can lay out your page before adding any content. For example, you could draw a cell along the top of your page to hold a header graphic, another cell on the left side of the page to hold a navigation bar, and a third cell on the right to hold content.



(Another approach is to draw each cell only when you're ready to put content into it. This approach allows you maximum flexibility; it leaves more blank space in the layout table for a longer time, which allows you to move or resize cells more easily.)

When you draw a layout cell that isn't inside a layout table, Dreamweaver automatically creates a layout table as a container for the cell. A layout cell cannot exist outside of a layout table.

You can lay out your page using several layout cells within one layout table, which is the most common approach to web-page layout, or you can use multiple layout tables for more complex layouts. Using multiple layout tables isolates sections of your layout, so that they aren't affected by changes in other sections.

You can also nest layout tables, by placing a new layout table inside an existing layout table. This structure allows you to simplify the table structure when the rows or column in one part of the layout don't line up with the rows or columns in another part of the layout. For example, using nested layout tables you could easily create a two-column layout with four rows in the left column and three rows in the right column. For more information, see "Drawing a nested layout table" on page 245.

Switching into and out of Layout view

Before you can draw layout tables or layout cells, you must switch from Standard view into Layout view.

Tip: If you create a table in Standard view, then switch into Layout view, the resulting layout table may contain empty layout cells. You may need to delete these empty layout cells before you can create new layout cells or move layout cells around. When you create a new layout that you're going to edit in Layout view, it's easier to create the table in Layout view than to create it in Standard view.

To switch into Layout view:

- 1 If Design view isn't visible, choose View > Design or View > Code and Design.
Layout view cannot be enabled or disabled in Code view.
- 2 Choose View > Table View > Layout View or click the Layout View button in the Insert bar's Layout category. (In the Dreamweaver 4-style floating workspace, with a vertical Insert bar, the layout-related items appear at the bottom of the panel, rather than in a separate category.)
A gray bar labeled Layout View appears across the top of Design view, to indicate that you're in Layout view. If there are tables on your page, they appear as layout tables.

To switch out of Layout view:

- 1 If Design view isn't visible, choose View > Design or View > Code and Design.
Layout view cannot be enabled or disabled in Code view.
- 2 Choose View > Table View > Standard View or click the Standard View button in the Insert bar's Layout category.

Drawing layout cells and tables

You can draw layout cells and tables on your page in Layout view. When you draw a layout cell that isn't inside a layout table, Dreamweaver automatically creates a layout table as a container for the cell. A layout cell cannot exist outside of a layout table.

Note: When Dreamweaver automatically creates a layout table, the table initially appears to fill the entire Design view, even if you change the size of your Document window. This full-window default layout table allows you to draw layout cells anywhere in Design view. You can set the table to a specific size by clicking the table border and then dragging the table's resize handles.

To draw a layout cell:



- 1 Make sure you are in Layout view (see “Switching into and out of Layout view” on page 242), then click the Draw Layout Cell button in the Layout category of the Insert bar. (In the Dreamweaver 4-style floating workspace, with a vertical Insert bar, the layout-related items appear at the bottom of the panel, rather than in a separate category.)

The pointer changes to a plus sign (+).

- 2 Position the pointer where you want to start the cell on the page, then drag to create the layout cell. To create multiple cells without having to click the Draw Layout Cell button each time, create each layout cell by Control-dragging (Windows) or Command-dragging (Macintosh).

The cell appears outlined in blue on your page. (Blue is the default outline color for layout cells. To change the outline color, see “Setting Layout view preferences” on page 254.) The width of each cell is displayed in the column header area at the top of the column, if layout table tabs are showing (see “Setting Layout view preferences” on page 254). For more information on column widths, see “Setting column width” on page 250.

A light grid of lines appears, extending from the edges of the new layout cell out to the edges of the layout table that contains it. These lines help you align new cells with old cells, and help you visualize the underlying HTML table's structure.

Dreamweaver automatically snaps the edges of new cells into alignment with nearby edges of existing cells. (Layout cells cannot overlap.) Cell edges also automatically snap to the edges of the containing layout table if you draw a cell close to the edge of a table. To temporarily disable snapping, hold down Alt (Windows) or Option (Macintosh) while drawing the cell.

To draw a layout table:

1 Make sure you are in Layout view. Then do one of the following:



- To draw one layout table, click the Draw Layout Table button in the Layout category of the Insert bar. The pointer changes to a plus sign (+).
- To draw more than one layout table without having to repeatedly click the Draw Layout Table button, Control-click (Windows) or Command-click (Macintosh) the Draw Layout Table button. When you finish drawing a layout table, you can immediately draw another one.

2 Position the pointer on the page, then drag to create the layout table.

If there's no other content on the page, the new table is automatically positioned at the upper left corner of the page.

The table appears outlined in green on your page. (Green is the default outline color for layout tables. To change the outline color, see “Setting Layout view preferences” on page 254.) A tab labeled *Layout Table* appears at the top of each table you draw, to help you select the table and distinguish it from other elements of your page.

The width of the table (in pixels, or as a percentage of the page width) is displayed in the column header area along the top of the table, if layout table tabs are showing (see “Setting Layout view preferences” on page 254). For more information on table and column width, see “Setting column width” on page 250.



You can create a layout table in an empty area of your page layout, or around existing layout cells and tables, or nested inside an existing layout table. Tables cannot overlap each other, but a table can be completely contained inside another table. For more information, see “Drawing a nested layout table” on page 245.

Tip: You can't draw a layout table next to existing content. If your page already contains any content, you can draw a new layout table only below the bottom of the existing content. If you try to draw a layout table below existing content but the no-draw pointer appears, try resizing the Document window to create more blank space between the bottom of the existing content and the bottom of the window.

Drawing a nested layout table

You can draw a layout table inside another layout table, to create a nested table. The cells inside a nested table are isolated from changes made to the outer table; for example, when you change the size of a row or column in the outer table, the cells in the inner table don't change size.

You can insert multiple levels of nested tables. A nested layout table cannot be larger than the table that contains it.

Note: If you draw a layout table in the middle of your page before drawing a layout cell, the table you draw is automatically nested inside a larger table.



To draw a nested layout table:



- 1 Make sure you are in Layout view, then click the Draw Layout Table button in the Layout category of the Insert bar.

The pointer changes to a plus sign (+).

- 2 Point to an empty (gray) area in an existing layout table, then drag to create the nested layout table.

Note: You can't create a layout table inside a layout cell. You can create a nested layout table only in an empty area of an existing layout table, or around existing cells.

To draw a layout table around existing layout cells or tables:

- 1 Make sure you are in Layout view, then click the Draw Layout Table button in the Layout category of the Insert bar.

The pointer changes to a plus sign (+).

- 2 Drag to draw a rectangle around a set of existing layout cells or tables.

A new nested layout table appears, enclosing the existing cells or tables.

Tip: To make an existing layout cell fit snugly into one corner of the new nested table, start dragging near the corner of the cell; the new table's corner snaps to the cell's corner. You can't start dragging in the middle of a layout cell, because you can't create a layout table entirely inside a layout cell.

Snapping layout cells to the grid

You can turn on the Dreamweaver grid to use as a visual guide while you draw your layout. You can make page elements automatically snap to the grid as you move them, and change the grid or control the snapping behavior by specifying grid settings. Snapping works whether or not the grid is visible.

To show or hide the grid:

Choose View > Grid > Show Grid.

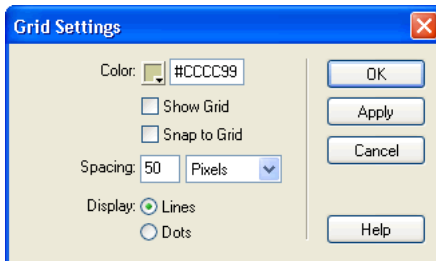
To enable or disable snapping:

Choose View > Grid > Snap to Grid.

To change grid settings:

- 1 Choose View > Grid > Grid Settings.

The Grid Settings dialog box appears.



- 2 Set options as desired.

For more information, click the Help button in the dialog box.

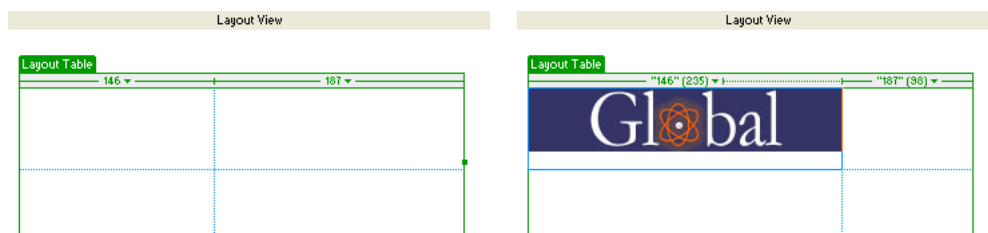
- 3 Click OK.

Adding content to a layout cell

You can add text, images, and other content to layout cells in Layout view just as you would add content to table cells in Standard view. Click in the cell where you want to add content, then type text or insert other content.

You can insert content only into a layout cell, not into an empty (gray) area of a layout table, so before you can add content, you must first create layout cells (see “Drawing layout cells and tables” on page 243).

A layout cell expands automatically when you add content that is wider than the cell. As the cell expands, the column that the cell is in also expands, which might change the sizes of surrounding cells. The column header area for that column changes to show the width that appears in the code, followed by the visual width of the column (the width as it appears on your screen) in parentheses. (For more information about column widths, see “Setting column width” on page 250.)



To add text to a layout cell:

Place the insertion point in the layout cell where you want to add text and do one of the following:

- Type text into the cell. The cell automatically expands as you type, if necessary.
- Paste text copied from another document. Use the Paste command. For more information, see “Inserting and formatting HTML text” on page 271.

To add an image to a layout cell:

- 1 Place the insertion point in the layout cell where you want to add the image.
- 2 Do one of the following:
 - Click the Insert Image button in the Insert bar’s Common category.
 - Choose Insert > Image.
- 3 In the Select Image dialog box, select an image file.

For more information, see “Inserting an image” on page 298.

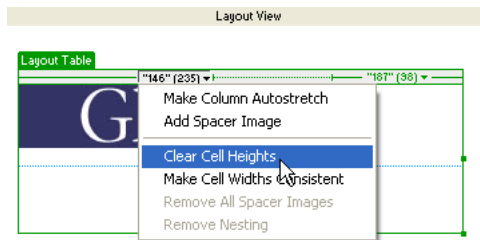
Clearing automatically set cell heights

When you create a layout cell, Dreamweaver automatically specifies a height for the cell, to make the cell display at the height you drew even though the cell is empty. After you insert content into the cell, you may no longer need the height to be specified, so you can remove the explicit cell heights from the table.

To clear cell heights, do one of the following:

- Select Clear Cell Heights from the column-header menu.

Dreamweaver clears all the heights specified in the table. Some of the table cells may shrink vertically.



- Select a layout table by clicking the tab at the top of the table, then click the Clear Row Heights button in the Property inspector.

Dreamweaver clears all the heights specified in the table.

Moving and resizing layout cells and tables

To adjust your page layout, you can move and resize layout cells and nested layout tables. (The outermost layout table can only be resized.)

Layout cells cannot overlap. You cannot move or resize a cell to make it cross the boundaries of the layout table that contains it. A layout cell cannot be made smaller than its contents.

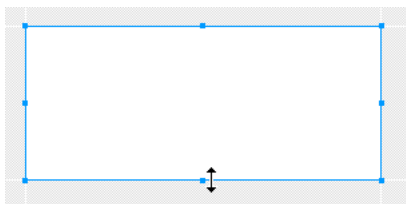
A layout table cannot be resized to be smaller than the smallest rectangle containing all of its cells. A layout table also cannot be resized so that it overlaps other tables or cells.

To resize a layout cell:

- 1 Select a cell by clicking an edge of the cell or by Control-clicking (Windows) or Command-clicking (Macintosh) anywhere in the cell.

Selection handles appear around the cell.

- 2 Drag a selection handle to resize the cell.



The cell edges automatically snap to align with other cells' edges.

To move a layout cell:

- 1 Select a cell by clicking an edge of the cell or by Control-clicking (Windows) or Command-clicking (Macintosh) anywhere in the cell.
Selection handles appear around the cell.
- 2 Do one of the following:
 - Drag the cell to another location within its layout table.
 - Press the arrow keys to move the cell 1 pixel at a time. Hold down Shift while pressing an arrow key to move the cell 10 pixels at a time.

To resize a layout table:

- 1 Select a table by clicking the tab at the top of the table.
Selection handles appear around the table.
- 2 Drag the selection handles to resize the table.
The table edges automatically snap to align with the edges of other cells and tables.

To move a layout table:

- 1 Select a table by clicking the tab at the top of the table.
Selection handles appear around the table.
- 2 Do one of the following:
 - Drag the table to another location on the page.
Note: You can move a layout table only if it's nested inside another layout table.
 - Press the arrow keys to move the table 1 pixel at a time. Hold down Shift while pressing an arrow key to move the table 10 pixels at a time.

Related topic

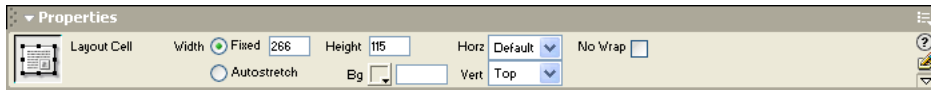
To use the Dreamweaver grid as a guide while you move or resize your cells and tables, see “Snapping layout cells to the grid” on page 246.

Formatting layout cells and tables

You can change the appearance of any layout cell or table by using the Property inspector.

Formatting layout cells

You can set various attributes of a layout cell in the Property inspector, including width and height, background color, and alignment of the cell's contents.



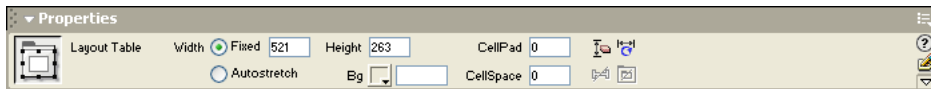
To format a layout cell in the Property inspector:

- 1 Select a cell by clicking an edge of the cell or by Control-clicking (Windows) or Command-clicking (Macintosh) anywhere in the cell.
- 2 Open the Property inspector by choosing Window > Properties.
- 3 Change the cell's formatting by setting properties.

For more information, click the Help button in the Property inspector.

Formatting layout tables

You can set various attributes of a layout table in the Property inspector, including width, height, padding, and spacing.



To format a layout table:

- 1 Select a table by clicking the tab at the top of the table, or by clicking the `<table>` tag in the tag selector.
- 2 Open the Property inspector by choosing Window > Properties.
- 3 Change the table's formatting by setting properties.

For more information, click the Help button in the Property inspector.

Setting column width

A column in Layout view can have either a fixed width or a width that automatically expands to fill as much of the browser window as possible ("autostretch"). Information about the width appears in the column header area at the top of each column of the selected table, if layout table tabs are showing (see "Setting Layout view preferences" on page 254).

A fixed-width column has a specific numeric width, such as 300 pixels; the width appears in the column header area (unless the column is too narrow for the numbers to be displayed). An autostretch column's width changes automatically depending on the browser window's width; the column header area for an autostretch column shows a wavy line instead of a number. If your layout includes an autostretch column, the layout always fills the entire width of the visitor's browser window.

Note: The width you specify for a column applies to all the cells in that column.

You can make only one column in a given layout table autostretch. A common layout is to make the column containing the main content of the page autostretch, which automatically sets all the other columns to fixed width.

For example, suppose your layout has a wide image on the left side of the page and a column of text on the right. You might set the left column to a fixed width and make the sidebar area autostretch.



When you make a column autostretch, Dreamweaver inserts spacer images in the fixed width columns to ensure that those columns stay as wide as they should be, unless you specify that no spacer image should be used. A spacer image is a transparent image, used to control spacing, that is not visible in the browser window. For more information, see “Using spacer images” on page 252.

Note: Making a column autostretch before your layout is complete may have unexpected effects on table layout. To prevent columns from growing unexpectedly wider or narrower, create your complete layout before making a column autostretch, and use spacer images when making a column autostretch. (However, if each column contains other content that will keep the column at the desired width, you don't need spacer images.)

Sometimes the visual width of a column as it appears in Layout view doesn't match the width specified in the HTML code. In that case, two numbers appear in the column header area. (For more information on inconsistent widths, click the Help button in the layout table Property inspector.)

To make a column autostretch, do one of the following:

- In the column header area at the top of a fixed width column, select Make Column Autostretch from the column header menu. You can make only one column in a given table autostretch.



- Select a cell in the column by clicking an edge of the cell or by Control-clicking (Windows) or Command-clicking (Macintosh) anywhere in the cell. Then, in the Property inspector, click Autostretch.

To set a column to a fixed width, do one of the following:

- In the column header area at the top of an autostretch column, select Make Column Fixed Width from the column header menu.

Make Column Fixed Width specifies a width for the column (in the code) that matches the current visual width of the column.

- Select a cell in the column by clicking an edge of the cell or by Control-clicking (Windows) or Command-clicking (Macintosh) anywhere in the cell. Then, in the Property inspector, click Fixed and type a numeric value.

If you enter a numeric value that is less than the width of the column's content, Dreamweaver automatically sets the width to match the width of the content.

To set the width of cells as specified in the HTML code to match the cells' visual width:

Select Make Cell Widths Consistent from the column header menu for any column.

Related topics

"Setting Layout view preferences" on page 254

Using spacer images

A spacer image (also known as a *spacer GIF*) is a transparent image that's used to control spacing in autostretch tables. A spacer image consists of a single-pixel transparent GIF image, stretched out to be a specified number of pixels wide. A browser can't draw a table column narrower than the widest image contained in a cell in that column, so placing a spacer image in a table column requires browsers to keep the column at least as wide as the image.

Dreamweaver automatically adds spacer images when you set a column to autostretch, unless you specify that no spacer image should be used. You can manually insert and remove spacer images in each column if you prefer. Columns that contain spacer images have a double bar in the column header area.

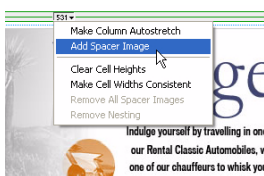
Note: If you choose not to use spacer images in autostretch tables, fixed-width columns may change size or even disappear completely from Design view when they do not contain content. (The columns still exist in the code, even if they don't appear in Design view.)

When you insert a spacer image in a column or set a column to autostretch, a dialog box appears, asking how you want to set up your spacer image file. If you have already set a spacer image for the site in the Dreamweaver preferences, the dialog box does not appear (see “Setting Layout view preferences” on page 254).

You can manually insert and remove spacer images in specific columns or remove all spacer images in the page.

To set up a spacer image:

- 1 Make a column autostretch, or choose Add Spacer Image from the column header menu.



The Choose Spacer Image dialog box appears.

- 2 Choose one of the options.

For more information, click the Help button in the dialog box.

- 3 Click OK.

To insert a spacer image into a column:

Select Add Spacer Image from the column header menu.

The spacer image is inserted into the column. The image is not visible, but the column may shift slightly and a double bar appears at the top to indicate that it contains a spacer image.

To remove a spacer image from a column:

Select Remove Spacer Image from the column header menu.

The spacer image is removed. The column may shift.

To remove all spacer images from a table:

Do one of the following:

- Select Remove All Spacer Images from the column header menu of any column in the table.
- Click the Remove All Spacers button in the layout table Property inspector.



The layout of your whole table may shift. If there is no content in some columns, the columns may disappear completely from the Design view.

Setting Layout view preferences

Use the Layout View category in the Preferences dialog box to specify information about spacer image files and about the colors that Dreamweaver uses to draw layout tables and layout cells.

By default, a tab labeled *Layout Table* appears at the top of each layout table in Layout view, and a set of column-width controls appears at the top of the currently selected layout table. You can prevent the tabs and controls from appearing if you prefer.

Note: Even when tabs are showing, the layout table at the top of the page doesn't have a tab when it's selected. The tab for a layout table at the top of the page appears only when the table is not selected.

To set Layout view preferences:

- 1 Choose Edit > Preferences.
- 2 Select Layout View from the Category list.
- 3 Make changes as necessary.

For more information, click the Help button in the dialog box.

- 4 Click OK to close the dialog box.

To turn off the layout table tabs and column-width controls in Layout view:

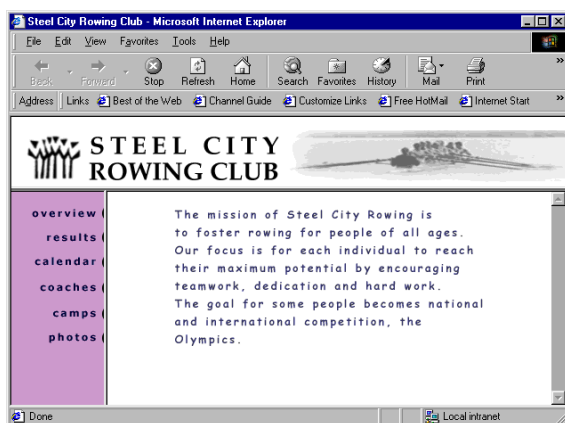
Choose View > Table View > Show Layout Table Tabs.

CHAPTER 18

Using Frames

Frames provide a way to divide a browser window into multiple regions, each of which can display a different HTML document. In the most common use of frames, one frame displays a document containing navigation controls, while another frame displays a document with content.

For example, your frame layout might consist of three frames: one narrow frame on the side that contains a navigation bar, one frame that runs along the top, containing the logo and title of the website, and one large frame that takes up the rest of the page and contains the main content. Each of these frames displays a separate HTML document.



In this example, the document displayed in the top frame never changes as the visitor navigates the site. The side frame navigation bar contains links; clicking one of these links changes the content of the main content frame, but the contents of the side frame itself remain static. The main content frame on the right displays the appropriate document for whichever link the visitor clicks on the left.

A detailed discussion of all the ways to design and use frames, and the code required for hand-coding them, is beyond the scope of this chapter. If you need detailed information about the code used in advanced frame layouts, see a book about frames and framesets.

This chapter contains the following sections:

- “About frames and framesets” on page 256
- “Deciding whether to use frames” on page 257
- “About creating frame-based web pages in Dreamweaver” on page 258

- “Creating frames and framesets” on page 259
- “Selecting frames and framesets” on page 261
- “Saving frame and frameset files” on page 263
- “Viewing and setting frame properties” on page 264
- “Viewing and setting frameset properties” on page 264
- “Controlling frame content with links” on page 265
- “Handling browsers that can’t display frames” on page 266
- “Using JavaScript behaviors with frames” on page 267

About frames and framesets

A frame is a region in a browser window that can display an HTML document independent of what’s being displayed in the rest of the browser window.

A frameset is an HTML file that defines the layout and properties of a set of frames, including the number of frames, the size and placement of the frames, and the URL of the page to be initially displayed in each frame. The frameset file itself doesn’t contain HTML content to be displayed in a browser, except in the `noframes` section (see “Handling browsers that can’t display frames” on page 266); the frameset file simply provides information to the browser about how a set of frames should be displayed and what documents should be displayed in them.

To view a set of frames in a browser, enter the URL of the frameset file; the browser then opens the relevant documents to display in the frames. The frameset file for a site is often named `index.html`, so that it displays by default if a visitor doesn’t specify a filename.

Note that a frame is not a file. It’s easy to think of the document that’s currently displayed in a frame as an integral part of the frame, but the document isn’t actually part of the frame—any frame can display any document.

Note: The word *page* can be used loosely to refer either to a single HTML document or to the entire contents of a browser window at a given moment, even if several HTML documents are being displayed at once. The phrase “a page that uses frames,” for example, usually refers to a set of frames and the documents that initially appear in those frames.

In Dreamweaver, you can create a frameset in either of the following ways:

- To create a frameset with the current document displayed in one of the frames, use the Insert bar’s Frames category.
- To create a frameset with all its frames blank, use the Frameset category in the New Document dialog box.

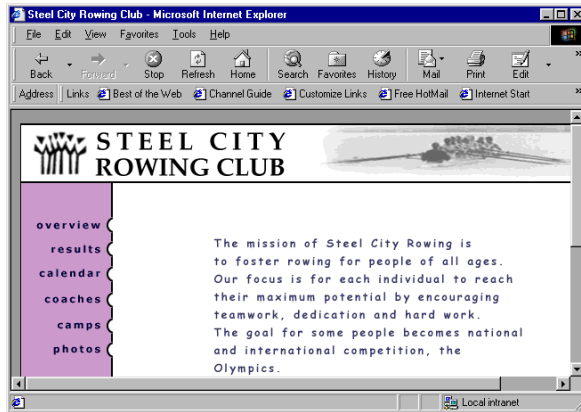
You can format all of your frames and framesets through the Property inspector. You can set scrolling on or off, set width and height, name each frame, and more. For more information, see “Viewing and setting frame properties” on page 264 and “Viewing and setting frameset properties” on page 264.

A site that appears in a browser as a single page comprising three frames actually consists of at least four separate HTML documents: the frameset file, plus the three documents containing the content that initially appears inside the frames. When you design a page using framesets in Dreamweaver, you must save each of these four files in order for the page to work properly in the browser. For more information about how to create successful web pages that use frames, see “About creating frame-based web pages in Dreamweaver” on page 258.

Deciding whether to use frames

The most common use of frames is for navigation. A set of frames often includes one frame containing a navigation bar and another frame to display the main content pages.

However, designing with frames can be confusing, and in many cases you can create a web page without frames that accomplishes many of the same goals as a set of frames. For example, if you want a navigation bar to appear on the left side of your page, you can either replace your page with a set of frames, or just include the navigation bar on every page in your site. (Dreamweaver helps you create multiple pages that use the same layout; see “About Dreamweaver templates” on page 435.) The following image shows a page design with a frame-like layout that doesn’t use frames.



Many professional web designers prefer not to use frames, and many people who browse the web dislike frames. In most cases this dislike is due to having encountered sites that use frames poorly or unnecessarily (such as a frameset that reloads the contents of the navigation frames every time the visitor clicks a navigation button). When frames are used well (such as when they’re used to keep navigation controls static in one frame while allowing the contents of another frame to change), they can be very useful for some sites.

Not all browsers provide good frame support, and frames may be difficult for visitors with disabilities to navigate, so if you do use frames, always provide a `noFrames` section in your frameset, for visitors who can’t view them. You may also want to provide an explicit link to a frameless version of the site, for visitors whose browsers support frames but who don’t like using frames.

Advantages to using frames include the following:

- A visitor’s browser doesn’t need to reload the navigation-related graphics for every page.
- Each frame has its own scrollbar (if the content is too large to fit in a window), so a visitor can scroll the frames independently. (For example, a visitor who has scrolled down to the bottom of a long page of content in a frame doesn’t need to scroll back up to the top to use the navigation bar if the navigation bar is in a different frame.)

Disadvantages to using frames include the following:

- Precise graphical alignment of elements in different frames can be difficult.
- Testing the navigation can be time-consuming.

- The URLs of the individual framed pages aren't displayed in the browser, so it can be difficult for a visitor to bookmark a specific page (unless you provide server code that allows them to load a framed version of a particular page).

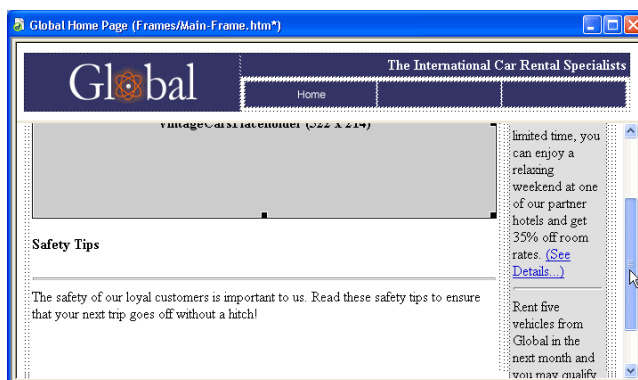
Related topics

“Controlling frame content with links” on page 265

“Handling browsers that can't display frames” on page 266

About creating frame-based web pages in Dreamweaver

Dreamweaver allows you to view and edit all of the documents associated with a set of frames, all in one Document window. This approach lets you see approximately how the framed pages will appear in a browser as you edit them. However, some aspects of this approach can be confusing until you get used to them.



In particular, remember that each frame displays a separate HTML document. Even if the documents are empty, you must save them all before you can preview them (because the frameset can be accurately previewed only if it contains the URL of a document to display in each frame).

To ensure that your frameset appears correctly in browsers:

- 1 Create your frameset and specify a document to appear in each frame (see “Creating frames and framesets” on page 259).
- 2 Save every file that's going to be displayed in a frame. Remember that each frame displays a separate HTML document, and each document must be saved. Also save the frameset file. (See “Saving frame and frameset files” on page 263.)
- 3 Set the properties for each frame and for the frameset. This includes naming each frame, setting scrolling and non-scrolling options, and more. (See “Viewing and setting frame properties” on page 264 and “Viewing and setting frameset properties” on page 264.)

Tip: You may also want to set the title attribute for a frame, to improve accessibility. (Note that the title attribute is not the same as the name attribute.) To set the title attribute, select the frame and choose Modify > Edit Tag; then select the Style Sheet/Accessibility category and enter a title in the Title text box. Alternatively, enable the accessibility authoring option for frames; for more information, see “Authoring for accessibility” on page 337.

- 4 Make sure to set the Target property in the Property inspector for all your links so that the linked content appears in the correct area (see “Controlling frame content with links” on page 265).

Creating frames and framesets

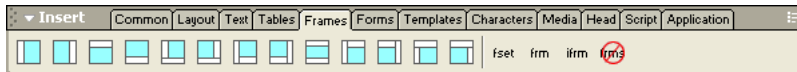
There are two ways to create a frameset in Dreamweaver: you can design it yourself, or you can select from several predefined framesets. Choosing a predefined frameset automatically sets up all the framesets and frames needed to create the layout and is the easiest way to create a frames-based layout quickly. You can insert a predefined frameset only in the Document window's Design view.

Creating a predefined frameset

Predefined framesets make it easy for you to select the type of frameset you want to create.

There are two ways to create a predefined frameset: using the Insert bar and using the New Document dialog box. The Insert bar allows you to create a frameset and display the current document in one of the new frames; the New Document dialog creates a new empty frameset.

The predefined frameset icons in the Frames category of the Insert bar and in the Framesets category of the New Document dialog box provide a visual representation of each frameset as applied to the current document.



When you apply a frameset using the Insert bar, Dreamweaver automatically sets up the frameset to display the current document (the document in which the insertion point is located) in one of the frames. The blue area of a predefined frameset icon represents the current document, and the white areas represent frames that will display other documents.

To create a predefined frameset and display an existing document in a frame:

- 1 Place the insertion point in a document.
- 2 Do one of the following:
 - In the Frames category of the Insert bar, click the icon for a predefined frameset.
 - Choose a predefined frameset from the Insert > Frames submenu.

To create a new empty predefined frameset:

- 1 Choose File > New.
- 2 In the New Document dialog box, select the Framesets category.
- 3 Select a frameset from the Framesets list.
- 4 Click Create.

Creating and editing a frameset

Before creating a frameset or working with frames, make the frame borders visible in the Document window's Design view by choosing View > Visual Aids > Frame Borders.

To create a frameset:

Choose a splitting item (such as Split Frame Left or Split Frame Right) from the Modify > Frameset submenu.

The window is split into frames, and the document you started with appears in one of the frames.

To split a frame into smaller frames, do one of the following:

- To split the frame where the insertion point is, choose a splitting item from the Modify > Frameset submenu.
- To split a frame or set of frames vertically or horizontally, drag a frame border from the edge of the Design view into the middle of the Design view.
- To split a frame using a frame border that isn't at the edge of the Design view, Alt-drag (Windows) or Option-drag (Macintosh) a frame border.
- To divide a frame into four frames, drag a frame border from one of the corners of the Design view into the middle of a frame.

Tip: To create three frames, start with two frames and then split one of them. It's not easy to merge two adjacent frames without editing the frameset code, so turning four frames into three frames is harder than turning two frames into three frames.

To delete a frame:

Drag a frame border off the page or to a border of the parent frame.

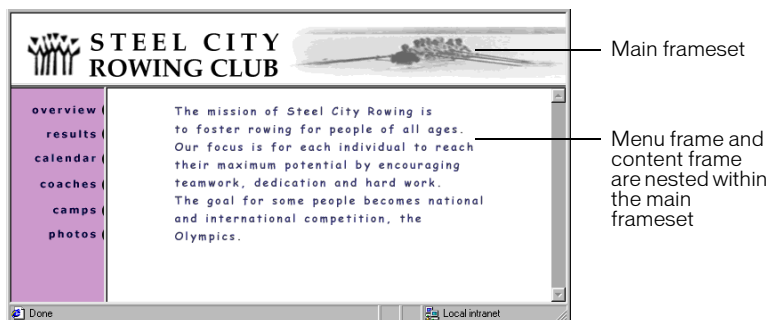
If there's unsaved content in a document in a frame that's being removed, Dreamweaver prompts you to save the document.

Note: You can't remove a frameset entirely by dragging borders. To remove a frameset, close the Document window that displays it. If the frameset file has been saved, delete the file.

About nested framesets

A frameset inside another frameset is called a nested frameset. A single frameset file can contain multiple nested framesets. Most web pages that use frames are actually using nested frames, and most of the predefined framesets in Dreamweaver also use nesting. Any set of frames in which there are different numbers of frames in different rows or columns requires a nested frameset.

For example, the most common frame layout has one frame in the top row (where the company's logo appears) and two frames in the bottom row (a navigation frame and a content frame). This layout requires a nested frameset: a two-row frameset, with a two-column frameset nested in the second row.



Dreamweaver takes care of nesting framesets as needed; if you use the frame-splitting tools in Dreamweaver, you don't need to worry about the details of which frames are nested and which aren't. For more information about the frame-splitting tools, see "Creating and editing a frameset" on page 259.

There are two ways to nest framesets in HTML: the inner frameset can be defined either in the same file as the outer frameset, or in a separate file of its own. Each predefined frameset in Dreamweaver defines all of its framesets in the same file.

Both kinds of nesting produce the same visual results; it's not easy to tell, without looking at the code, which kind of nesting is being used. The most likely situation in which an external frameset file would be used in Dreamweaver is when you use the Open in Frame command to open a frameset file inside a frame; doing this may result in problems with setting targets for links. It's generally simplest to keep all framesets defined in a single file.

Selecting frames and framesets

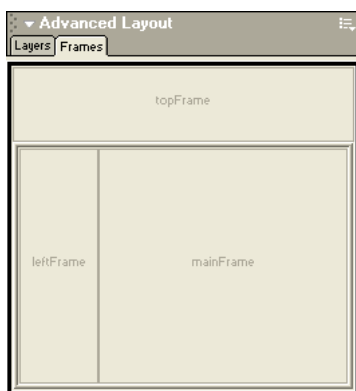
To make changes to the properties of a frame or frameset, begin by selecting the frame or frameset you want to change. You can select a frame or frameset either in the Document window or by using the Frames panel.

When you select a frame or frameset, a selection outline appears around the frame or frameset in both the Frames panel and the Document window's Design view.

Selecting frames and framesets in the Frames panel

The Frames panel provides a visual representation of the frames within a frameset. You can click a frame or frameset in the Frames panel to select that frame or frameset in the document, and then you can view or edit the properties of the selected item in the Property inspector. For more information, see “Viewing and setting frame properties” on page 264 and “Viewing and setting frameset properties” on page 264.

The Frames panel shows the hierarchy of the frameset structure in a way that may not be apparent in the Document window. In the Frames panel, a very thick border surrounds each frameset; each frame is surrounded by a thin gray line and is identified by a frame name.



To display the Frames panel:

Choose Window > Others > Frames.

To select a frame in the Frames panel:

Click the frame in the Frames panel.

To select a frameset in the Frames panel:

Click the border that surrounds the frameset in the Frames panel.

Selecting frames and framesets in the Document window

In the Document window's Design view, when a frame is selected, its borders are outlined with a dotted line; when a frameset is selected, all the borders of the frames within the frameset are outlined with a light dotted line.

Note: Placing the insertion point in a document that's displayed in a frame is not the same as selecting a frame. There are various operations (such as setting frame properties) for which you must select a frame.

When you select a frame or frameset, the Property inspector displays the properties of the selected item, allowing you to change the properties (see "Viewing and setting frame properties" on page 264 and "Viewing and setting frameset properties" on page 264).

To select a frame in the Document window:

Alt-click (Windows) or Option-Shift-click (Macintosh) inside a frame in Design view.

To select a frameset in the Document window:

Click one of the frameset's internal frame borders in Design view. (Frame borders must be visible to do this; choose View > Visual Aids > Frame Borders to make frame borders visible if they aren't.)

Note: It's generally easier to select framesets in the Frames panel than in the Document window. For more information, see "Selecting frames and framesets in the Frames panel" on page 261.

To select a different frame or frameset, do one of the following:

- To select the next or previous frame or frameset at the same hierarchical level as the current selection, press Alt-Left Arrow or Alt-Right Arrow (Windows) or Command-Left Arrow or Command-Right Arrow (Macintosh). Using these keys, you can cycle through frames and framesets in the order in which they're defined in the frameset file.
- To select the parent frameset (the frameset that contains the current selection), press Alt-Up Arrow (Windows) or Command-Up Arrow (Macintosh).
- To select the first child frame or frameset of the currently selected frameset (that is, first in the order in which they're defined in the frameset file), Press Alt-Down Arrow (Windows) or Command-Down Arrow (Macintosh).

Opening a document in a frame

You can specify the initial content of a frame by either inserting new content into an empty document in a frame, or opening an existing document in a frame.

To open an existing document in a frame:

- 1 Place the insertion point in a frame.
- 2 Choose File > Open in Frame.
- 3 Select a document to open in the frame and click OK (Windows), Choose (Mac OS X), or Open (Mac OS 9).

The document appears in the frame.

- 4 To make that document the default document to display in the frame when the frameset is opened in a browser, save the frameset.

Saving frame and frameset files

Before you can preview a frameset in a browser, you must save the frameset file and all of the documents that are to be displayed in the frames. You can save each frameset file and framed document individually, or you can save the frameset file and all documents appearing in frames at once.

When you use visual tools in Dreamweaver to create a set of frames, each new document that appears in a frame is given a default filename. For example, the first frameset file is named `UntitledFrameset-1`, while the first document in a frame is named `UntitledFrame-1`.

When you select one of the save commands, a dialog box appears, ready to save a document with its default filename. Because the default filenames are so similar, it may be difficult for you to determine which document you are saving. To identify the frame that displays the document you're saving, look at the frame selection outline in the Document window (in Design view).



To save a frameset file:

- 1 Select the frameset in the Frames panel or the Document window.
- 2 Choose one of the following:
 - To save the frameset file, choose File > Save Frameset.
 - To save the frameset file as a new file, choose File > Save Frameset As.

If the frameset file has not previously been saved, these two commands are equivalent.

To save a document that appears in a frame:

Click in the frame, then choose File > Save Frame or File > Save Frame As.

To save all files associated with a set of frames:

Choose File > Save All Frames.

This saves all open documents in the frameset, including the frameset file and all framed documents. If the frameset file has not yet been saved, a heavy border appears around the frameset in the Design view, and a dialog box allows you to choose a filename. Then for each frame that hasn't yet been saved, a heavy border appears around the frame, and a dialog box allows you to choose a filename.

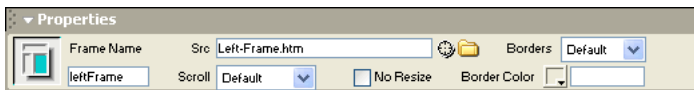
Note: If you used File > Open in Frame to open a document in a frame, then when you save the frameset, the document you opened in the frame becomes the default document to be displayed in that frame. If you don't want that document to be the default, don't save the frameset file.

Viewing and setting frame properties

Use the Property inspector to view and set most frame properties. To change the background color of a frame, set the background color of the document in the frame.

To view or set frame properties:

- 1 Select a frame by doing one of the following:
 - Alt-click (Windows) or Shift-Option-click (Macintosh) a frame in the Document window's Design view.
 - Click a frame in the Frames panel.
- 2 Choose Window > Properties to open the Property inspector if it isn't already open.



- 3 To see all of the frame properties, click the expander arrow in the lower right corner of the Property inspector.

For more information, click the Help button in the Property inspector.

To change the background color of a document in a frame:

- 1 Place the insertion point in the frame.
- 2 Choose Modify > Page Properties.
- 3 Click the Background pop-up menu to select a color.

Related topics

“Viewing and setting frameset properties” on page 264

“Setting document properties” on page 112

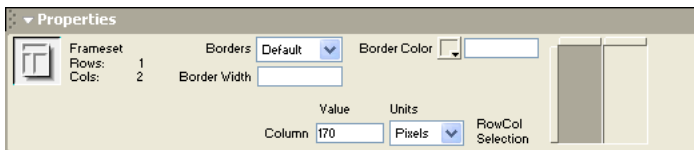
Viewing and setting frameset properties

Use the Property inspector to view and set most frameset properties. To set the title of the selected frameset, use the Page Properties dialog box or the Title field in the Document window's toolbar.

To view or set frameset properties:

- 1 Select a frameset by doing one of the following:
 - Click a border between two frames in the frameset in the Document window's Design view.
 - Click the border that surrounds a frameset in the Frames panel.
- 2 Choose Window > Properties to open the Property inspector if it isn't already open.

- 3 To see all of the frameset properties, click the expander arrow in the lower right corner of the Property inspector.



For more information, click the Help button in the Property inspector.

To set a title for a frameset document:

- 1 Select a frameset by doing one of the following:
 - Click a border between two frames in the frameset in the Document window's Design view.
 - Click the border that surrounds a frameset in the Frames panel.
- 2 In the Title field of the Document toolbar, type a name for the document.

When a visitor views the frameset in a browser, the title appears in the browser's title bar.

Related topics

“Viewing and setting frame properties” on page 264

“Setting document properties” on page 112

“Using the Document toolbar” on page 39

Controlling frame content with links

To use a link in one frame to open a document in another frame, you must set a target for the link. The `target` attribute of a link specifies the frame or window in which the linked content opens. For example, if your navigation bar is in the left frame, and you want the linked material to appear in the main content frame on the right, you must specify the name of the main content frame as the target for each of the navigation bar links. When a visitor clicks a navigation link, the specified content opens in the main frame.

To select a frame in which to open a file, use the Target pop-up menu in the Property inspector. You can set a file to replace the document being displayed in another frame, to appear in place of the entire frameset, to appear in the frame where the link was (by not choosing a target), or to open in a new browser window.

To target a frame:

- 1 In Design view, select text or an object.
- 2 In the Link field of the Property inspector, do one of the following:
 - Click the folder icon and select the file to link to.
 - Drag the Point to File icon to select the file to link to.

3 In the Target pop-up menu, choose the frame or window in which the linked document should appear.

- If you named your frames in the Property inspector, the frame names appear in this menu. Select a named frame to open the linked document in that frame.

Note: Frame names appear only when you're editing a document within a frameset. When you edit a document in its own Document window, outside of the frameset, frame names do not appear in the Target pop-up menu. If you're editing a document outside of the frameset, you can type the target frame's name into the Target text box.

- `_blank` opens the linked document in a new browser window, leaving the current window untouched.
- `_parent` opens the linked document in the parent frameset of the frame the link appears in, replacing the entire frameset.
- `_self` opens the link in the current frame, replacing the content in that frame.
- `_top` opens the linked document in the current browser window, replacing all frames.

Tip: If you're linking to a page outside of your site, always use `target="_top"` or `target="_blank"` to ensure that the page doesn't appear to be part of your site.

Handling browsers that can't display frames

Dreamweaver lets you specify content to display in text-based browsers and in older graphical browsers that do not support frames. This content is stored in the frameset file, wrapped in a `noframes` tag. When a browser that doesn't support frames loads the frameset file, the browser displays only the content enclosed by the `noframes` tag.

Note: Content in the `noframes` area should be more than just a note saying "You should upgrade to a browser that can handle frames." Some people have good reasons for using a system that doesn't allow them to view frames. Try to make your content as accessible as possible to such visitors.

To provide content for browsers that don't support frames:

1 Choose **Modify > Frameset > Edit NoFrames Content**.

Dreamweaver clears the Design view, and the words "NoFrames Content" appear at the top of the Design view.

2 To create the NoFrames content, do one of the following:

- In the Document window, type or insert the content just as you would for an ordinary document.
- Choose **Window > Code Inspector**, place the insertion point between the `body` tags that appear inside the `noframes` tags, and type the HTML code for the content.

3 Choose **Modify > Frameset > Edit NoFrames Content** again to return to the normal view of the frameset document.

Using JavaScript behaviors with frames

There are several JavaScript behaviors and navigation-related commands that are particularly appropriate for use with frames.

Set Text of Frame replaces the content and formatting of a given frame with the content you specify. The content can include any valid HTML. Use this action to dynamically display information in a frame. (See “Set Text of Frame” on page 366.)

Go to URL opens a new page in the current window or in the specified frame. This action is particularly useful for changing the contents of two or more frames with one click. (See “Go to URL” on page 361.)

The **Insert Navigation Bar** command adds a navigation bar to a page; after inserting a navigation bar, you can attach behaviors to its images and set which image displays based on a visitor’s actions. For example, you may want to show a button image in its Up or Down state to let a visitor know which page of a site is being viewed. (See “Inserting a navigation bar” on page 415.)

The **Insert Jump Menu** command lets you set up a menu list of links that open files in a browser window when clicked. You can also target a particular window or frame in which the document will open. (See “Inserting a jump menu” on page 413.)

Part V

Adding Content

Use the visual tools in Dreamweaver to add a variety of content to your web pages. Add and format elements such as text, images, colors, movies, sound, and other forms of media. Be sure to make your pages accessible to visitors with disabilities.

This part contains the following chapters:

- Chapter 19, “Inserting and Formatting Text”
- Chapter 20, “Inserting Images”
- Chapter 21, “Dreamweaver Integration with Other Applications”
- Chapter 22, “Inserting Media”
- Chapter 23, “Dreamweaver and Accessibility”

CHAPTER 19

Inserting and Formatting Text

Macromedia Dreamweaver MX offers several ways for you to add and format text in a document. This chapter describes how to insert text, set font type, size, color, and alignment attributes, as well as how to create and apply your own custom styles using HTML styles and Cascading Style Sheet (CSS) styles.

This chapter covers the following topics:

- “Inserting and formatting HTML text” on page 271
- “Formatting text” on page 273
- “Using HTML styles to format text” on page 279
- “About Cascading Style Sheets” on page 285
- “Converting CSS styles to HTML tags” on page 293
- “Searching and replacing text” on page 294

Inserting and formatting HTML text

Formatting in Dreamweaver is similar to using a standard word processor. Use the Text > Paragraph Format submenu or the Format pop-up menu in the Property inspector to set the default formatting style (Paragraph, Preformatted, Heading 1, Heading 2, and so on) for a block of text. To change the font, size, color, and alignment of selected text, use the Text menu or the Property inspector. To apply text formatting such as bold, italic, code, underline, and so on, use the Text > Style submenu.

You can also combine several standard HTML tags to form a single style, called an HTML style. For example, you can manually apply HTML formatting using a combination of tags and attributes, and save that formatting as an HTML style; it's stored in the HTML Styles panel. The next time you want to format text using that combination of HTML tags, you can simply select the saved style from the HTML Styles panel. HTML styles are supported by almost all web browsers and save time over manually formatting text.

Another kind of style, called a CSS style (CSS stands for Cascading Style Sheets), lets you apply text and page formatting with the advantage of automatic updating. You can store CSS styles directly in the document or, for more power and flexibility, in an external style sheet. If you attach an external style sheet to several web pages, all the pages automatically reflect any changes you make to the style sheet. To access CSS styles, use the CSS Styles panel or the CSS mode of the text Property inspector. For more information about using the text Property inspector to apply HTML or CSS styles, see Setting Text property options in Dreamweaver Help.

Manual HTML formatting and HTML styles apply formatting using standard HTML tags (such as `b`, `i`, `font`, and `code`) that are supported by all popular web browsers. CSS styles define the formatting for all text in a particular class or redefine the formatting for a particular HTML tag (such as `h1`, `h2`, `p`, or `li`). CSS styles are supported only by the web browsers Netscape Navigator 4.0 and later versions, and Microsoft Internet Explorer 4.0 and later versions.

You can use CSS styles, HTML styles, and manual HTML formatting within the same page. Formatting is applied in a hierarchical manner, manual HTML formatting overrides formatting applied by an HTML style or CSS style, and CSS styles embedded in a document override external CSS styles. See “About Cascading Style Sheets” on page 285.

Adding text to a document

There are a number of ways to add text to a Dreamweaver document. You can type text directly in the Dreamweaver document window, or you can cut and paste or import text from other documents.

To add text to your document, do one of the following:

- Type text directly into the Document window.
- Copy text from another application, switch to Dreamweaver, position the insertion point in the Design view of the Document window, and choose Edit > Paste. Dreamweaver doesn't preserve text formatting applied in the other application, but it does preserve line breaks.

Importing text from other documents

You can import tabular data into your document by first saving the files (such as Microsoft Excel files or a database files) as delimited text files. For additional information on importing and formatting table data, see “Importing tabular data” in the Chapter 19, “Inserting and Formatting Text,” on page 271.

You can also import text from Microsoft Word HTML documents. For information on importing Word HTML documents, see “Opening existing documents” on page 111.

To import tabular data:

- 1 Choose File > Import > Import Tabular Data, or choose Insert > Tabular Data.

The Import Table dialog box appears.

Browse for the file you want or enter its name in the text box.

- 2 Select the delimiter used when the file was saved as delimited text. Your options are Tab, Comma, Semicolon, Colon, and Other.

If you select Other, a blank field appears next to the option. Enter the character that was used as a delimiter.

- 3 Use the remaining options to format or define the table into which the data will be imported.
- 4 Click OK when you're done.

Adding space between characters

HTML only allows for one space between characters; to add additional space in a document you must insert a non-breaking space.

You can set a preference to automatically add non-breaking spaces in a document. To set this preference, choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X) and in the General category make sure Allow Multiple Consecutive Spaces is checked.

To insert a non-breaking space, do one of the following:

- In the Insert bar, select Character, then click the Insert Non-breaking Space icon.
- Choose Insert > Special Characters > Non-Breaking Space.
- Press Control+Shift+Spacebar (Windows) or Option+Spacebar (Macintosh).

Adding paragraph spacing

Dreamweaver works similarly to many word processing application: you press Enter (Windows) or Return (Macintosh) to create a new paragraph. Web browsers automatically insert a blank line of space between paragraphs. You can add a single line of space between paragraphs by inserting a line break.

To add a paragraph return:

- Press Enter (Windows) or Return (Macintosh).

To add a line break, do one of the following:

- Press Shift+Enter (Windows) or Shift+Return (Macintosh).
- In the Insert bar, select Character, then click the Line Break icon.
- Choose Insert > Special Characters > Line Break.

Formatting text

You can apply HTML text formatting to one letter or build an entire site using Text > Paragraph Format commands or options in the Property inspector. This kind of manual formatting replaces or overrides formatting set by an HTML style or CSS style.

When you apply HTML text formatting, you use the Property inspector and commands in the Text menu, such as Text > Paragraph Format and Text > Style.

Related topics

“Using HTML styles to format text” on page 279

“Creating an HTML style” on page 282

“Creating a new CSS style” on page 288

Setting and changing fonts and styles

Use options in the Property inspector or the Text menu to set or change font characteristics for selected text. You can set the font type, style (such as bold or italic), and size.

When you use the Property inspector to apply bold or italic style, Dreamweaver automatically applies the `` or `` tag, respectively. If you are designing pages for viewers with 3.0 or older version browsers, you should change this preference in the General category of the Preferences dialog box (Edit > Preferences).

HTML font sizes are relative, not specific, point sizes. Users set the point size of the default font for their browsers; this is the font size that they will see when you select Default or 3 in the Property inspector or Text > Size submenu. Sizes 1 and 2 will appear smaller than the default font size; sizes 4 through 7 will appear larger. Also, fonts generally look larger in Windows than on the Macintosh, though Macintosh Internet Explorer 5 uses the same default font size as Windows.

Tip: One way to ensure consistency with font size is to use CSS styles with your font size set in pixels. For more information on CSS, see “About Cascading Style Sheets” on page 285.

Related topic

“Modifying font combinations” on page 275

To set or change font characteristics:

- 1 Select the text. If no text is selected, the change applies to subsequent text you type.
- 2 Choose from the following options:
 - To change the font, choose a font combination from the Property inspector or from the Text > Font submenu.

Choose Default to remove previously applied fonts; Default applies the default font for the selected text (either the browser default font or the font assigned to that tag in the CSS style sheet).

- To change the font style, click Bold or Italic in the Property inspector, or choose a font style (Bold, Italic, Underline, and so on) from the Text > Style submenu.
- To change the font size, choose a size (1 through 7) from the Property inspector or from the Text > Size submenu.
- To increase or decrease the size of selected text, choose a relative size (+ or -1 to + 4 or -3) from the Property inspector or from either the Text > Size Change submenu.

Note: These numbers indicate a relative difference from the basefont size. The default basefont value is 3. Thus, a +4 value results in a font size of 3 + 4, or 7. The maximum sum for your font size values is 7. If you try to set them higher, they display as 7. Dreamweaver does not display the `basefont` tag (which goes in the `head` section), although the font size should display properly in a browser. To test this, compare text set at 3 and text set at +3.

Modifying font combinations

Use the Edit Font List command to set the font combinations that appear in the Property inspector and the Text > Font submenu.

Font combinations determine how a browser displays text in your web page. A browser uses the first font in the combination that is installed on the user's system; if none of the fonts in the combination are installed, the browser displays the text as specified by the user's browser preferences.

To modify font combinations:

1 Choose Text > Font > Edit Font List.

2 Select the font combination from the list at the top of the dialog box.

The fonts in the selected combination are listed in the Chosen Fonts list in the lower left corner of the dialog box. To the right is a list of all available fonts installed on your system.

3 Choose from the following options:

- To add or remove fonts from a font combination, click the arrows button (<< or >>) between the Chosen Fonts list and the Available Fonts list.
- To add or remove a font combination, click the plus (+) and minus (–) buttons at the top of the dialog box.
- To add a font that is not installed on your system, type the font name in the text field below the Available Fonts list and click the << button to add it to the combination. Adding a font not installed on your system is useful, for example, for specifying a Windows-only font when you are developing pages on a Macintosh.
- To move the font combination up or down in the list, click the arrow buttons at the top of the dialog box.

To add a new combination to the font list:

1 Choose Text > Font > Edit Font List.

2 Select a font from the Available Fonts list and click the << button to move the font to the Chosen Fonts list.

3 Repeat step 2 for each subsequent font in the combination.

To add a font that is not installed on your system, type the font name in the text field below the Available Fonts list and click the << button to add the font to the combination. Adding a font not installed on your system is useful, for example, for specifying a Windows-only font when you are developing pages on a Macintosh.

4 When you have finished selecting specific fonts, select a generic font family from the Available Fonts menu and click the << button to move the generic font family to the Chosen Fonts list.

Generic font families include cursive, fantasy, monospace, sans-serif, and serif. If none of the fonts in the Chosen Fonts list are available on the user's system, the text appears in the default font associated with the generic font family. For example, the default monospace font on most systems is Courier.

Formatting paragraphs

Use the Format pop-up menu in the Property inspector or the Text > Paragraph Format submenu to apply the standard paragraph and heading tags.

When you apply a heading tag to a paragraph, Dreamweaver automatically adds the next line of text as a standard paragraph. To change this setting, choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X), then in the General category, under Editing Options make sure Switch to Plain Paragraph after Heading is unchecked.

Related Topics

“Creating an HTML style” on page 282

Setting Text property options (in Dreamweaver Help)

Defining CSS Type properties (in Dreamweaver Help)

To apply a paragraph or heading tag:

- 1 Place the insertion point in the paragraph, or select some of the text in the paragraph.
- 2 Using the Text > Paragraph Format submenu or the Format pop-up menu in the Property inspector, choose an option:
 - Choose a paragraph format (for example, Heading 1, Heading 2, Preformatted Text, and so on). The HTML tag associated with the selected style (for example, h1 for Heading 1, h2 for Heading 2, pre for Preformatted text, and so on) is applied to the entire paragraph.
 - Choose None to remove a paragraph format.

Aligning text

You align text on the page using the Property inspector or the Text > Align submenu. You can center any element on a page using the Text > Align > Center command.

To align text:

- 1 Select the text you want to align or simply insert the pointer at the beginning of the text.
- 2 Click an alignment option (Left, Right, or Center) in the Property inspector, or choose Text > Align and choose an alignment command.

To center elements:

- 1 Select the element (image, plug-in, table, or other page element) you want to center.
- 2 Choose Text > Align > Center.

Note: You can align and center complete blocks of text; you cannot align or center part of a heading or part of a paragraph.

Indenting text

Using the Indent command applies the `blockquote` HTML tag to a paragraph of text, indenting text on both sides of the page.

To indent text and remove indentation:

- 1 Place the insertion point in the paragraph you want to indent.

- 2 Click the Indent or Outdent button in the Property inspector, choose Text > Indent or Outdent, or choose List > Indent or Outdent from the context menu.

Note: You can apply multiple indents to a paragraph. Each time you choose this command the text indents further from both sides of the document.

Changing the text color

You can change the color of selected text so that the new color overrides the text color set in Page Properties. (If no text color has been set in Page Properties, the default text color is black.)

To change the color of text:

- 1 Select the text.
- 2 Choose from the following options:
 - Choose a color from the palette of browser-safe colors by clicking the color picker in the Property inspector.
 - Choose Text > Color. The system color picker dialog box appears. Select a color and click OK.
 - Enter the color name or hexadecimal number directly in the Property inspector field.
 - To define the default text color, use the Modify > Page Properties command. See “Defining default text colors” on page 115.

To return text to the default color:

- 1 In the Property inspector, click the color box to open the palette of web-safe colors.
- 2 Click the Strike-through button (the white square button with a red line through it, found in the upper right corner).

Creating bulleted and numbered lists

You can create numbered (ordered) lists, bulleted (unordered) lists, and definition lists from existing text or from new text as you type in the Document window. Definition lists do not use leading characters like bullet points or numbers and are often used in glossaries or descriptions. Lists can also be nested. Nested lists are lists that contain other lists. For example, you might want an ordered or bulleted list nested within another numbered or ordered list.

For information about setting a specific list type and other list options for an entire list or a specific list item (for example, reset numbering or use Roman numerals in an ordered list, or to set square bullets), see Setting List property options in Dreamweaver Help.

To create a new list:

- 1 In the Dreamweaver document, place the insertion point where you want to add a list, then do one of the following:
 - Click either the Bulleted or Numbered List button in the Property inspector
 - Choose Text > List and select the type of list desired—Unordered (bulleted) List, Ordered (numbered) List, or Definition List.

The leading character for the specified list item appears in the document window.

- 2 Type the list item text, then press Enter (Windows) or Return (Macintosh) to create another list item.
- 3 To complete the list, press Enter twice (Windows) or press Return twice (Macintosh).

To create a list using existing text:

- 1 Select a series of paragraphs to make into a list.
- 2 Click the Bulleted or Numbered List button in the Property inspector, or choose Text > List and select the type of list desired—Unordered List, Ordered List, or Definition List.

To create a nested list:

- 1 Select the list items you want to nest.
- 2 Click the Indent button in the Property inspector, or choose Text > Indent.
Dreamweaver indents the text and creates a separate list with the original list's HTML attributes.
- 3 Apply a new list type or style to the indented text by following the same procedure used above.

Inserting dates

Dreamweaver provides a convenient Date object, which inserts the current date in whatever format you prefer (with or without the time) and provides the option of updating that date whenever you save the file.

Note: The dates and times shown in the Insert Date dialog box are not the current date, nor do they reflect the dates/times that a visitor sees when they display your site. They are examples only of the way you want to display this information.

To insert the current date into a document:

- 1 In the Document window, place the insertion point where you want the date to be inserted.
- 2 Do one of the following:
 - Choose Insert > Date.
 - In the Insert bar, select Common, then click the Date button.
- 3 In the resulting dialog box, select a format for the name of the day of the week, a format for the date, and a format for the time.
- 4 If you want the inserted date to be updated every time you save the document, select Update Automatically on Save. If you want the date to become plain text when it's inserted, and never update automatically, deselect that option.
- 5 Click OK to insert the date.

Tip: If you have selected Update Automatically on Save, you can edit the date format after it has been inserted into the document by clicking on the formatted text and selecting Edit Date Format in the Property inspector.

Inserting special characters

Certain special characters are represented in HTML by a name or a number, referred to as an **entity**. HTML includes entity names for characters such as the copyright symbol (©), the ampersand (&), and the registered-trademark symbol (®). Each entity has both a name (such as —) and a numeric equivalent (such as —).

Tip: HTML uses the angle brackets < > in its code, but you may need to express the special characters for greater than or less than without Dreamweaver interpreting them as code. In this case, use > for greater than (>) and < for less than (<).

Unfortunately, many browsers (especially older browsers, and browsers other than Navigator and Internet Explorer) don't properly display many of the named entities.

You can insert several special characters (in the form of HTML entities) by choosing the Characters category in the Insert bar.

To insert a special character into a document:

- 1 In the Document window, place the insertion point where you want to insert a special character.
- 2 Do one of the following:
 - Choose the name of the character from the Insert > Characters submenu.
 - In the Insert bar, choose the Characters category and select the character you want.

Tip: There are many other special characters available; to select one of them, choose Insert > Characters > More or select the Insert More Characters icon in the Insert bar, select a character, then click OK.

Using horizontal rules

Horizontal rules (lines) are useful for organizing information. On a page, you can visually separate text and objects with one or more rules.

To create a horizontal rule:

- 1 In the Document window, place the insertion point where you want to insert a horizontal rule.
- 2 Do one of the following:
 - Choose Insert > Horizontal Rule.
 - In the Insert bar, select Common, and then click the Horizontal Rule button.

To modify a horizontal rule:

- 1 In the Document window, select the horizontal rule.
- 2 Choose Window > Properties to open the Property inspector, and modify the properties as desired.

W and **H** specify the width and height of the rule in pixels or as a percentage of the page size.

Align specifies the alignment of the rule (Default, Left, Center, or Right). This setting applies only if the width of the rule is less than the width of the browser window.

Shading specifies whether the rule is drawn with shading. Deselect this option to draw the rule in a solid color.

Using HTML styles to format text

By setting up HTML styles you can quickly and consistently apply font formatting to text in your documents. HTML styles can consist of single or multiple HTML font tag attributes such as color, face, size, as well as other formatting attributes such as bold or italic. For example, you can create an HTML style for text which is Arial, size 4+, green, and italic. You can then quickly apply the style by selecting the text you want it applied to and then selecting the HTML style from the HTML Styles panel.

Since HTML styles consist only of font tags, they can be viewed by browsers that don't support cascading style sheets (CSS), such as 3.0 and earlier browsers.

One disadvantage of HTML styles is that changes you make to an HTML style aren't automatically updated in the document. If you applied a style and later change it, you'll have to reapply the style to the text in order to update the formatting.

Unlike CSS styles, HTML style formatting only affects text that you apply it to, or text that you create using a selected HTML style. If you want the ability to change formatting, and automatically update all instances of that formatting, use CSS styles; see “About Cascading Style Sheets” on page 285.

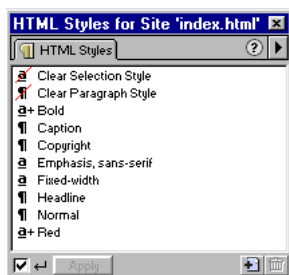
You can use the HTML Styles panel to set up the HTML styles you use in your site, and then share them with other users, local sites, or remote sites. For information, see “Using your HTML styles in other sites” on page 284.

To display the HTML Styles panel, do one of the following:

- Choose Window > HTML Styles.
- Press Control + F11 (Windows) or Command + F11 (Macintosh).
- Click the HTML Styles icon in the Launcher.

Using the HTML styles panel

The HTML Styles panel displays the HTML Styles that are available for the current local site.



There are two types of HTML Styles: Paragraph Styles and Selection Styles. They can be identified by the symbol in the first column of the HTML Styles panel:

Paragraph Styles are preceded by a paragraph mark and allow you to format paragraphs.

Selection Styles are preceded by an **a** and allow you to format text you have selected. The first two styles in the HTML Styles panel, Clear Selection Style and Clear Paragraph Style, let you remove all format tags from the text you select and apply them to.

In the HTML Styles panel above, notice the Bold style. A plus sign (+) preceding a style indicates the style adds formatting to the text it is applied to. If a style doesn't have a plus sign, any existing style is cleared before the selected style is applied. For example, using the HTML Style "a+ Red" simply adds red to any formatting which has already been applied to the selected text, while using the HTML style "Emphasis" clears any existing formatting before applying this style.

To view the style attributes of an existing HTML style:

- 1 In the HTML Styles panel, select a style.
- 2 While in the HTML Styles panel, right-click (Windows) or Control-click (Macintosh) and choose Edit from the context menu, or double-click the HTML style and choose Edit from the context menu.
- 3 In the Define HTML Style dialog box, specify the settings for the style.

The Apply To options determine whether the style applies to the selected text (Selection) or the current text block (Paragraph). The When Applying options determine whether the settings for the style are added to the original text formatting (Add to Existing Style), or removed from the existing formatting and replaced with the new settings (Clear Existing Style).

To apply an existing HTML style:

In the HTML Styles panel, select a style.

- If the Auto Apply checkbox at the bottom of the panel is selected, click the style once.
- If the Auto Apply checkbox is not selected, click the style, then click Apply.

To clear text formatting in your document:

- 1 Select the formatted text.
- 2 In the HTML Styles panel, click Clear Paragraph Style or Clear Selection Style.

Clear Paragraph Style removes any formatting from the current text block in the document.

Clear Selection Style removes any formatting from the selected text.

Note: You can use Clear Paragraph Style and Clear Selection Style to remove any formatting (except CSS formatting), regardless of how the original formatting was applied (for example, through the HTML Styles panel or the Property inspector).

To remove a style from the HTML Styles panel:

- 1 In the HTML Styles panel, deselect the checkbox to turn off the Auto Apply option.
- 2 Select an HTML style.
- 3 Click the Delete Style (trash can) icon in the lower right corner of the panel.

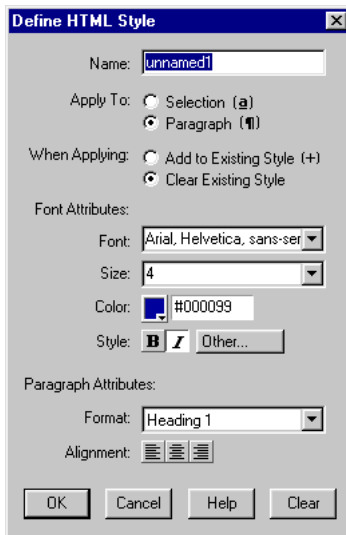
Creating an HTML style

You can create styles in two ways: you can format the text in the document and then create a style based on the selected text or you can create a style in the HTML Styles panel by selecting the formatting attributes you want to apply.

To create a new HTML style:

- 1 In the HTML Styles panel, click the New Style icon; you can also choose Text > HTML Styles > New Style.

The Define HTML Style dialog box appears.



- 2 In the Name text box, enter a name for the style.
- 3 Under Apply To, choose how this style will be applied by doing one of the following:
 - To set a selection style, choose Selection.
 - To set a paragraph style, choose Paragraph.

Note: A paragraph style applies to the entire text block in which the insertion point is located, regardless of what text is actually selected.

- 4 For When Applying, choose whether to apply the HTML style in addition to the existing style of the selected text or paragraph, or to clear the formatting of the selection or paragraph and apply the new HTML style.

Note: The hierarchy for applying styles is as follows: HTML styles take precedence over CSS styles, which take precedence over external CSS styles. For more information, see “About Cascading Style Sheets” on page 285.

- 5 Under Font Attributes, use the pop-up menus to select the formatting options you want to apply.
- 6 If you are creating a paragraph style, under Paragraph Attributes, in the Format pop-up menu, choose a paragraph tag (for example, paragraph, Heading1, Preformatted).

- 7 For Alignment, click the left, center, or right alignment button to set the paragraph alignment you want.
- 8 Click OK.

The style is added to the HTML Styles list.

To create a new HTML style based on existing text:

- 1 In your document, select or create text that has the formatting you want to use for your new HTML style.

Tip: You can use the Property inspector to view and apply formatting, then save the formatting as an HTML style.

- 2 In the HTML Styles panel, click the New Style icon (+) located in the lower right corner.
The Define HTML Style dialog box appears.

- 3 In the Name text box, enter a name for the style.

- 4 Under Apply To, choose how this style will be applied by doing one of the following:

- To set a selection style, choose Selection.
- To set a paragraph style, choose Paragraph.

Note: A paragraph style applies to the entire text block in which the insertion point is located, regardless of what text is actually selected.

- 5 For When Applying, choose whether to apply the HTML style in addition to the existing style of the selected text or paragraph, or to clear the formatting of the selection or paragraph and apply the new HTML style.

Note: The hierarchy for applying styles is as follows: HTML styles take precedence over CSS styles, which take precedence over external CSS styles. For more information, see “About Cascading Style Sheets” on page 285.

- 6 If you are creating a paragraph style, under Paragraph Attributes, in the Format pop-up menu, choose a paragraph tag (for example, paragraph, Heading1, Preformatted).
- 7 For Alignment, click the left, center, or right alignment button to set the paragraph alignment you want.
- 8 Click OK.

Applying an HTML style

Applying a style is as easy as selecting the text or paragraph you want the style applied to, and then selecting the style in the HTML Styles panel.

To apply an HTML Style:

- 1 In the lower left corner of the HTML Styles panel make sure the Apply option is selected, to automatically apply the style you select.
- 2 In the Document window do one of the following to select the text you want the style applied to:
 - Place the insertion point anywhere in a paragraph to apply a paragraph style.
 - Use the cursor to select the text you want a selection style applied to.
- 3 In the HTML Styles panel, click the HTML style you want to apply to the text.

The text automatically updates in the Document window.

Editing an HTML style

When you edit an HTML style, Dreamweaver does not automatically update text that was previously formatted using the HTML style. To update the style to previously formatted text, you must manually re-apply the style.

To edit an existing HTML style:

- 1 Make sure no text is selected in the Document window.
- 2 In the HTML Styles panel, make sure the Auto Apply checkbox is deselected.
If the Auto Apply option is turned on, the HTML style will be applied when you select it in the HTML Styles panel.
- 3 In the HTML Styles panel, do one of the following:
 - Right-click (Windows) or Control-click (Macintosh) the style, then choose Edit from the context menu.
 - Double-click the style.
The Define HTML Style dialog box appears.
- 4 In the dialog box, set style attribute options for the style.
- 5 To reset it to the default options, click Clear.

Using your HTML styles in other sites

You can use the HTML Styles panel to record the HTML styles you use in your site, and then share them with other users, local sites, or remote sites.

HTML styles are automatically stored in your local site's Library folder in a file named Styles.xml. You can copy the Styles.xml file from the Library folder of one local site to the Library folder of another local site and reuse styles you create.

Note: Each site can only contain one Styles.xml file; therefore, if you created new HTML styles in the site to which you are copying a Styles.xml file, the file you copy there will replace the existing file. You won't lose any formatting changes you've already applied, and you can recreate the styles by selecting text in the document and defining a new HTML style.

To share your HTML styles with other sites or users:

- 1 Choose Window > Site or Site > Site Files to open the Site panel in Site Files view.
- 2 In the Site panel, locate and then open the Library folder.
You'll notice a file called styles.xml. This file contains all your HTML styles for the site. You can put, get, check in, check out, and copy this file as you would any other file in your site. You can also create Design Notes for the styles.xml file. You must first check out the styles.xml file before you can create or edit a style for a remote site.

For more information on using these options, see "Setting up a remote folder" on page 63.

About Cascading Style Sheets

Cascading Style Sheets (CSS) are a collection of formatting rules which control the appearance of content in a web page. With CSS styles you have great flexibility and control of the exact page appearance, from precise positioning of layout to specific fonts and styles.

CSS styles let you control many properties that cannot be controlled using HTML alone. For example, you can assign custom list bullets and specify different font sizes and units (pixels, points, and so on). By using CSS styles and setting font sizes in pixels, you can ensure a more consistent treatment of your page layout and appearance in multiple browsers. In addition to text formatting, you can control the format and positioning of a block-level elements in a web page. For example, you can set margins, borders, float text around other text, and so on.

A CSS style rule consists of two parts—the selector and the declaration. The selector is the name of the style (such as `TR`, or `P`) and the declaration defines what the style elements are. The declaration consists of two parts, the property (such as `font-family`), and value (such as `Helvetica`). The term cascading refers to your ability to apply multiple style sheets to the same web page. For example, you can create one style sheet to apply color and another to apply margins, and apply them both to the same page to create the design you want.

A major advantage of CSS styles is that they provide easy update capability; when you update a CSS style, the formatting of all the documents that use that style are automatically updated to the new style.

In Dreamweaver, you use the CSS Styles panel to create, view, and attach style attributes to your documents. For information about using the CSS Styles panel, see “Using the CSS Styles panel” on page 286. In addition to styles and style sheets you create, you can use style sheets that come with Dreamweaver to apply styles to your documents. See “Creating a document based on a Dreamweaver design file” on page 109.

You can define the following types of CSS style sheets in Dreamweaver:

- Custom CSS styles, also called class styles, let you set style attributes to any range or block of text. See “Applying a custom (class) CSS style” on page 289.
- HTML tag styles redefine the formatting for a particular tag, such as `h1`. When you create or change a CSS style for the `h1` tag, all text formatted with the `h1` tag is immediately updated.
- CSS selector styles redefine the formatting for a particular combination of tags (for example, `td h2` applies whenever an `h2` header appears inside a table cell) or for all tags that contain a specific `id` attribute (for example, `#myStyle` applies to all tags that contain the attribute-value pair `ID="myStyle"`).

CSS style sheets reside in the `head` area of a document. CSS styles can define the formatting attributes for HTML tags, ranges of text identified by a `class` attribute. Dreamweaver MX recognizes styles defined in existing documents as long as they conform to CSS style guidelines.

Tip: To display the O'Reilly CSS reference guide included with Dreamweaver, click the Reference button found on the toolbar and choose O'Reilly CSS Reference from the pop-up menu.

Manual HTML formatting overrides formatting applied with CSS (or HTML) styles. For CSS styles to control the formatting of a paragraph, you must remove all manual HTML formatting or HTML styles.

Most styles attributes you apply can be viewed in the Document window. You can also preview the document in a browser window to see it applied. Some of the CSS style attributes are rendered differently in Microsoft Internet Explorer 4.0 and Netscape Navigator 4.0, and some are not currently supported by any browser.

While working in the CSS Styles panel, you can use the Design Time Style Sheet feature of Dreamweaver. This feature allows you to hide or show style sheet attributes while you are designing a page in Dreamweaver (for example to view a document with a style sheet designed for Navigator or for Internet Explorer). For information about using Design Time Style Sheets see “Using Design Time style sheets” on page 291.

Using the CSS Styles panel

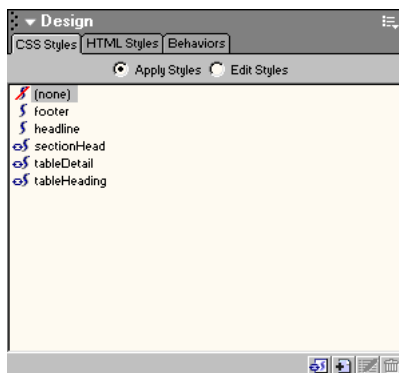
You use the CSS Styles panel to create, view properties of, and apply CSS styles to elements in a document.

To open the CSS Styles panel, do one of the following:

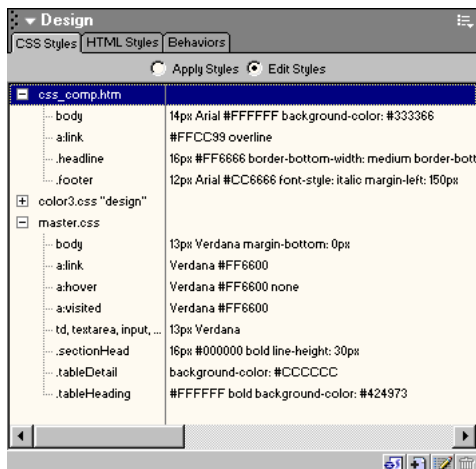
- Choose Window > CSS Styles.
- Press Shift+F11.

The Apply Styles and Edit Styles radio buttons located at the top of the CSS Styles panel let you select different views of the CSS styles associated with the current document.

You use the Apply Styles view to select a class style to apply to an element in your document. The Apply Styles view only displays custom (class) styles. Redefined HTML styles and selector styles do not appear in this pane. Since they are associated with an HTML tag, their style attributes are automatically applied to any tags in the document affected by the defined style. For example, if you define style attributes for the table tag, a table in your document will automatically update with the style definitions you selected.



The Edit Styles view allows you to look at the style definition of the styles associated with the current document. The Edit Styles view displays style definition of custom (class) CSS styles, redefined HTML tags, and CSS selector styles. You also use the Edit Styles view to view Design Time style sheets you have applied to the document.



As you create or attach a CSS style sheet, the style's name and attributes appear in the CSS Styles panel's Edit Styles view. Edit Styles lists all the selectors defined in all style tags and externally linked or imported style sheets. Apply Styles lets you view styles applied in the current document, as well as styles available in an attached external style sheet.

When you create a custom (class) style, it appears in the CSS Styles panel and in the Text > CSS Styles submenu.

The following buttons are located at the bottom of the CSS Styles panel:



Attach Style Sheet opens the Link External Style Sheet dialog box. Select an external style sheet to link to or import into your current document. For information about attaching an external style sheet, see “Creating and linking to an external CSS style sheet” on page 289.

New CSS Style opens the New CSS Style dialog box. You use the New CSS Styles dialog to select the type of style you're creating—for example, to create a class style, redefine an HTML tag, or to define a CSS selector. For information about creating a new style, see “Creating a new CSS style” on page 288.

Edit Style Sheet opens the CSS Style Definition dialog box. Edit any of the styles in the current document or in an external style sheet. For information about updating a style sheet, see “Editing a CSS style” on page 290.

Delete CSS Style removes the selected style from the CSS Styles panel, and removes the formatting from any element to which it was applied.

Note: Right-click (Windows) or Control-click (Macintosh) the CSS Styles panel to open a context menu of options for working with CSS Style Sheet commands.

Creating a new CSS style

Create a CSS style to automate the formatting of HTML tags or a range of text identified by a `class` attribute.

To create a new CSS style:

1 Place the insertion point in the document, then do one of the following to open the New CSS Style dialog box:

- In the CSS Styles panel (Window > CSS Styles), click the New CSS Style button (+) located in the lower right area of the panel.
- In the text Property inspector, click the Toggle CSS/HTML Mode button, if necessary, to switch to CSS Mode, then in the CSS Style pop-up menu, select New CSS Style.
- Choose Text > CSS Styles > New CSS Style.

The New CSS Style dialog box appears.

2 Define the type of CSS style you want to create:

- To create a custom style that can be applied as a `class` attribute to a range or block of text, select Make Custom Style (Class), then in the Name field, enter a name for the style.

Note: Custom style (class) names must begin with a period and can contain any combination of letters and numbers. For example, `.myhead1`. If you don't enter a beginning period, Dreamweaver automatically enters it for you.

- To redefine the default formatting of a specific HTML tag, select Redefine HTML Tag, then in the Tag field, enter an HTML tag or choose one from the pop-up menu.
- To define the formatting for a particular combination of tags or for all tags that contain a specific `Id` attribute, select Use CSS Selector, then in the Selector field, enter one or more HTML tags or choose one from the pop-up menu. The selectors available from the pop-up menu are `a:active`, `a:hover`, `a:link`, and `a:visited`.

3 Select the location in which the style will be defined:

- To create an external style sheet, choose New Style Sheet File.
- To embed the style in the current document, choose This Document Only.

4 Click OK.

The Style Definition dialog box appears.

5 Choose the style options you want to set for the new CSS style.

6 When you are done setting style attributes, click OK.

For information on specific CSS style settings, see the following topics in Dreamweaver Help:

- Defining CSS Type properties
- Defining CSS Style Background properties
- Defining CSS Style Block properties
- Defining CSS Style Box properties
- Defining CSS Style Border properties
- Defining CSS Style List properties
- Defining CSS Style Positioning properties
- Defining CSS Style Extensions properties

Applying a custom (class) CSS style

Custom (class) CSS styles are the only type of CSS style that can be applied to any text in a document, regardless of which tags control the text. All custom (class) styles associated with the current document are displayed in the Apply Style view of the CSS Styles panel and in the CSS mode of the text Property inspector.

You'll see most styles updated immediately, however, you should preview your page in a browser to verify a style was applied as expected. When you apply two or more CSS styles to the same text, the styles may conflict and produce unexpected results. See “About conflicting styles” on page 292 for more information.

To apply a custom CSS style:

- 1 In the document, select the text to which you want to apply a CSS style.

Place the insertion point in a paragraph to apply the style to the entire paragraph.

If you select a range of text within a single paragraph, the CSS style affects only the selected range.

To specify the exact tag to which the CSS style should be applied, select the tag in the tag selector located at the bottom left of the Document window.

- 2 To apply a class style, do one of the following:

- In the CSS Styles panel (Window > CSS Styles), select Apply Styles, then in the CSS Styles list, click the name of the style you want to apply.
- In the text Property inspector, click the Toggle CSS/HTML Mode button, if necessary, to switch to CSS Mode, then in the CSS Style pop-up menu, select the class style you want to apply.
- In the Document window, right-click (Windows) or Control-click (Macintosh) the selected text, and in the context menu choose CSS Styles and then select the style you want to apply.
- Select Text > CSS Styles, and in the submenu select the style you want to apply.

To remove a custom style from a selection:

Select the object or text you want to remove the style from, then do one of the following:

- In the text Property inspector, click the Toggle CSS/HTML Mode button, if necessary, to switch to CSS Mode, then in the CSS Style pop-up menu, select No CSS Style.
- In the CSS Styles panel, select Apply Styles view, then choose No CSS Style.

Creating and linking to an external CSS style sheet

A CSS style sheet is an external text file containing styles and formatting specifications. When you edit an external CSS style sheet, all documents linked to that CSS style sheet are updated to reflect those edits. You can export the CSS styles found in a document to create a new CSS style sheet, and attach or link to an external style sheet to apply the styles found there.

In addition to CSS style sheets you create, you can use CSS style sheets that ship with Dreamweaver to attach to pages in a site. For information about using the design style sheets that come with Dreamweaver, see “Creating a document based on a Dreamweaver design file” on page 109.

For information about applying a style, see “Applying a custom (class) CSS style” on page 289.

To link to or import an external CSS style sheet:

1 Open the CSS Styles panel by doing one of the following:

- Choose Window > CSS Styles
- Press Shift + F11
- Click the CSS Styles icon in the Launcher.

2 In the CSS Styles panel, click the Attach Style Sheet button.

The Link External Style Sheet appears.

3 In the Link External Style Sheet dialog box, do one of the following:

- Click Browse (Windows) or Choose (Macintosh) to browse to an external CSS style sheet, or type the path to the style sheet in the File/URL box.

4 In Add As, select one of the options:

- To create a link between the current document and an external style sheet, choose Link. This creates a link href tag in the HTML code, and references the URL where the published style sheet is located. This method is supported by both Microsoft Internet Explorer and Netscape Navigator.
- To reference an external style sheet, choose Import. This creates an @import tag in the HTML code, and references the URL where the published style sheet is located. This method does not work with Netscape Navigator.

5 Click OK

The name of the external CSS style sheet appears in the Edit Styles view of the CSS Styles panel, and custom (class) styles appear in the Apply Styles view preceded by the external style sheet identifier.

Editing a CSS style

You can easily edit both internal and external styles you have applied to a document. In Edit Styles view, select the style you want to change, then open the CSS Style Definition dialog box and modify the style.

When you edit a CSS style sheet that controls the text in your document, you instantly reformat all of the text controlled by that CSS style sheet. Edits to an external style sheet affect all the documents linked to it.

You can set an external editor for editing style sheets. For information about setting up an external editor, see “Launching an external editor for media files” on page 320.

Note: If you have installed TopStyle, a CSS editor from Bradbury Software on your computer, Dreamweaver automatically detects it and sets it as your external editor for .css files. An evaluation copy of Topstyle comes with Dreamweaver.

To edit a CSS style:

- 1 In the CSS Styles Panel, click the Edit Styles radio button to view the styles in the current document.
- 2 Click the style you want to edit to select it, then do one of the following:
 - Double-click the selected style.
 - Right-click (Windows) or Control-click (Macintosh), then choose Edit.
 - Click the Edit Style Sheet button located at the bottom of the CSS Styles panel.
The CSS Style Definition dialog box opens.
- 3 Modify the style as desired, then click OK.

Editing a CSS Style Sheet

A CSS style sheet typically includes one or more styles. You can edit an individual style in a CSS style sheet (see “Editing a CSS style” on page 290), but sometime you want to edit several styles.

The Edit Style Sheet dialog box lets you work with style sheets in a number of ways. Use it to link to an external CSS style sheet, create a new CSS style sheet, edit an existing CSS style sheet, or to duplicate or remove a CSS style sheet.

To edit a CSS style sheet:

- 1 In the CSS Styles panel, select Edit Styles.
- 2 In the Styles list, click the style sheet you want to edit to select it, then do one of the following:
 - Click the Edit Style Sheets button located at the bottom of the CSS Styles panel.
 - Right-click the style sheet, then choose Edit Style Sheet, then in the dialog box that appears, select the style sheet you want to edit, and then click Edit.
The styles for the selected CSS style sheet appear in the dialog box.
- 3 In the dialog box, select the style you want to edit, then click Edit.
The CSS Style Definition dialog box appears.
- 4 Modify the styles as desired, then click OK.

Using Design Time style sheets

Design Time style sheets allow you to show or hide design applied by a CSS style sheet as you work in a Dreamweaver document. For example, you can use this option to include or exclude the effect of a Macintosh-only or a Windows-only style sheet as you design a page.

Design Time style sheets only apply while you are designing working in the Dreamweaver document; when the page is displayed in a browser window, only the styles that are actually attached to or embedded in the document appear in a browser.

To show or hide a CSS style sheet at design time:

- 1 Open the Design Time Style Sheets dialog box by doing one of the following:
 - Right-click in the CSS Styles panel, in the context menu select Design Time Style Sheet.
 - Choose Text > CSS Styles > Design Time Style Sheets.

- 2 In the dialog box, set options to show or hide a selected style sheet.
 - To display a CSS style sheet at design time, click the (+) plus button above Show Only at Design Time, then in the Select a Style Sheet dialog box, browse to the CSS style sheet you want to show.
 - To hide a CSS style sheet, click the (+) plus button above Hide at Design Time, then in the Select a Style Sheet dialog box, browse to the CSS style sheet you want to show.
 - To remove a style sheet from either list, click the style sheet you want to remove, then click the appropriate (–) minus button.
- 3 Click OK to close the dialog.

The CSS Styles panel updates with the selected style sheet's name along with an indicator, "hidden" or "design," to reflect the style sheet's status.

Exporting styles to create a CSS style sheet

You can export styles from a document to create a new CSS style sheet. You can then link to other documents to apply these styles.

To export CSS styles from a document and create a CSS style sheet:

- 1 Choose File > Export > Export CSS Styles or choose Text > Export Style Sheet.

The Export Styles as CSS File dialog box appears.

- 2 Enter a name for your style and click Save.

The style is saved as a CSS style sheet.

About conflicting styles

When you apply two or more CSS styles to the same text, the styles may conflict and produce unexpected results. Browsers apply style attributes according to the following rules:

- If two styles are applied to the same text, the browser displays all attributes of both styles unless specific attributes conflict. For example, one style may specify blue as the text color and the other style may specify red.
- If attributes from two styles applied to the same text conflict, the browser displays the attribute of the innermost style (the style closest to the text itself).
- If there is a direct conflict, the attributes from CSS styles (styles applied with the `class` attribute) overrule attributes from HTML tag styles.

In the example that follows, the style defined for `h1` might specify the font, size, and color for all `h1` paragraphs, but the custom CSS style `.Blue` applied to this paragraph overrules the color setting in the `H1` style. The second custom CSS style `.Red` overrules `.Blue` because it is inside the `.Blue` style.

```
<h1><span class="Blue">This paragraph is controlled by the .Blue custom style  
and h1  
HTML tag style.<span class="Red">Except this sentence is controlled by the .Red  
style.</span>  
Now we're back to the .Blue style.</span></h1>
```

Converting CSS styles to HTML tags

If you have used CSS styles to specify text formatting (such as the family, size, color, and decoration of fonts) and later decide that you want to make the formatting viewable on a 3.0 browser, you can use the File > Convert > 3.0 Browser Compatible command to convert as much of the style information as possible to HTML tags.

Note: Not all CSS styles can be converted to HTML, because HTML tags do not cover or support all the attributes possible in CSS.

To convert a file that uses CSS styles to a 3.0 browser-compatible file:

1 Choose File > Convert > 3.0 Browser Compatible.

2 In the dialog box that appears, select CSS Styles to HTML Markup.

When you choose the Layers to Tables option, Dreamweaver replaces all layers with a single table that preserves the original positioning.

CSS styles are replaced, where possible, with HTML tags such as `b` and `font`. Any CSS markup that cannot be converted to HTML is removed. See CSS to HTML markup conversion table for information on which styles are converted and which are removed.

3 Click OK. Dreamweaver opens the converted file in a new, untitled window.

Note: You must perform this conversion procedure each time you change the original file in order to update the 3.0-compatible file. For this reason, it's best to perform this procedure only after you are completely satisfied with your original file.

CSS to HTML markup conversion table

The CSS attributes listed in the following table are converted to HTML markup with the File > Convert > 3.0 Browser Compatible command. (See “Converting CSS styles to HTML tags” on page 293.) Attributes not listed in the table are removed.

CSS attribute	Converted to
color	FONT COLOR
font-family	FONT FACE
font-size	FONT SIZE="[1-7]"
font-style: oblique	I
font-style: italic	I
<i>font-weight</i>	B
list-style-type: square	UL TYPE="square"
list-style-type: circle	UL TYPE="circle"
list-style-type: disc	UL TYPE="disc"
list-style-type: upper-roman	OL TYPE="I"
list-style-type: lower-roman	OL TYPE="i"
list-style-type: upper-alpha	OL TYPE="A"
list-style-type: lower-alpha	OL TYPE="a"
list-style	UL or OL with TYPE as appropriate

CSS attribute	Converted to
text-align	P ALIGN or DIV ALIGN as appropriate
text-decoration: underline	U
text-decoration: line-through	STRIKE

Checking spelling

Use the Check Spelling command in the Text menu to check the spelling in the current document. The Check Spelling command ignores HTML tags and attribute values.

By default, the spelling checker uses the U.S. English spelling dictionary. To change the dictionary, choose Edit > Preferences > General or Dreamweaver > Preferences > General (Mac OS X), then in the Spelling Dictionary pop-up menu select the dictionary you want to use. Dictionaries for additional languages can be downloaded from the Dreamweaver Support Center.

To check and correct spelling:

- 1 Choose Text > Check Spelling or press Shift+F7.

When Dreamweaver encounters an unrecognized word the Check Spelling dialog box appears.

- 2 In the Check Spelling dialog box, choose the appropriate option based on how you want the discrepancy handled:

Add to Personal adds the unrecognized word to your personal dictionary.

Ignore ignores this instance of the unrecognized word.

Ignore All ignores all instances of the unrecognized word.

Change replaces this instance of the unrecognized word with text that you type in the Change To text box or with the selection in the Suggestions list.

Change All replaces all instances of the unrecognized word in the same manner.

Searching and replacing text

You can search the current document, selected files, a directory, or an entire site for text, text surrounded by specific tags, or HTML tags and attributes. You use different commands to search for files, and to search for text and/or HTML 1 tags within files: Locate in Local Site and Locate in Remote Site search for files, while Edit > Find and Replace search for text and tags within files.

To search for text and/or HTML within documents:

- 1 Choose from the following options:

- In Design view, from the Document window or Site panel, choose Edit > Find and Replace.
- While in Code view, right-click (Windows) or Control-click (Macintosh) and choose Find and Replace from the context menu.

- 2 In the Find or Replace dialog box that appears, use the Find In option to specify which files to search:

- Current Document confines the search to the active document. This option is available only when you choose Find or Replace with the Document window active, or from the context menu in the Code inspector.

- Entire Local Site expands the search to all the HTML documents, library files, and text documents in the current site. After you choose Current Site, the name of the current site appears to the right of the pop-up menu. If this is not the site you want to search, choose a different site from the current site's pop-up menu in the Site panel.
 - Selected Files in Site confines the search to the files and folders that are currently selected in the Site panel. This option is available only when you choose Find or Replace with the Site panel active (that is, in front of the Document window).
 - Folder confines the search to a specific group of files. After choosing Folder, click the folder icon to browse to and select the directory you want to search.
- 3** Use the Search For option to specify the kind of search you want to perform.
- Source Code lets you search for specific text strings in the HTML source code. See “Searching and replacing tags and attributes” on page 196.
 - Text lets you search for specific text strings in the Document window. A text search ignores any HTML that interrupts the string. For example, a search for **the black dog** would match both the black dog and the `<i>black</i>` dog.
 - Text (Advanced) lets you search for specific text strings that are either within or not within a tag or tags. For example, in a document that contains the following HTML, searching for **tries not inside i** would find only the second instance of the word `tries`: `John <i>tries</i> to get his work done on time, but he doesn't always succeed. He tries very hard.` See “Searching for text contained in specific tags” on page 197.
 - Specific Tag lets you search for specific tags, attributes, and attribute values, such as all `td` tags with `valign` set to `top`. See “Searching for tags and attributes” on page 197.

Note: Pressing Control+Enter or Shift+Enter (Windows), or Control+Return, Shift+Return, or Command+Return (Macintosh), adds line breaks within the text search fields, allowing you to search for a Return character. Make sure to deselect the Ignore Whitespace Differences option when performing this search, if you're not using regular expressions. Note that this finds a Return character in particular, not simply the occurrence of a line break; for instance, it doesn't find a `
` tag or a `<p>` tag. Return characters appear as spaces in the Design view, not as line breaks.

4 Use the following options to expand or limit the search:

- The Match Case option limits the search to text that exactly matches the case of the text you want to find. For example, if you search for **the brown derby**, you will not find `The Brown Derby`.

Note: The Ignore Whitespace Differences option, when selected, treats all whitespace as a single space for the purposes of matching. For example, with this option selected, **this text** would match `this text` but not `this text`. This option is not available when the Use Regular Expressions option is selected; you must explicitly write your regular expression to ignore whitespace. Note that `<p>` and `
` tags do not count as whitespace.

- The Use Regular Expressions option causes certain characters and short strings (such as `?`, `*`, `\w`, and `\b`) in your search string to be interpreted as regular expression operators. For example, a search for **the b\w*\b dog** will match both the black dog and the barking dog. See “About regular expressions” on page 199.

Note: If you are working in Code view and make changes to your document, and try to find and replace anything other than source code, a dialog box appears letting you know that Dreamweaver is synchronizing the two views before doing the search. For more information on synchronizing views, see “Viewing your code” on page 175.

CHAPTER 20

Inserting Images

Images are commonly used to add graphical interfaces (such as navigation buttons), visual appeal (for example, photographs), or interactive design elements, such as rollover images or an image map.

In Macromedia Dreamweaver MX, you can work in Design view or Code view to insert images in a document. As you add images in a Dreamweaver document, you can set or modify image properties and view the changes directly in the Document window.

To set up an efficient web design workflow, you can select an image editor preference, and automatically launch it to edit images while you work in Dreamweaver.

This chapter contains the following sections:

- “About images” on page 297
- “Inserting an image” on page 298
- “Resizing an image” on page 301
- “Creating a rollover image” on page 301
- “Using an external image editor” on page 302
- “Applying behaviors to images” on page 303

About images

Many different types of graphic file formats exist, but three graphic file formats are generally used in web pages—GIF, JPEG, and PNG. Currently, GIF and JPEG file formats are the best supported and can be viewed by most browsers.

PNG files are best suited for almost any type of web graphic due to their flexibility and small file size; however, the display of PNG images is only partially supported in Microsoft Internet Explorer (4.0 and later browsers) and Netscape Navigator (4.04 and later browsers). So unless you are designing for a specific target audience using a browser that supports the PNG format, use GIFs or JPEGs for broader appeal.

GIF (Graphic Interchange Format) files use a maximum of 256 colors, and are best for displaying noncontinuous-tone images or those with large areas of flat colors, such as navigation bars, buttons, icons, logos, or other images with uniform colors and tones.

JPEG (Joint Photographic Experts Group) file format is the superior format for photographic or continuous-tone images, because JPEG files can contain millions of colors. As the quality of a JPEG file increases, so does the file size and the file download time. You can often strike a good balance between the quality of the image and the file size by compressing a JPEG file.

PNG (Portable Network Group) file format is a patent-free replacement for GIFs that includes support for indexed-color, grayscale, and true-color images, and alpha channel support for transparency. PNG is the native file format of Macromedia Fireworks MX. PNG files retain all the original layer, vector, color, and effects information (such as drop shadows), and all elements are fully editable at all times. Files must have the .png file extension to be recognized as PNG files by Macromedia Dreamweaver MX.

Note: PNG support in browsers is inconsistent; therefore, if you work with PNG files, you should also export them as GIFs or JPEGs to ensure you have web-ready versions of them.

Inserting an image

When you insert an image into a Dreamweaver document, Dreamweaver automatically generates a reference to the image file in the HTML source code. To ensure that this reference is correct, the image file must be in the current site. If it is not in the current site, Dreamweaver asks whether you want to copy the file into the site.

You can also add images as dynamic content; for information, see “Making images dynamic” on page 521.

To insert an image:

1 Place the insertion point where you want the image to appear in the Document window, then do one of the following:



- In the Common category of the Insert bar, click the Image icon.
- In the Common category of the Insert bar, drag the Image icon to the Document window (or to the Code view window if you are working in the code).
- Choose Insert > Image.
- Drag an image from the Assets panel (Window > Assets) to the desired location in the Document window; then skip to step 3.
- Drag an image from the Site panel to the desired location in the Document window; then skip to step 3.
- Drag an image from the desktop to the desired location in the Document window; then skip to step 3.

2 In the dialog box that appears, do one of the following:

- Choose File System to choose a graphic file.
- Choose Data Source to choose a dynamic image source.

3 Browse to choose the image or content source you want to insert.

If you are working in an unsaved document, Dreamweaver generates a file:// reference to the image file. When you save the document anywhere in the site, Dreamweaver converts the reference to a document-relative path.

4 In the Property inspector (Window > Properties), set properties for the image.

For more information, see Setting image properties in Dreamweaver Help.

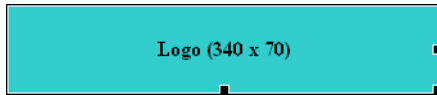
Related Topics

“Setting a background image or background page color” on page 113

“Using spacer images” on page 252

Inserting an image placeholder

An image placeholder is a graphic you use until final artwork is ready to be added to a web page. You can define certain attributes of the placeholder: you can set the placeholder's size and color, as well as provide it with a text label. An image placeholder's color, size attributes, and label appear when the image placeholder is viewed in the Dreamweaver Document window.



When viewed in a browser window the label and size text do not appear.

To insert an image placeholder:

1 In the Document window, place the insertion point where you want to insert a placeholder graphic.

2 Do one of the following:



- In the Insert bar, select Common then click the Image Placeholder icon.
- In the Insert bar, select Common, then drag the Image Placeholder icon to the Document window.
- Choose Insert > Image Placeholder.

The Image Placeholder dialog box appears.

3 In the dialog box, select options for the image placeholder.

For detailed information about setting image placeholder options, click the Help button in the dialog box.

4 Click OK.

Related topics

“Replacing an image placeholder” on page 299

“Resizing an image” on page 301

“Updating a Dreamweaver image placeholder in Fireworks” on page 308.

Replacing an image placeholder

An image placeholder is not a graphic image that displays in a browser. Before you publish your site you should replace any image placeholders you've added with web-friendly graphic files such as GIFs or JPEGs.

If you have Fireworks MX, you can create a new graphic from the Dreamweaver image placeholder. The new image is set to the same size as the placeholder image. You can edit the image, then replace it in Dreamweaver. For information about creating a replacement image in Fireworks MX, see “Updating a Dreamweaver image placeholder in Fireworks” on page 308.

To update the image source:

1 In the Document window, do one of the following:

- Double-click the image placeholder
- Click the image placeholder to select it, then in the Property inspector (Window > Properties), click the folder icon next to the Src field

The Image Source dialog box appears.

2 In the dialog box, navigate to the image you want to replace the image placeholder with.

3 Click OK.

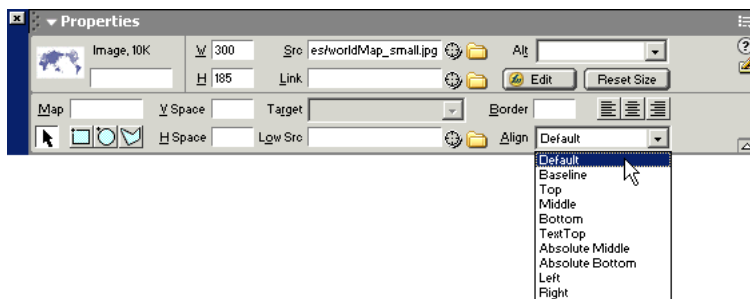
The selected image appears in the document.

Aligning an image

Use the image Property inspector to set the alignment of an image in relation to other elements in the same paragraph or line.

Note: HTML does not provide a way to wrap text around the contours of an image, as you can with some word processing applications

You can align an image to text, another image, a plug-in, or other elements in the same line. You can also use the alignment buttons (left, right, and center) to set the horizontal alignment of an image.



Default generally specifies a baseline alignment. (The default may vary depending on the site visitor's browser.)

Baseline and **Bottom** align the baseline of the text (or other element in the same paragraph) to the bottom of the selected object.

Top aligns the top of an image to the top of the tallest item (image or text) in the current line.

Middle aligns the middle of the image with the baseline of the current line.

TextTop aligns the top of the image with the top of the tallest character in the text line.

Absolute Middle aligns the middle of the image with the middle of the text in the current line.

Absolute Bottom aligns to the bottom of the image with the bottom of the line of text (which includes descenders, as in the letter *g*).

Left places the selected image on the left margin, wrapping text around it to the right. If left-aligned text precedes the object on the line, it generally forces left-aligned objects to wrap to a new line.

Right places the image on the right margin, wrapping text around the object to the left. If right-aligned text precedes the object on the line, it generally forces right-aligned objects to wrap to a new line.

Resizing an image

You can visually resize elements such as images, plug-ins, Macromedia Shockwave or Flash movies, applets, and ActiveX controls in the Design View of the Dreamweaver Document window. Visual resizing helps you determine how an element affects the layout at different dimensions.

Resizing changes the `width` and `height` attributes of the element back to their original size. The W and H fields in the Property inspector display the current width and height of the element as you resize it. The element's file size does not change.

Bitmap elements such as GIF, JPEG, and PNG images may become distorted if you increase or decrease their `width` and `height` attributes. To maintain the same aspect ratio, hold down the Shift key while you resize a bitmap. However, it is recommended that you visually resize bitmaps in Dreamweaver only to determine the layout. After you have determined the ideal size for your image, edit the file in an image-editing application. Editing the image may also reduce its file size and thus reduce download time.

To resize an element:

- 1 Select the element (for example, an image or Shockwave movie) in the Document window.
Resize handles appear at the bottom and right sides of the element and in the bottom right corner. If resize handles don't appear, either click away from the element you want to resize and then reselect it, or click the appropriate tag in the tag selector to select the element.
- 2 Resize the element by doing one of the following:
 - To adjust the width of the element, drag the selection handle on the right side.
 - To adjust the height of the element, drag the bottom selection handle.
 - To adjust the width and the height of the element at the same time, drag the corner selection handle.
 - To preserve the element's proportions (its width-to-height ratio) as you adjust its dimensions, Shift-drag the corner selection handle.

Elements can be visually resized to a minimum of 8 x 8 pixels. To adjust the width and height of an element to a smaller size (for example, 1 x 1 pixel), use the Property inspector to enter a numeric value.

To return a resized element to its original dimensions, in the Property inspector, delete the values in the W and H fields, or click the Reset Size button.

Creating a rollover image

A rollover is an image that, when viewed in a browser, changes when the pointer moves across it. A rollover actually consists of two images: the primary image (the image displayed when the page first loads) and a secondary image (the image that appears when the pointer moves over the primary image). Both images in a rollover should be the same size; if the images are not the same size, Dreamweaver automatically resizes the second image to match the properties of the first image.

You cannot see the effect of a rollover image in the Dreamweaver Document window. To see the rollover effect, press F12 to preview the page in a browser, then roll the pointer over the image.

Rollover images are automatically set to respond to the onmouseover event. For information about setting an image to respond to a different event (for example, a mouse click) or about editing a rollover to display a different image, see “Swap Image” on page 373.

A more complex form of rollover image is a navigation bar. To create a navigation bar, use the Insert > Interactive Images > Navigation Bar command (see “Inserting a navigation bar” on page 415).

To create a rollover:

- 1 In the Document window, place the insertion point where you want the rollover to appear.
- 2 Insert the rollover using one of these methods:
 - In the Insert bar, select Common, then click the Rollover Image icon.
 - In the Insert bar, select Common, then drag the Rollover Image icon to the desired location in the Document window.
 - Choose Insert > Interactive Images > Rollover Image.

The Insert Rollover Image dialog box appears.

- 3 In the Image Name text box, type a name for the rollover.
- 4 In the Original Image text box, click Browse and select the image you want displayed when the page loads, or enter the image file’s path in the text box.
- 5 In the Rollover Image text box, click Browse and select the image you want displayed when the pointer rolls over the original image, or enter the image file’s path in the text box.
- 6 If you want the images preloaded in the browser’s cache so no delay occurs when the user rolls the pointer over the image, select the Preload Images option.
- 7 In the When Clicked Go to URL text box, click Browse and select the file, or type the path to the file that you want to open when a user clicks the rollover image.

Note: If you don’t set a link for the image, Dreamweaver inserts a null link (#) in the HTML source code to which the rollover behavior is attached. If you remove the null link, the rollover image will no longer work.

- 8 Click OK to close the Insert Rollover Image dialog box.
- 9 Choose File > Preview in Browser or press F12.
- 10 In the browser, move the pointer over the original image.

The display should switch to the rollover image.

Using an external image editor

While in Dreamweaver, you can open a selected image in an external image editor; when you return to Dreamweaver after saving the edited image file, any changes you made to the image are visible in the Document window.

You can use Macromedia Fireworks as your external image editor. Fireworks 3 and later use Design Notes to track where the original PNG file is stored on your local hard disk; when you open the image in Fireworks, it lets you edit the original PNG.

If you set any other image-editing application as the external image editor and launch the editor from within Dreamweaver, the application launches and opens the selected image. Use the image editor to modify the image, save your changes, and then view the updated image in Dreamweaver.

If the image file was generated from a PNG file, you can manually open the original file, make changes, and save the changed image. Dreamweaver still updates the image in the Document window when you return.

If you don't see an updated image after returning to the Dreamweaver window, select the image and then click the Refresh button in the Property inspector.

Launching an external image editor

You choose Edit > Preferences > File Types/Editor or Dreamweaver > Preferences > File Types/Editor (Mac OS X) to set an external image editor for the image file types you specify. For more information about choosing an image editor, see Setting external image editor preferences in Dreamweaver Help.

To launch the external image editor, do one of the following:

- Double-click the image you want to edit.
- Right-click (Windows) or Control-click (Macintosh) the image you want to edit, then choose Edit With >Browse and select an editor.
- Select the image you want to edit, and click Edit in the Property inspector.
- Double-click the image file in the Site panel to launch the primary image editor. If you haven't specified an image editor, Dreamweaver launches the default editor for the file type.

Note: When you open an image from the Site panel, the Fireworks integration features are not in effect; Fireworks does not open the original PNG file. To use the Fireworks integration features, open images from within the Document window.

Applying behaviors to images

You can apply any available behavior to an image or image hotspot. When you apply a behavior to a hotspot, Dreamweaver inserts the HTML source code into the `area` tag. Three behaviors apply specifically to images: Preload Images, Swap Image, and Swap Image Restore.

Preload Images loads images that do not appear on the page right away (such as those that will be swapped in with timelines, behaviors, layers, or JavaScript) into the browser cache. This prevents delays caused by downloading when it is time for the images to appear. See "Preload Images" on page 364.

Swap Image swaps one image for another by changing the `SRC` attribute of the `img` tag. Use this action to create button rollovers and other image effects (including swapping more than one image at a time). See "Swap Image" on page 373.

Swap Image Restore restores the last set of swapped images to their previous source files. This action is automatically added whenever you attach the Swap Image action to an object by default; you should never need to select it manually. See "Swap Image Restore" on page 374.

You can also use behaviors to create more sophisticated navigation systems, such as a navigation bar or a jump menu. See "Creating navigation bars" on page 414 and "Creating jump menus" on page 412.

CHAPTER 21

Dreamweaver Integration with Other Applications

Macromedia Fireworks MX and Macromedia Flash MX are powerful web development tools designed to create graphics and SWF movie which are viewable in web pages. Macromedia Dreamweaver MX is tightly integrated with these tools to enable you to simplify your web design workflow.

To set up smooth integration between Dreamweaver and Flash or Fireworks, make sure to enable Design Notes when you define your Dreamweaver sites. By default, this option is preselected in the Site definition setup. For information about enabling Design Notes, see “Enabling and disabling Design Notes” on page 98.

When you export files from Fireworks or Flash directly to a Dreamweaver defined site, Design Notes which contain references to the PNG or Flash authoring file (FLA) are automatically exported to the site along with the web-ready file (GIF, JPEG, or SWF).

You can easily insert Fireworks images or tables and Flash movies in a Dreamweaver document. You can also take advantage of the integration features between Dreamweaver and Fireworks or Flash to make changes to an image or movie after you’ve inserted in a Dreamweaver document.

While working in Dreamweaver, you can also initiate the graphic production process by inserting, and then updating an image placeholder graphic. For information about image placeholders, see “Inserting an image placeholder” on page 299. After you insert an image placeholder in Dreamweaver, you can launch Fireworks MX to create a new graphic. In Fireworks, you can design the graphic, add hotspots or behaviors or whatever elements you want. You can then save the graphic as a PNG and export it as a web-ready graphic file or files, such as a GIF, or JPEG, or in the case of a sliced table export it as HTML and images.

When you return to Dreamweaver the replacement image or Fireworks table is updated in the document.

This chapter contains the following sections:

“Specifying launch-and-edit preferences for Fireworks source files” on page 307

“Working with Dreamweaver and Fireworks” on page 306

“Editing a Fireworks image or table” on page 309

“Creating a Web photo album” on page 313

“Working with Dreamweaver and Flash” on page 314

“Editing a Flash movie from Dreamweaver” on page 315

About Fireworks and Flash integration

Roundtrip editing and Design Notes enable Dreamweaver to integrate operations with Fireworks and Flash. Roundtrip editing ensures that code updates transfer correctly between Dreamweaver and these other applications, for example to preserve rollover behaviors, or links to other files, while Design Notes allow Dreamweaver to locate the appropriate source document for an exported image or movie file. For information about using Design Notes in Dreamweaver, see “Using Design Notes in Fireworks and Flash with Dreamweaver” on page 101.

In addition to location information, Design Notes contain other pertinent information about exported files. For example, when you export a Fireworks table, Fireworks writes a Design Note for each exported image file in the table. If the exported file contains hotspots or rollovers, the JavaScript for the hotspots or rollovers is contained in the HTML document that Fireworks exports.

For best results, when developing graphics and movies for web publication, save the Fireworks and Flash source and web-ready files in a Dreamweaver defined site. This ensures that any user sharing the site will be able to locate the source document when editing a Fireworks image or table or when editing a SWF movie while working in Dreamweaver.

The key to developing a smoothly integrated workflow with these applications is to first set up the work environment. For information about setting up the Dreamweaver and Fireworks work environment, see “Working with Dreamweaver and Fireworks” on page 306. For information about setting up the Dreamweaver and Flash work environment, see “Working with Dreamweaver and Flash” on page 314.

Working with Dreamweaver and Fireworks

Dreamweaver and Fireworks recognize and share many of the same file edits, including changes to links, image maps, table slices, and more. Together, the two applications provide a streamlined workflow for editing, optimizing, and placing web graphics files in HTML pages.

To set up an integrated work environment, you must complete some preliminary tasks, such as defining a local site in Dreamweaver and making sure Design Notes are enabled for the site. Design Notes are automatically enabled unless you alter your default settings.

You must also set Fireworks as the primary external image editor for Dreamweaver to easily launch Fireworks for editing. For information about setting Fireworks as an external editor, see “Using an external image editor” on page 302.

To ensure smooth integration for launch-and-edit, when you are ready to export your Fireworks HTML and graphic files export them to the Dreamweaver site folder. When you export a GIF or JPEG graphic from Fireworks to a Dreamweaver site folder, Fireworks creates a folder named `_notes` in the same folder. This folder contains the Design Notes, small files with the Macromedia Note file extension (`.mno`).

Design Notes contain information about the graphic files that Fireworks exports. When you launch and edit a Fireworks image or table in Dreamweaver, Dreamweaver uses this information to locate the source PNG. For information about specifying whether Dreamweaver automatically launches the PNG file when available, see “Specifying launch-and-edit preferences for Fireworks source files” on page 307.

When you select an image in a Dreamweaver document that was exported from Fireworks and has a corresponding `.mno` file, the Property inspector displays the Fireworks icon and the file’s source path.

Setting Fireworks as the external image editor for Dreamweaver allows you to go between Dreamweaver and Fireworks effortlessly when you need to edit an image. In Dreamweaver preferences, set Fireworks as the primary editor for your graphic file types—GIF, PNG, and JPEG files. For information about setting an image editor, see “Using an external image editor” on page 302 or Setting external image editor preferences in Dreamweaver Help.

Specifying launch-and-edit preferences for Fireworks source files

The Fireworks launch-and-edit preferences let you specify how to handle source PNG files when launching Fireworks files from another application, such as Dreamweaver.

Dreamweaver recognizes the Fireworks launch-and-edit preferences only in certain cases where you launch and optimize a Fireworks image. Specifically, you must be launching and optimizing an image that is not part of a Fireworks table and does not contain a correct Design Notes path to a source PNG file. In all other cases, including all launch-and-edit cases of Fireworks images, Dreamweaver automatically launches the source PNG file, prompting you to locate the source file if it cannot be found.

To specify launch-and-edit preferences for Fireworks:

- 1 In Fireworks, choose Edit > Preferences or Fireworks > Preferences (Mac OSX) and then click the Launch and Edit tab (Windows) or choose Launch and Edit from the pop-up menu (Macintosh).
- 2 Specify the preference options to use when editing or optimizing Fireworks images placed in an external application:

Always Use Source PNG automatically launches the Fireworks PNG file that is defined in the Design Note as the source for the placed image. Updates are made to both the source PNG and its corresponding placed image.

Never Use Source PNG automatically launches the placed Fireworks image, whether or not a source PNG file exists. Updates are made to the placed image only.

Ask When Launching lets you specify each time whether or not to launch the source PNG file. When you edit or optimize a placed image, Fireworks displays a message prompting you to make a launch-and-edit decision. You can also specify global launch-and-edit preferences from this message.

Inserting a Fireworks image in a Dreamweaver document

Fireworks graphics can be placed in a Dreamweaver document a number of ways. You can place a Fireworks exported graphic directly in a Dreamweaver document using the Insert Image command, or you can create a new Fireworks graphic from a Dreamweaver image placeholder.

To insert a Fireworks image into a Dreamweaver document:

- 1 In the Dreamweaver document, place the insertion point where you want the image to appear, then do one of the following:
 - Choose Insert > Image.
 - In the Common category of the Insert bar, click the Image button or drag it to the document.
- 2 Navigate to the desired Fireworks exported file, and click OK (Windows) or Open (Macintosh).

Note: If the Fireworks file is not in the current Dreamweaver site, a message appears asking whether you want to copy the file to the root folder. Click Yes.

Updating a Dreamweaver image placeholder in Fireworks

You can create a placeholder image in a Dreamweaver document and then launch Fireworks to design a graphic image or Fireworks table to replace it. For information about inserting an image placeholder, see “Inserting an image placeholder” on page 299.

To create a new image from an image placeholder, you must have both Dreamweaver MX and Fireworks MX installed on your system. Fireworks recognizes the following image placeholder settings you may have set while working with the image placeholder in Dreamweaver: image size which correlates to the Fireworks canvas size, image ID which Fireworks uses as the default document name for the source file and export file you create, text alignment, and behaviors that are recognized by Fireworks (such as swap image, pop-up menu, navigation bar, and set text). Fireworks also recognizes links you attached to the image placeholder while working in Dreamweaver.

Note: Although links you’ve added to an image placeholder can’t be seen in Fireworks, they are preserved. If you draw a hotspot and add a link in Fireworks MX, it will not delete the link you added to the image placeholder in Dreamweaver. However, if you draw out a slice in Fireworks in the new image, it will delete the link in the Dreamweaver document when you replace the image placeholder.

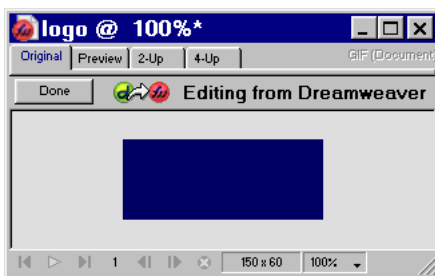
These image placeholder settings are disabled in the image placeholder Property inspector since they are not recognized by Fireworks: image alignment, color, Vspace and Hspace, and maps.

In Fireworks MX, create the new image, then click Done. Fireworks prompts you to save the file as a PNG file (source document), and to export the file in a web-ready format such as a GIF, JPEG, or in the case of sliced images as HTML and images. The new image or Fireworks table automatically replaces the image placeholder in the Dreamweaver document.

To edit a Dreamweaver placeholder image in Fireworks:

- 1 Make sure you’ve already set Fireworks as the image editor for .png files. For information, see Setting external image editor preferences in Dreamweaver Help.
- 2 In the Document window, click the image placeholder to select it.
- 3 Do one of the following to launch Fireworks for editing:
 - In the Property inspector, click Create.
 - Press Control (Windows) or Command (Macintosh) then double-click the image placeholder.
 - Right-click the image placeholder, then choose Create Image in Fireworks.

Fireworks launches, in Editing from Dreamweaver mode



- 4 Use Fireworks options to design the image.

- 5 When you are finished, click Done.

The Save As dialog box appears. Fireworks prompts you to save the PNG file.

- 6 In the Save In field, choose the folder you defined as your Dreamweaver local site folder.

If you named the image placeholder when you inserted it in the Dreamweaver document, Fireworks populates the File Name field with that name. You can change the name if you want to.

- 7 Click Save, to save the PNG file.

The Export dialog box appears. Use this dialog box to export the image as a GIF.

- 8 In the dialog box, for Save In choose the Dreamweaver local site folder.

- 9 The Name text field automatically updates to the same name you used for the PNG file. Enter text to change the name if you want to.

- 10 For Save As Type, select the type of file or files you want to export, for example Images only or HTML and Images.

- 11 Click Save, to save the exported file.

The file is saved, and focus returns Dreamweaver. In the Dreamweaver document, the exported file or Fireworks table replaces the image placeholder.

Editing a Fireworks image or table

You can launch Fireworks to edit images inserted in a Dreamweaver document. When you launch and edit an image or an image slice that is part of a Fireworks table, Dreamweaver launches Fireworks and the PNG file from which the image or table was exported.

When the image is part of a Fireworks table, you can launch the entire Fireworks table for edits, as long as the `<!--fw table-->` comment exists in the HTML code. If the source PNG was exported from Fireworks to a Dreamweaver site using the Dreamweaver Style HTML and images setting, the Fireworks table comment is automatically inserted in the HTML code.

To launch and edit a Fireworks image placed in Dreamweaver:

- 1 In Dreamweaver, choose Window > Properties to open the Property inspector if it isn't already open.

- 2 Click the image or image slice to select it.

The Property inspector identifies the selection as a Fireworks image or table based on the selected item, and displays the name of the PNG source file.

- 3 To launch Fireworks for editing, do one of the following:

- In the Property inspector, click Edit.
- Hold down Control (Windows) or Command (Macintosh), and double-click the selected image.
- Right-click (Windows) or Control-click (Macintosh) the selected image, and choose Edit With Fireworks from the context menu.

Fireworks launches and opens the associated PNG for editing.

Note: If Fireworks cannot locate the source file, you are prompted to locate the PNG source file. When you work with the Fireworks source file, changes are saved to both the source file and the exported file; otherwise, only the exported file is updated.

- 4 In Fireworks, edit the source PNG.
- 5 When you are finished making edits, click Done.

Fireworks saves the changes in the PNG file, exports the updated image (or HTML and images), and returns focus to Dreamweaver. In Dreamweaver, the updated image or table appears.

Opening a Fireworks Pop-up Menu in Dreamweaver

Fireworks supports image-based pop-up menus and HTML-based pop-up menus. Dreamweaver only supports HTML-based pop-up menus. In Dreamweaver, you can open a Fireworks pop-up menu and make edits to all menu item properties except for the background images of image-based pop-up menus.

The Show Pop-Up Menu behavior, in Dreamweaver, allows you to edit or update the contents of a Fireworks HTML-based pop-up menu. You can add, delete, or change menu items, rearrange them, and set where a menu is positioned in a page. For information about setting or modifying pop-up menu options in Dreamweaver, see “Show Pop-Up Menu” on page 370.

If the pop-up menu you want to edit is an image-based pop-up menu and you want to preserve the image-based cell backgrounds, you should use edit the pop-up menu in Fireworks rather than in Dreamweaver.

To edit the background images in an image-based pop-up menu, select the image you want to update, then click Edit. For information about editing a Fireworks image see “Editing a Fireworks image or table” on page 309.

To open the Fireworks pop-up menu:

- 1 In the Dreamweaver document, select the hotspot or image that triggers the pop-up menu.
- 2 Open the Behaviors panel (Shift+F3), if it isn't already open, then in the Actions list, double-click Show Pop-Up Menu.

The Show Pop-Up Menu dialog box appears.

- 3 Make the changes you want to make to the pop-up menu.
- 4 When you finish modifying the pop-up menu, click OK.

Launching Fireworks to optimize an image

You can launch Fireworks from Dreamweaver to make quick export changes, such as resizing an image or changing the file type, to placed Fireworks images and animations. Fireworks lets you make changes to optimization settings, animation settings, and the size and area of the exported image.

To change optimization settings for a Fireworks image placed in Dreamweaver:

- 1 In Dreamweaver, select the desired image and choose Commands > Optimize Image in Fireworks.
- 2 If prompted, specify whether to launch a source Fireworks file for the placed image.
- 3 In Fireworks, make the desired edits in the Optimization dialog box:
 - To edit optimization settings, click the Options tab. For more information, see *Using Fireworks*.
 - To edit the size and area of the exported image, click the File tab.

- 4 When you are finished editing the image, click Update.

Clicking Update exports the image using the new optimization settings, updates the GIF or JPEG placed in Dreamweaver, and saves the PNG source file if a source file was selected.

If you changed the format of the image, the Dreamweaver link checker prompts you to update references to the image. For example, if you changed the format of an image called `my_image` from GIF to JPEG, clicking OK at this prompt changes all references to `my_image.gif` in your site to `my_image.jpg`.

Inserting Fireworks HTML code in a Dreamweaver document

The Export command in Fireworks lets you export and save optimized images and HTML files to a location inside the desired Dreamweaver site folder. You can then insert the file in Dreamweaver.

For information about exporting Fireworks files as HTML, see *Using Fireworks*.

Dreamweaver lets you insert Fireworks-generated HTML code, complete with associated images, slices, and JavaScript, into a document. This insertion feature makes it easy for you to create design elements in Fireworks then incorporate them into an existing Dreamweaver document.

To insert Fireworks HTML into a Dreamweaver document:

- 1 In Dreamweaver, place the insertion point in the document where you want the Fireworks HTML code to begin.
- 2 Do one of the following:
 - Choose Insert > Interactive Images > Fireworks HTML.
 - In the Common category of the Insert bar, click the Insert Fireworks HTML button.
- 3 In the dialog box that appears, click Browse to choose the desired Fireworks HTML file.
- 4 Select the Delete File After Insertion option to move the original Fireworks HTML file to the Recycle Bin (Windows) or Trash (Macintosh) when the operation is complete.

Use this option if you no longer need the Fireworks HTML file after inserting it. This option does not affect the source PNG file associated with the HTML file.

Note: If the HTML file is on a network drive, it is permanently deleted—not moved to the Recycle Bin or Trash.
- 5 Click OK to insert the HTML code, along with its associated images, slices, and JavaScript, into the Dreamweaver document.

Pasting Fireworks HTML into Dreamweaver

A fast way to place Fireworks-generated images and tables in Dreamweaver is to copy and paste Fireworks HTML code directly into a Dreamweaver document.

To copy and paste Fireworks HTML into Dreamweaver:

- 1 In Fireworks, choose Edit > Copy HTML Code.
- 2 Follow the wizard as it guides you through the settings for exporting your HTML and images. When prompted, specify your Dreamweaver site folder as the destination for the exported images.

The wizard exports the images to the specified destination and copies the HTML code to the Clipboard.

- 3 In Dreamweaver, place the insertion point in the document where you want to paste the HTML code, and choose Edit > Paste.

All HTML and JavaScript code associated with the Fireworks files you exported is copied into the Dreamweaver document, and all links to images are updated.

To export and paste Fireworks HTML into Dreamweaver:

- 1 In Fireworks, choose File > Export.
- 2 In the Export dialog box, specify your Dreamweaver site folder as the destination for the exported images.
- 3 In the Save As pop-up menu, choose HTML and Images.
- 4 In the HTML pop-up menu, choose Copy to Clipboard, then click Save.
- 5 In Dreamweaver, place the insertion point in the document where you want to paste the exported HTML code, and choose Edit > Paste.

All HTML and JavaScript code associated with the Fireworks files you exported is copied into the Dreamweaver document, and all links to images are updated.

Updating Fireworks HTML placed in Dreamweaver

In Fireworks, the File > Update HTML command provides an alternative to the launch-and-edit technique for updating Fireworks files placed in Dreamweaver. With Update HTML, you can edit a source PNG image in Fireworks and then automatically update any exported HTML code and image files placed in a Dreamweaver document. This command lets you update Dreamweaver files even when Dreamweaver is not running.

To update Fireworks HTML placed in Dreamweaver:

- 1 In Fireworks, open the source PNG and make the desired edits to it.
- 2 Choose File > Save
- 3 In Fireworks, choose File > Update HTML.
- 4 Navigate to the Dreamweaver file containing the HTML you want to update, and click Open.
- 5 Navigate to the folder destination where you want to place the updated image files, and click Select (Windows) or Choose (Macintosh).

Fireworks updates the HTML and JavaScript code in the Dreamweaver document. Fireworks also exports updated images associated with the HTML and places the images in the specified destination folder.

If Fireworks cannot find matching HTML code to update, it gives you the option of inserting new HTML code into the Dreamweaver document. Fireworks places the JavaScript section of the new code at the beginning of the document and places the HTML table or link to the image at the end.

Creating a Web photo album

The Create Web Photo Album command in Dreamweaver lets you automatically generate a web site that showcases an album of images from a given folder. This command uses JavaScript to call Fireworks, which creates a thumbnail and a larger-sized image for each of the images in the folder. Dreamweaver then creates a web page containing all the thumbnails, as well as links to the larger images. To use the Create Web Photo Album, you must have both Dreamweaver and Fireworks 4 or later installed on your system.

Before you begin, place all of the images for your photo album in a single folder. (The folder is not required to be in a site.) In addition, make sure that the image filenames end in extensions recognized by the Create Web Photo Album command (.gif, .jpg, .jpeg, .png, .psd, .tif, or .tiff). Images with unrecognized file extensions are not included in the photo album.

To create a Web photo album:

- 1 In Dreamweaver, choose **Commands > Create Web Photo Album**.
- 2 In the Photo Album Title text field, enter a title. The title will be displayed in a gray rectangle at the top of the page containing the thumbnails.

If desired, you can enter up to two lines of additional text to appear directly beneath the title, in the Subheading Info and Other Info text fields.

- 3 Choose the folder containing source images by clicking the Browse button next to the Source Images Folder text field. Then choose (or create) a destination folder in which to place all the exported images and HTML files by clicking the Browse button next to the Destination Folder text field.

The destination folder should not already contain a photo album—if it does, and if any new images have the same names as previously used images, you might overwrite existing thumbnail and image files.

- 4 Specify display options for the thumbnail images:
 - Choose a size for the thumbnail images from the Thumbnail Size pop-up menu. Images are scaled proportionally to create thumbnails that fit within a square that has the indicated pixel dimensions.
 - To display the filename of each original image below the corresponding thumbnail, select Show Filenames.
 - Enter the number of columns for the table that displays the thumbnails.
- 5 Choose a format for the thumbnail images from the Thumbnail Format pop-up menu:
 - GIF WebSnap 128** creates GIF thumbnails that use a web adaptive palette of up to 128 colors.
 - GIF WebSnap 256** creates GIF thumbnails that use a web adaptive palette of up to 256 colors.
 - JPEG–Better Quality** creates JPEG thumbnails with relatively higher quality and larger file sizes.
 - JPEG–Smaller File** creates JPEG thumbnails with relatively lower quality and smaller file sizes.
- 6 Choose a format for the large-size images from the Photo Format pop-up menu. A large-size image of the specified format is created for each of your original images. You may specify a format for the large-size images that differs from the format you specified for the thumbnails.

Note: The Create Web Photo Album command does not let you use your original image files as the large-size images, because original image formats other than GIF and JPEG might not display properly on all browsers. Note that if your original images are JPEG files, the large-size images generated may have larger file sizes or lower quality than the original files.

- 7 Choose a Scale percentage for the large-size images.

Setting Scale to 100% creates large-size images the same size as the originals. Note that the scale percentage is applied to all of the images; if your original images aren't all the same size, scaling them by the same percentage may not produce the desired results.

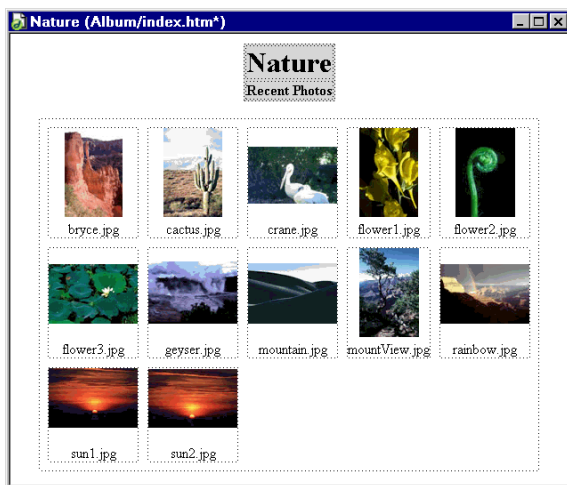
- 8 Select Create Navigation Page for Each Photo to create an individual web page for each source image, containing navigation links labeled Back, Home, and Next.

If you select this option, the thumbnails link to the navigation pages. If you don't select this option, the thumbnails link directly to the large-size images.

- 9 Click OK to create the HTML and image files for the web photo album.

Fireworks launches (if it's not already running) and creates the thumbnails and large-size images. This may take several minutes if you've included a large number of image files. When the processing is complete, Dreamweaver becomes active again, and creates the page containing the thumbnails.

- 10 When a dialog appears that says "Album Created," click OK. You may have to wait a few seconds for your photo album page to appear. The thumbnails are shown in alphabetical order by filename.



Note: Clicking the Cancel button in the Dreamweaver dialog box after processing has begun does not stop the process of creating the photo album; it merely prevents Dreamweaver from displaying the main photo album page.

Working with Dreamweaver and Flash

You can easily insert a Flash movie (SWF file) in a Dreamweaver document. You can then use the Property inspector to set playback and display options for the SWF file in the web page. For information about inserting a Flash movie in Dreamweaver see "Inserting Flash movies" on page 326.

The link checker in Dreamweaver lets you easily edit links in files you insert in a Dreamweaver document, including Flash movies. You can update the link in the SWF movie, and then update the change in the Flash authoring document. For information, see "Updating links in a SWF file" on page 316.

If you have Macromedia Flash MX and Dreamweaver MX you can also update a movie that has been placed in a Dreamweaver document. When both applications are installed on your computer, and a SWF movie is selected in the Dreamweaver document, the Property inspector displays an active Edit button. If you do not have Flash MX, the Edit button is disabled.

When Edit is clicked, Dreamweaver launches Flash, and Flash attempts to locate the Flash authoring file (.FLA) for the selected SWF file. Information about the original source file is automatically stored in a Design Note for the SWF file when you export it to a Dreamweaver site (if Design Notes are enabled for the Dreamweaver site). If Flash cannot locate the Flash authoring file, you are prompted to locate it. You cannot update a SWF file directly, you make changes to the source file, then export it as a SWF movie file.

After successfully launching a source document to edit, you can work in Flash to make changes to a movie. Click Done when you are finished making changes. Flash updates the Flash authoring document, re-exports the movie file, closes, and then returns the focus to the Dreamweaver document. You can view the updated SWF in the document by clicking Play in the Property inspector, or press F12 to preview your page in a browser window.

Related Topics

“Editing a Flash movie from Dreamweaver” on page 315

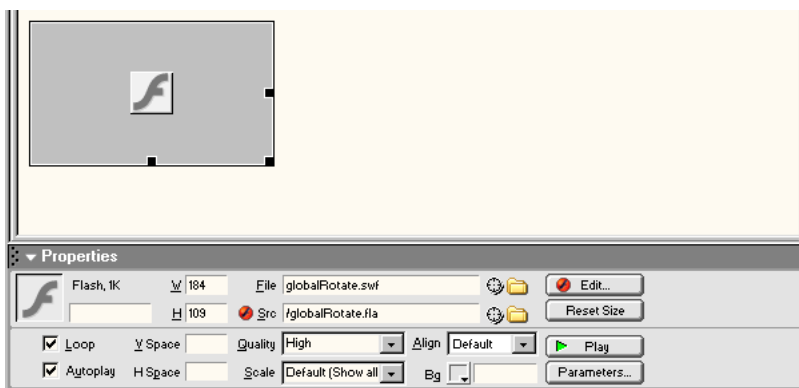
“Updating links in a SWF file” on page 316

Editing a Flash movie from Dreamweaver

You cannot directly edit a SWF file, if you want to make changes to an exported SWF, select the Flash movie placeholder in the Dreamweaver document, and in the Property inspector click Edit. Edit launches Flash and if the path to the source document (FLA) is available, also launches the FLA file. When the edits are complete, Flash saves the changes in the FLA source document and re-exports the SWF movie file.

To launch and edit a Flash movie inserted from Dreamweaver:

- 1 In Dreamweaver, choose Window > Properties to open the Property inspector, if it isn't already open.
- 2 In the Dreamweaver document, do one of the following:
 - Click the Flash movie placeholder to select it; then in the Property inspector click Edit.

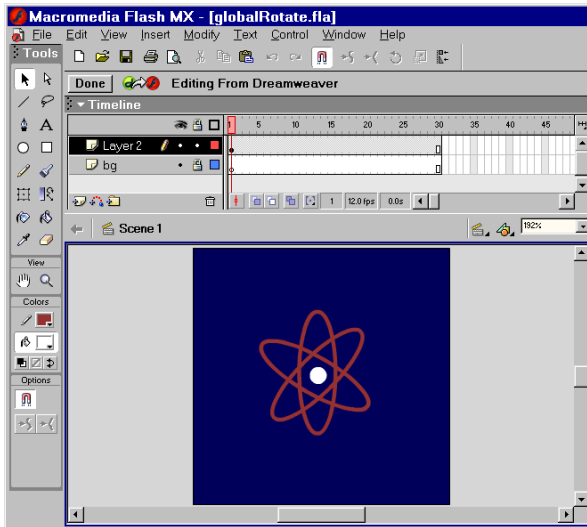


- Hold down Control (Windows) or Command (Macintosh), and double-click the movie placeholder for the movie you want to edit.
- Right-click (Windows) or Control-click (Macintosh) the desired movie, and choose Edit With Flash from the context menu.

Dreamweaver launches Flash, and opens the FLA file for you to edit or prompts you to open it if it cannot locate the FLA file.

Note: If the FLA file or SWF file is locked, Dreamweaver prompts you to check the file out, cancel the request, or view the file.

- 3 In Flash, edit the movie. The Document window indicates that you are modifying a movie from Dreamweaver.



- 4 When you are finished making edits, click Done.

Clicking Done saves changes to the source FLA file, and updates the SWF file.

Updating links in a SWF file

You can update a link in a SWF file in the Dreamweaver Site Map view. For information about Site Map view, see “Using the site map” on page 81. Before you can create a site map, you must set a home page for your site. In Site Map view, you need to display dependent files in order to update a link in a SWF file. By default the site map does not show dependent files, for information about displaying dependent files, see “Showing and hiding site map files” on page 85.

Select the link you want to update, then select one of the change links options depending on what you want to do—change an individual link or change the link sitewide.

Any links updated by Dreamweaver in the SWF file are conveyed to the FLA source document when a launch-and-edit is performed. Dreamweaver automatically logs any link changes to the SWF file in the Design Notes, and when Flash passes the changes to the FLA file it removes them from the Design Notes.

To update a URL link in a SWF file:

- 1** Set up a home page for the site, if you haven't already done so. You need to set up a home page in order to build a site map.
- 2** Open Site Map view.
- 3** To show dependent files, choose View > Show Dependent Files (Windows) or Site > Site Map View > Show Dependent Files (Macintosh).

The link appears beneath the SWF file.

- 4** Change the link by doing one of the following:
 - To change the link in the selected SWF, right-click the link, then choose Change Link, then in the dialog box that appears in the URL field, type the new URL path.
 - To update all instances of the link, choose Site > Change Link Sitewide, then in the dialog box that appears, in the Change All Links To field, browse to or type the path of the link you are changing and in the Into Links To field, browse to or type the path of the new URL.
- 5** Click OK.

CHAPTER 22

Inserting Media

Macromedia Dreamweaver lets you add sound and movies to your website quickly and easily. You can incorporate and edit multimedia files and objects, such as Macromedia Flash and Shockwave movies, Java applets, QuickTime, Active X, and audio files. You can attach Design Notes to these objects, which let you communicate with your team. You can also insert Flash button and text objects from within Dreamweaver itself.

This chapter contains the following sections:

- “Inserting and playing media objects” on page 320
- “Launching an external editor for media files” on page 320
- “Using Design Notes with media objects” on page 321
- “About Flash content” on page 322
- “Inserting a Flash button object” on page 322
- “Inserting a Flash text object” on page 324
- “Inserting Flash movies” on page 326
- “Inserting Shockwave movies” on page 326
- “Adding video” on page 327
- “Adding sound to a page” on page 327
- “Inserting Netscape Navigator plug-in content” on page 329
- “Inserting an ActiveX control” on page 330
- “Inserting a Java applet” on page 330
- “Using behaviors to control media” on page 331

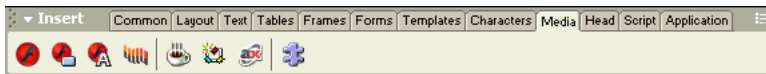
Inserting and playing media objects

You can insert a Flash movie or object, QuickTime or Shockwave movie, Java applet, ActiveX control, or other audio or video objects in a Dreamweaver document.

You use the Media category of the Insert bar or the Insert menu to select the type of object you want to insert in a document. Flash movie and objects, Shockwave, Applets, and ActiveX have defined buttons. Use the Netscape Navigator plug-in button to insert other media files. For more information, see “Inserting Netscape Navigator plug-in content” on page 329.

To insert a media object in a page:

- 1 Place the insertion point in the Document window where you want to insert the object, then do one of the following:
 - In the Insert bar, select Media, then click the button for the type of object you want to insert, or drag it to the Document window.



- Choose the appropriate object from the Insert > Media or Insert > Interactive Images submenu.

In most cases, a dialog box appears letting you select a source file and specify certain parameters for the media object.

Tip: To prevent such dialog boxes from appearing, choose Edit > Preferences > General or Dreamweaver > Preferences > General (Mac OS X) and deselect the Show Dialog When Inserting Objects option. To override whatever preference is set for showing dialog boxes, hold down the Control (Windows) or Option (Macintosh) key while inserting the object. (For example, to insert a placeholder for a Shockwave movie without specifying the file, hold down Control or Option and click the Shockwave button.)

Dreamweaver automatically inserts the necessary HTML source code to make the object or placeholder appear on the page. To specify a source file, set dimensions and other parameters and attributes, use the Property inspector for each object.

Launching an external editor for media files

You can double-click most files in the Site panel to edit them directly. If the file is an HTML file, it opens in Dreamweaver. If it's another kind of file, such as an image file, it opens in an appropriate external editor, such as Macromedia Fireworks.

Each file type that Dreamweaver doesn't handle directly can be associated with one or more external editors found on your system. The editor that launches when you double-click the file in the Site panel is called the primary editor. You can set which editor is associated with a file type in File Types/Editors preferences.

If more than one editor is associated with the file type, you can launch a secondary editor for a particular file. Right-click (Windows) or Control-click (Macintosh) the filename in the Site panel, and choose an editor from the Open With submenu of the context menu.

To explicitly specify which external editors should be launched for a given file type, choose Edit > Preferences and select File Types/Editors from the Category list. Filename extensions, such as .gif, .wav, and .mpg, are listed on the left under Extensions. Associated editors for a selected extension are listed on the right under Editors.

You can also browse to find an external editor to edit your file. Right-click (Windows) or Control-click (Macintosh) the file in the Design view of the Document window, and choose Edit With > Browse or select the file and choose > Edit > Edit with External Editor.

To add a file type to the list of extensions in File Types/Editors preferences:

- 1 Click the plus (+) button above the Extensions list.
- 2 Enter a file type extension (including the period at the beginning of the extension), or several related extensions separated by spaces.

For example, you might enter `.css`, `.png` `.jpg`.

To add an editor for a given file type:

- 1 Select the file type extension in the Extensions list.
- 2 Click the plus (+) button above the Editors list.
- 3 In the dialog box that appears, choose an application to add to the Editors list.

For example, choose the application icon for Excel to add that application to your Editors list.

To remove a file type from the list:

- 1 Select the file type in the Extensions list.

Note: You can't undo after removing a file type, so be sure that you want to remove it.

- 2 Click the minus (-) button above the Extensions list.

To make an editor the primary editor for a file type:

- 1 Select the file type.
- 2 Select the editor (or add it if it isn't on the list).
- 3 Click Make Primary.

To dissociate one editor from a file type:

- 1 Select the file type in the Extensions list.
- 2 Select the editor in the Editors list.
- 3 Click the minus (-) button above the Editors list.

Using Design Notes with media objects

As with other objects in Dreamweaver, you can add Design Notes to a media object. For more information on working with Design Notes, see "Using Design Notes" on page 97.

To add Design Notes to a media object:

- 1 Right-click (Windows) or Control-click (Macintosh) the object in the Document window.

Note: You must define your site before adding Design Notes to any object (see "Enabling and disabling Design Notes" on page 98).

- 2 Choose Design Notes from the context menu.
- 3 Enter the information you want in the Design Note.

Tip: You can also add a Design Note to a media object from the Site panel by selecting the file, revealing the context menu, and choosing Design Notes from the context menu.

About Flash content

Macromedia Flash technology is the premier solution for the delivery of vector-based graphics and animations. The Flash Player is available as both a Netscape Navigator plug-in and an ActiveX control for Microsoft Internet Explorer on the PC, and it is incorporated in the latest versions of Netscape Navigator, Microsoft Internet Explorer, and America Online.

Dreamweaver comes with Flash objects you can use whether you have Flash or not. You can use these objects to create Flash buttons and Flash text you can insert in a Dreamweaver document. If you have Flash, see “Working with Dreamweaver and Flash” on page 314 for information about using these applications in an integrated manner.

Before you use the Flash commands available in Dreamweaver, you should know about the three different Flash file types:

The Flash file (.fla) is the source file for any project and is created in the Flash program. This type of file can only be opened in Flash (not in Dreamweaver or in browsers). You can open the Flash file in Flash, then export it as an SWF or SWT file to use in browsers.

The Flash movie file (.swf) is a compressed version of the Flash (.fla) file, optimized for viewing on the web. This file can be played back in browsers and previewed in Dreamweaver, but cannot be edited in Flash. This is the type of file you create when using the Flash button and text objects. For more information, see “Inserting a Flash button object” on page 322, “Inserting a Flash text object” on page 324, and “Inserting Flash movies” on page 326.

The Flash template (.swt) files enable you to modify and replace information in a Flash movie file. These files are used in the Flash button object, which lets you modify the template with your own text or links, to create a custom SWF to insert in your document. In Dreamweaver, these template files can be found in the Dreamweaver/Configuration/Flash Objects/Flash Buttons and Flash Text folders.

You can download new button templates from the Macromedia Exchange for Dreamweaver website and place them in your Flash Button folder. For more information on creating new button templates, see the article on that topic found on the Macromedia website at http://www.macromedia.com/go/flash_buttons.

Inserting a Flash button object

The Flash button object is an updateable button that is based on a Flash template. You can customize a Flash button object, adding text, background color, and links to other files. Flash buttons can be inserted while working in Design view or in Code view.

Note: You must save your document before inserting a Flash button or text object.

You use the Insert Flash Button dialog box to select from a set of designed Flash buttons. You can view an example of the button in the Sample field. Click the sample to see how it functions in the browser. While you are defining the Flash button (for example, changing text or font choices), the Sample field does not automatically update to reflect the changes. These changes will appear when you close the dialog box and view the button in Design view.

To insert a Flash button object:

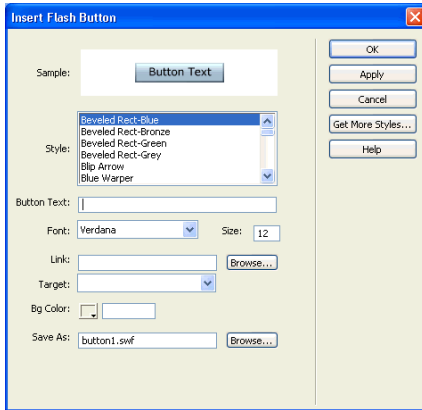
- 1 In the Document window, place the insertion point where you want to insert the Flash button.
- 2 To open the Insert Flash Object dialog box, do one of the following:



- In the Insert bar, select Media and then click the Flash Button icon.
- In the Insert bar, select Media and then drag the Flash Button icon to the Document window, or to the Code view window if you are working in the code.

- Choose Insert > Interactive Images > Flash Button.

The Insert Flash Button dialog box appears.



- 3 Select the button style you want from the Style list.
- 4 In the Button Text field (optional), type the text you want to appear.
This field only accepts changes if the selected button has a {Button Text} parameter defined. This is shown in the Sample field. The text you type in replaces the {Button Text} when you preview the file.
- 5 In the Font pop-up menu, select the font you want to use.
If the default font for a button is not available on your system, select another font from the pop-up menu. You will not see the font you selected in the Sample field, but you can click Apply to insert the button in the page to see what the text will look like.
- 6 In the Size field, enter a numeric value for the font size.
- 7 In the Link field (optional), enter a document-relative or absolute link for the button.
- 8 In the Target field (optional), specify the location in which the linked document will open. You can select a frame or window option in the pop-up menu. Frame names are listed only if the Flash object is being edited while in a frameset.
- 9 In the Bg Color field (optional), set the background color for the Flash movie. Use the color picker or type in a web hexadecimal value (such as #FFFFFF).
- 10 In the Save As field, enter a filename to save the new SWF file.
You can use the default filename (for example, button1.swf), or type in a new name. If the file contains a document-relative link, you must save the file to the same directory as the current HTML document to maintain document-relative links.
- 11 Click Get More Styles to go to the Macromedia Exchange site and download more button styles.
For more information, see “Adding extensions to Dreamweaver” on page 53.
- 12 Click Apply or OK to insert the Flash button in the Document window.

Tip: Select Apply to see changes in Design view, while keeping the dialog box open—this way you can continue making changes to the button.

Modifying a Flash button object

You can modify the properties and content of a Flash button object.

To modify a Flash button object:

- 1 In the Document window, click the Flash button object to select it.
- 2 Open the Property inspector, if it is not already open.

The Property inspector displays the Flash button properties. You can use the Property inspector to modify the button's HTML attributes, such as width, height, and Bg color.

- 3 To make changes to content, display the Insert Flash Button dialog box using one of the following methods:
 - Double-click the Flash button object.
 - Click Edit in the Property inspector.
 - Right-click (Windows) or Control-click (Macintosh) and choose Edit from the context menu.
- 4 In the Insert Flash Button dialog box, make your edits in the fields described in “Inserting a Flash button object” on page 322.

In the Design view, you can resize the object easily by using the resizing handles. You can return the object to its original size by selecting Reset Size on the Property inspector (see “Resizing an image” on page 301).

Playing a Flash button object in the document

You can preview a Flash button in the Dreamweaver Document window.

To see the Flash button object play in the Document window:

- 1 While in Design view, in the document, select the Flash button object.
- 2 In the Property inspector, click Play.
- 3 Click Stop to end the preview.

Note: You can't edit the Flash button object while it is playing.

It's also a good idea to preview your document in the browser to see exactly how the Flash button looks.

Inserting a Flash text object

The Flash text object lets you create and insert a Flash movie that contains just text. This allows you to create a small, vector-graphic movie with the designer fonts and text of your choice.

To insert a Flash text object:

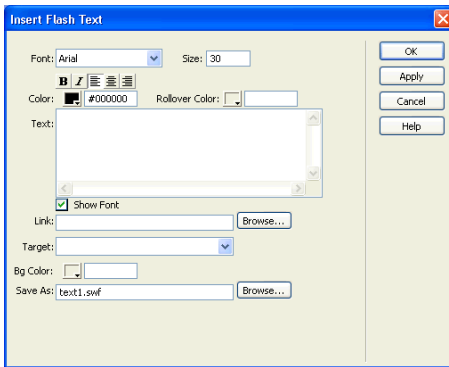
- 1 In the Document window, place the insertion point where you want to insert the Flash text.
- 2 To open the Insert Flash Text dialog box, do one of the following:



- In the Insert bar, select Media and then click the Flash Text icon.
- In the Insert bar, select Media and then drag the Flash Text icon to the Document window (or to the Code view window if you are working in the code).

- Choose Insert > Interactive Images > Flash Text.

The Insert Flash Text dialog box appears.



- 3 Select a font from the Font pop-up menu.

This menu lists all TrueType fonts currently loaded on your system.

- 4 Enter a font size in the Size field; these are point sizes.

- 5 Specify style attributes, like bold or italic, and text alignment by clicking the appropriate style buttons.

- 6 In the Color field, set the color of the text by using the color picker or entering a web hexadecimal value (such as #FFFFFF).

- 7 In the Rollover Color field, set the color that appears when the pointer rolls over the Flash text object. Use the color picker or enter a web hexadecimal value (such as #FFFFFF).

- 8 Enter the text you want in the Text field.

To see the font style displayed in the Text field, select Show Font.

- 9 If you want to associate a link with the Flash text object, enter a document-relative or absolute link in the Link field.

Site-relative links are not accepted because browsers don't recognize them within Flash movies. If you use a document-relative link, make sure to save the SWF file to the same directory as the HTML file. (Browsers vary in their interpretation of document-relative links and saving to the same directory ensures that the links will work correctly.)

- 10 If you entered a link, you can specify a target frame or target window for that link to go to in the Target field.

- 11 In the Bg Color field, choose a background color for the text. Use the color picker or enter a web hexadecimal value (such as #FFFFFF).

- 12 In the Save As field, enter a name for the file.

You can use the default filename (for example, tex1.swf), or type in a new name. If the file contains a document-relative link, you must save the file to the same directory as the current HTML document to maintain document-relative links.

13 Click Apply or OK to insert the Flash text in the Document window.

If you click Apply, the dialog box remains open, and you can preview the text in your document.

To modify or play the Flash text object, use the same procedure as you would for a Flash button (see “Modifying a Flash button object” on page 324).

Inserting Flash movies

When you insert a Flash movie into a document, Dreamweaver uses both the `object` tag (defined by Microsoft Internet Explorer for ActiveX controls) and the `embed` tag (defined by Netscape Navigator) to get the best results in all browsers. When you make changes in the Property inspector for the movie, Dreamweaver maps your entries to the appropriate parameters for both the `object` and `embed` tags.

To insert a Flash movie:

1 In the Design view of the Document window, place the insertion point where you want to insert the movie, then do one of the following:



- In the Insert bar, select Media and then click the Insert Flash icon.
- In the Insert bar, select Media and then drag the Insert Flash icon to the Document window.
- Choose Insert > Media > Flash.

2 In the dialog box that appears, select a Flash movie file (.swf).

A Flash placeholder appears in the Document window (unlike Flash button and text objects).

For information about setting properties for a Flash movie, select the placeholder then click the Help button in the Property inspector.

To preview a Flash movie in the Document window:

1 In the Document window, click the Flash placeholder to select the Flash movie you want to preview.

2 In the Property inspector, click the Play button. Click Stop to end the preview. You can also preview the Flash movie in a browser by pressing F12.

Tip: To preview all Flash content in a page, press Control+Alt+Shift+P (Windows) or Shift+Option+Command+P (Macintosh). All Flash objects and movies are set to Play.

Inserting Shockwave movies

Shockwave, the Macromedia standard for interactive multimedia on the web, is a compressed format that allows media files created in Macromedia Director to be downloaded quickly and played by most popular browsers.

The software that plays Shockwave movies is available as both a Netscape Navigator plug-in and an ActiveX control. When you insert a Shockwave movie, Dreamweaver uses both the `object` tag (for the ActiveX control) and the `embed` tag (for the plug-in) to get the best results in all browsers. When you make changes in the Property inspector for the movie, Dreamweaver maps your entries to the appropriate parameters for both the `object` and `embed` tags.

To insert a **Shockwave** movie:

1 In the Document window, place the insertion point where you want to insert a Shockwave movie and then do one of the following:



- In the Insert bar, select Media and then click the Shockwave icon.
 - In the Insert bar, select Media and then drag the Shockwave icon to the Document window, or to the Code view window if you are working in the code.
 - Choose Insert > Media > Shockwave.
- 2 In the dialog box that appears, select a movie file.
- 3 In the Property inspector, enter the width and height of the movie in the W and H boxes.

Adding video

You can add video to your web page in different ways and using different formats. Video can be downloaded to the user or it can be streamed so that it plays while it is downloading. The most common streaming formats available on the web for the transmission of video files are RealMedia, QuickTime, and Windows Media. You must download a helper application to view these formats. With these formats, you can stream audio and video simultaneously.

If you'd like to include a short clip that can be downloaded rather than streamed, you can link to the clip or embed it in your page. These clips are often in the AVI or MPEG file format.

You can use Director to create Shockwave movies or Flash to create interactive, low-bandwidth, multimedia presentations for the web. With Flash, file size is surprisingly small, and the technology works across many platforms. (Of course, users must first download the free player plug-in before they can view these files.)

Adding sound to a page

There are several different types of sound files and formats, and several different ways to add sound to a web page. Some factors to consider before deciding on a format and method for adding sound are its purpose, your audience, file size, sound quality, and differences in browsers.

Note: Sound files are handled very differently and inconsistently by different browsers. You may want to add a sound file to a Flash movie, then embed the SWF file to improve consistency.

About audio file formats

The following list describes the more common audio file formats along with some of the advantages and disadvantages of each for web design.

.midi or .mid (Musical Instrument Digital Interface) format is for instrumental music. MIDI files are supported by many browsers and don't require a plug-in. Although their sound quality is very good, it can vary depending on a visitor's sound card. A small MIDI file can provide a long sound clip. MIDI files cannot be recorded and must be synthesized on a computer with special hardware and software.

.wav (Waveform Extension) format files have good sound quality, are supported by many browsers, and don't require a plug-in. You can record your own WAV files from a CD, tape, microphone, and so on. However, the large file size severely limits the length of sound clips that you can use on your web pages.

.aif (Audio Interchange File Format, or AIFF) format, like WAV format, has good sound quality, can be played by most browsers, and doesn't require a plug-in; you can also record AIFF files from a CD, tape, microphone, and so on. However, the large file size severely limits the length of sound clips that you can use on your web pages.

.mp3 (Motion Picture Experts Group Audio, or MPEG-Audio Layer-3) format is a compressed format that makes sound files substantially smaller. The sound quality is very good: if an MP3 file is recorded and compressed properly, its quality can rival that of a CD. New technology lets you "stream" the file so that a visitor doesn't have to wait for the entire file to download before hearing it. However, the file size is larger than a Real Audio file, so a whole song could still take quite a while to download over a normal phone line connection. To play MP3 files, visitors must download and install a helper application or plug-in such as QuickTime, Windows Media Player or RealPlayer.

.ra, .ram, .rpm, or Real Audio format has a very high degree of compression with smaller file sizes than MP3. Whole song files can be downloaded in a reasonable amount of time. Because the files can be "streamed" from a normal web server, visitors can begin listening to the sound before the file has completely downloaded. The sound quality is poorer than that of MP3 files, but new players and encoders have improved quality considerably. Visitors must download and install the RealPlayer helper application or plug-in to play these files.

Linking to an audio file

Linking to an audio file is a simple and effective way to add sound to a web page. This method of incorporating sound files lets visitors choose whether they want to listen to the file, and makes the file available to the widest audience. (Some browsers may not support embedded sound files.) See "Embedding a sound file" on page 328.

To create a link to an audio file:

- 1 Select the text or image you want to use as the link to the audio file.
- 2 In the Property inspector, click the folder icon to browse for the audio file, or type the file's path and name in the Link field.

Embedding a sound file

Embedding audio incorporates the sound player directly into the page, but the sound only plays if visitors to your site have the appropriate plug-in for the chosen sound file. Embed files if you want to use the sound as background music, or if you want more control over the sound presentation itself. For example, you can set the volume, the way the player looks on the page, and the beginning and ending points of the sound file.

To embed an audio file:

- 1 In Design view, place the insertion point where you want to embed the file and then do one of the following:
 - In the Insert bar, select Media then click the Plugin icon.
 - In the Insert bar, select Media then drag the Plugin icon to the Document window, or to the Code view window if you are working in the code.
 - Choose Insert > Media > Plugin.

For more information about the Plugin object, see "Inserting Netscape Navigator plug-in content" on page 329.

- 2 In the Property inspector, click the folder icon to browse for the audio file, or type the file's path and name in the Link field.
- 3 Enter the width and height by entering the values in the appropriate fields or by resizing the plug-in placeholder in the Document window.

These values determine the size at which the audio controls are displayed in the browser. For example, try a width of 144 pixels and a height of 60 pixels to see how the audio player appears in both Navigator and Internet Explorer.

Inserting Netscape Navigator plug-in content

Plug-ins enhance Netscape Navigator, providing ways to view media content in a wide variety of formats. Plug-ins are the means by which content files are played and displayed on your website. For example, typical plug-ins include RealPlayer and QuickTime, while some content files themselves include MP3s and QuickTime movies.

After you create content for a Navigator plug-in, you can use Dreamweaver to insert that content into an HTML document. Dreamweaver uses the `embed` tag to mark the reference to the content file.

To insert Navigator plug-in content:

- 1 In the Design view of the Document window, place the insertion point where you want to insert the content, then do one of the following:



- In the Insert bar, select Media and then click the Plugin icon.
 - Choose Insert > Media > Plugin.
- 2 In the dialog box that appears, select a content file for a Navigator plug-in.

Playing plug-ins in the Document window

You can preview movies and animations that rely on Navigator plug-ins—that is, elements that use the `embed` tag—directly in the Design view of the Document window. (You cannot preview movies or animations that rely on ActiveX controls in the Document window.) You can play all plug-in elements at one time to see how the page will look to the user, or you can play each one individually to ensure that you have embedded the correct media element.

To play movies, the proper plug-ins must be installed on your computer. When Dreamweaver starts up, it automatically searches for all installed plug-ins, looking first in the Configuration/Plugins folder and then in the plugin folders of all installed browsers.

To play plug-in content in the Document window:

- 1 Insert one or more media elements by choosing Insert > Media > Shockwave, Insert > Media > Flash, or Insert > Media > Plugin.
- 2 Play plug-in content:
 - Select one of the media elements you have inserted, and choose View > Plugins > Play or click the Play button in the Property inspector.
 - Choose View > Plugins > Play All to play all of the media elements on the selected page that rely on plug-ins.

Note: Play All only applies to the current document; it does not apply to other documents in a frameset, for example.

To stop playing plug-in content:

Select a media element and choose View > Plugins > Stop, or click the Stop button in the Property inspector.

You can also choose View > Plugins > Stop All to stop all plug-in content from playing.

Troubleshooting Navigator plug-ins

If you have followed the steps to play plug-in content in the Document window, but some of the plug-in content does not play, try the following:

- Make sure the associated plug-in is installed on your computer, and that the content is compatible with the version of the plug-in you have.
- Open the file Configuration/Plugins/UnsupportedPlugins.txt in a text editor and look to see if the problematic plug-in is listed. This file keeps track of plug-ins that cause problems in Dreamweaver and are therefore unsupported. (If you experience problems with a particular plug-in, consider adding it to this file.)
- Check that you have enough memory (and on the Macintosh, check that enough memory is allocated to Dreamweaver). Some plug-ins require an additional 2 to 5 MB of memory to run.

Inserting an ActiveX control

ActiveX controls (formerly known as OLE controls) are reusable components, somewhat like miniature applications, that can act like browser plug-ins. They run in Internet Explorer with Windows, but they don't run on the Macintosh or in Netscape Navigator. The ActiveX object in Dreamweaver lets you supply attributes and parameters for an ActiveX control in your visitor's browser.

Dreamweaver uses the `object` tag to mark the place on the page where the ActiveX control will appear, and to provide parameters for the ActiveX control.

To insert ActiveX control content:

- 1 In the Document window, place the insertion point where you want to insert the content and do one of the following:



- In the Insert bar, select Media and then click the ActiveX icon.
- In the Insert bar, select Media and then drag the ActiveX icon to the Document window (or to the Code view window if you are working in the code).
- Choose Insert > Media > ActiveX.

An icon marks where the ActiveX control will appear on the page in Internet Explorer.

Inserting a Java applet

Java is a programming language that allows the development of lightweight applications (*applets*) that can be embedded in web pages.

After you create a Java applet, you can insert it into an HTML document using Dreamweaver. Dreamweaver uses the `applet` tag to mark the reference to the applet file.

To insert a Java applet:

1 In the Document window, place the insertion point where you want to insert the applet, then do one of the following:



- In the Insert bar, select Media and then click the Applet icon.
 - In the Insert bar, select Media and then drag the Applet icon to the Document window (or to the Code view window if you are working in the code).
 - Choose Insert > Media > Applet. You can also drag the Flash button icon over to the Document window.
- 2 In the dialog box that appears, select a file containing a Java applet.

Using behaviors to control media

You can add behaviors to your page to start and stop various media objects.

Control Shockwave or Flash lets you play, stop, rewind, or go to a frame in a Shockwave or Flash movie (see “Control Shockwave or Flash” on page 357).

Play Sound lets you play a sound; for example, you can play a sound effect whenever the user moves the mouse pointer over a link (see “Play Sound” on page 363).

Check Plugin lets you check to see if visitors to your site have the required plug-in installed, then route them to different URLs, depending on whether they have the right plug-in. This only applies to Netscape Navigator Plug-ins, as the Check plugin behavior does not check for ActiveX controls. For more information, see “Check Plugin” on page 356.

CHAPTER 23

Dreamweaver and Accessibility

Accessibility refers to making websites and web products usable for people with visual, auditory, motor, and other disabilities. Macromedia Dreamweaver MX provides tools that make the product accessible and it provides tools that help you author accessible content.

Examples of accessibility features for software products and websites include screen reader support, text equivalents for graphics, keyboard shortcuts, change of display colors to high contrast, and so on.

As the number of people with disabilities who access the web continues to grow, it is increasingly important that developers make their products and websites accessible to all potential users. To that end, the U.S. government and other organizations have established legislation and guidelines to ensure that developers produce accessible content. For more information about two significant initiatives, see the World Wide Web Consortium Web Accessibility Initiative (<http://www.w3.org/wai>) and Section 508 of the Federal Rehabilitation Act (<http://www.section508.gov>).

If you are a Dreamweaver web designer who needs to use the Dreamweaver accessibility features, this chapter tells you about Dreamweaver screen reader support, keyboard navigation, and operating system accessibility support.

If you are a Dreamweaver web designer who needs to create accessible content, this chapter tells you about using the Dreamweaver Accessibility dialog boxes and testing your site for accessibility.

Designing accessible websites requires you to understand accessibility requirements and make many ongoing subjective decisions. Dreamweaver aids you in creating accessible websites. For example, Dreamweaver enables you to add text equivalents for graphics—and can even remind you to do so if you forget. However, no authoring tool can automate the development process. Designers need to think about how users with disabilities interact with webpages. The best way to ensure that a website is accessible is through deliberate planning, development, testing, and evaluation.

This chapter covers the following topics:

- “Using Dreamweaver accessibility features” on page 334
- “Authoring for accessibility” on page 337
- “Testing your website for accessibility” on page 343

Using Dreamweaver accessibility features

Dreamweaver provides features that make it accessible to users with disabilities. Specifically, Dreamweaver supports screen readers, operating system accessibility features, and keyboard navigation.

Note: The Dreamweaver 4 Workspace and Dreamweaver 5 Workspace modes both support Dreamweaver accessibility features, but we recommend that you use the Dreamweaver 4 workspace. To change modes, select **Edit > Preferences** or **Dreamweaver > Preferences** (Mac OS X), then select **General** from the category list on the left. Click the **Change Workspace** button, select **Dreamweaver 4 Workspace**, then click **OK**.

Using screen readers with Dreamweaver

A screen reader recites text that appears on the computer screen. It also reads non-textual information, such as button labels or image descriptions in the application, provided in accessibility tags or attributes during authoring.

Dreamweaver supports JAWS for Windows, from Freedom Scientific (<http://www.freedomscientific.com>), and Window Eyes screen readers, from GW Micro (<http://www.gwmicro.com>). As a Dreamweaver user, you can use a screen reader to assist you in creating your web pages. The screen reader starts reading in the top left corner of the Document window.

Using operating system accessibility features

Dreamweaver supports the Windows operating system high contrast setting. You activate this option through the Windows control panel. When high contrast is on, it affects Dreamweaver as follows:

- Dialog boxes and panels use system color settings.
For example, if you set the color to White on Black, then all Dreamweaver dialog boxes and panels appear with a white foreground color and black background.
- Code view syntax coloring is off.
Code view uses the system window and window text color, and ignores color settings in Preferences. For example, if you set the system color to White on Black, and then change text colors in Preferences > Code Coloring, Dreamweaver ignores the colors set in Preferences and displays the code text with a white foreground color and black background.
- Design view uses the background and text colors you set in **Modify > Page Properties** so that pages you design render colors as a browser will.

Using the keyboard to navigate Dreamweaver

You can use the keyboard to navigate Dreamweaver floating panels, Property inspector, dialog boxes, frames, and tables without a mouse.

This section covers the following topics:

- “Navigating panels” on page 335
- “Navigating the Property inspector” on page 335
- “Navigating dialog boxes” on page 336
- “Navigating frames” on page 336
- “Navigating tables” on page 336

Navigating panels

You can use the keyboard to navigate the panels.

Note: Tabbing and the use of arrow keys are supported for Windows only.

To navigate panels:

- 1 In the Document window, press Control+Alt+Tab to shift focus to a panel.
A white outline around the panel title bar indicates that focus is on that panel. The screen reader reads the panel title bar that has focus.
- 2 Press Control+Alt+Tab again to shift focus to the next panel.
Continue until you have focus on the panel you want to work in.
- 3 Press Control+Alt+Shift+Tab to shift focus to the previous panel, if necessary.
- 4 If the panel you want to work in is not open, use the keyboard shortcuts listed in the Windows menu to display the appropriate panel, then press Control+Alt+Tab to shift focus to that panel.
If the panel you want to work in is open, but not expanded, place focus on the panel title bar, and then press the Space bar. Press the Space bar again to collapse the panel.
- 5 Press the Tab key to move through the options in the panel.
A dotted outline around the option indicates that focus is on that option.
- 6 Use the arrow keys as appropriate:
 - If an option has choices, use the arrow keys to scroll through the choices, and then press the Space bar to make a selection.
 - If there are tabs in the panel group to open other panels, place focus on the open tab, and then use the left or right arrow key to open another tabs. Once you open a new tab, press the Tab key to move through the options in that panel.

Navigating the Property inspector

You can use the keyboard to navigate the Property inspector and make changes to your document.

Note: Tabbing and the use of arrow keys are supported for Windows only.

To navigate the Property inspector:

- 1 Press Control+F3 to display the Property inspector, if it is not visible.
- 2 Press Control+Alt+Tab until you shift focus to the Property inspector.
- 3 Press the Tab key to move through the Property inspector options.
- 4 Use the arrow keys as appropriate to move through option choices, and press Enter (Windows) or Return (Macintosh) to make a selection.
- 5 Press Control+Tab (Windows) or Option+Tab (Macintosh) to open and close the expanded section of the Property inspector, as necessary, or, with focus on the expander arrow at the bottom right, press the Space bar.

Navigating dialog boxes

You can use the keyboard to navigate dialog boxes.

Note: Tabbing and the use of arrow keys are supported for Windows only.

To navigate a dialog box:

- 1 Press the Tab key to move through the options in a dialog box.
- 2 Use the arrow keys to move through choices for an option.
For example, if an option has a pop-up menu, move focus to that option, and then use the down arrow to move through the choices.
- 3 If the dialog box has a Category list, press Control+Tab (Windows) or Option+Tab (Macintosh) to shift focus to the category list, and then use the arrow keys to move up or down the list.
- 4 Press Control+Tab (Windows) or Option+Tab (Macintosh) again to shift to the options for a category.
- 5 Press Enter (Windows) or Return (Macintosh) to exit the dialog box.

Navigating frames

If your document contains frames, you can use the arrow keys to shift focus to a frame.

Note: Tabbing and the use of arrow keys are supported for Windows only.

To select a frame:

- 1 Place the insertion point in the Document window.
- 2 Press Alt+Up Arrow to select the frame that currently has focus.
A dotted line indicates the frame that has focus.
- 3 Continue pressing Alt+Up Arrow to shift focus to the frameset, and then parent framesets, if there are nested framesets.
- 4 Press Alt+Down Arrow to shift focus to a child frameset or a single frame within the frameset.
- 5 With focus on a single frame, press Alt+Left or Right Arrow to move between frames.
- 6 Press Alt+Down Arrow to place the insertion point in the Document window.

Navigating tables

After you select a table, you can use the keyboard to navigate through it.

Note: Tabbing and the use of arrow keys are supported for Windows only.

To navigate a table:

- 1 In the Document window, do one of the following to select the table:
 - If the insertion point is to the left of the cell, press Shift+Right Arrow.
 - If the insertion point is to the right of the cell, press Shift+Left Arrow.
- 2 Press the down arrow to position the cursor in the first cell.
- 3 Use the arrow keys or press Tab to move to other cells as necessary.

Tip: Pressing Tab in a right-most cell adds another row to the table.

- 4 To select a cell, press Control+A while the insertion point is in the cell.
- 5 To exit the table, use the Select All command (Control+A in Windows or Option+A in Macintosh) twice, and then press the up, left, or right arrow key.

Authoring for accessibility

Dreamweaver assists you in creating accessible pages that contain useful content for screen readers and comply with government guidelines (see <http://www.section508.gov>).

When you activate the accessibility dialog boxes (see “Activating the Accessibility dialog boxes” on page 337), Dreamweaver prompts you to enter accessibility attributes whenever you insert page elements.

Note: To see sample pages that were designed for accessibility, select File > New. In the New Document dialog box, select Page Design (Accessibility) from the category list, and then select a page from the Page Design (Accessibility) list. For more information, see “Working with the New Document dialog box” on page 108.

Developing content for screen readers

To make information accessible to screen readers and your website users, Dreamweaver makes it easy to add text equivalents for graphic elements of a page, and enables you to mark up tables and forms in HTML for screen readers and other assistive technologies.

For example, you can add a product image to your document and associate a description of “Boy’s red jacket size large.” Then, when the image appears on a page for a user with visual disabilities, the screen reader reads the description, and the user knows which product is displayed on the page.

Dreamweaver prompts you to add text equivalents for graphic elements and accessibility-related mark up when you activate the Accessibility dialog boxes as described in “Activating the Accessibility dialog boxes” on page 337.

Activating the Accessibility dialog boxes

When you create accessible pages, you need to associate information, such as labels and descriptions, with your page objects to make your content accessible to all users. To do this, activate the Accessibility dialog boxes so that Dreamweaver prompts you for the information you need to add for accessibility. These dialog boxes appear when you insert an object for which you activated the corresponding Accessibility dialog box.

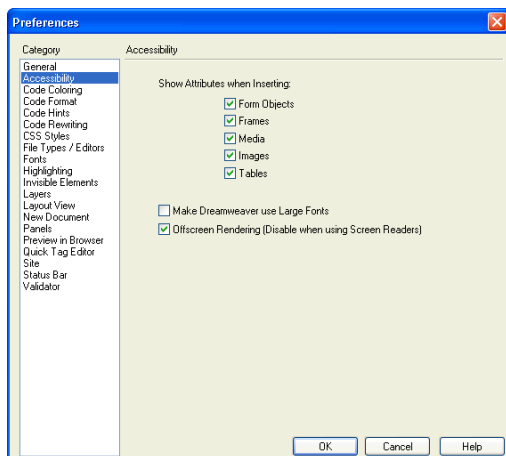
To activate the Accessibility dialog boxes:

- 1 Choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X).

The Preferences dialog box appears.

2 Select Accessibility from the category list on the left.

The Preferences dialog box displays accessibility options.



3 Complete the dialog box.

For more information, click the Help button in the dialog box.

4 Click OK.

For each element you select, an Accessibility dialog box prompts you to enter accessibility tags and attributes when you insert that element in a document.

For more information on adding accessible content to your website, see the following sections:

- “Inserting accessible images” on page 338
- “Inserting accessible form objects” on page 339
- “Inserting accessible frames” on page 340
- “Inserting accessible media objects” on page 341
- “Inserting accessible tables” on page 342

Inserting accessible images

When you insert an image, and you’ve selected the Images option in Accessibility preferences, Dreamweaver prompts you to enter information to make the image accessible.

Note: To make image placeholders and interactive images, such as a rollover images and navigation bars, accessible, enter Alternated text when you insert the image. For more information see “Inserting Images” on page 297.

To insert an accessible image:

- 1** Place the insertion point in the Document window where you want the image to appear.
- 2** Do one of the following:
 - Choose Insert > Image.
 - Click the Image button on the Common tab of the Insert bar.

- Drag the Image icon from the Insert bar into the document.

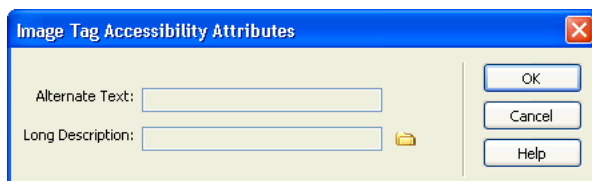
The Image Source dialog box appears.

- 3 Click Browse to choose a file or type the path for the image file.

If you are working in an unsaved document, Dreamweaver generates a file location reference to the image file. When you save the document anywhere in the site, Dreamweaver converts the reference to a document-relative path.

- 4 Click OK.

The Image Tag Accessibility Attributes dialog box appears.



- 5 Enter values in the Alt and Longdesc text boxes.

For more information, click the Help button in the dialog box.

Note: You can enter one or both text boxes depending on your needs.

- 6 Click OK.

The image appears in your document.

Note: If you press Cancel, the image appears in the document, but Dreamweaver does not associate accessibility tags or attributes with it.

To edit accessibility values for an image:

- 1 In the Document window, select the image.
- 2 Do one of the following:
 - Edit the appropriate image attributes in Code view.
 - Right-click (Windows) or Control-click (Macintosh), and then choose Edit Tag Code.
 - Edit the Alt value in the Property inspector.

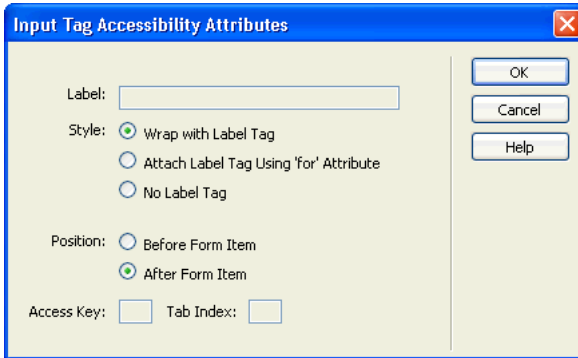
Inserting accessible form objects

When you insert a form object, and you've selected the Form Object option in Accessibility preferences, Dreamweaver prompts you to enter information to make the form object accessible.

To add an accessible form object:

- 1 In the document, place the insertion point where you want the form to appear.
- 2 Do one of the following to insert a form object:
 - Choose Insert > Form Object, and then select a form object to insert.
 - Select the Forms tab in the Insert bar, and then click an object button.

- Select the Forms tab in the Insert bar, and then drag the appropriate icon into the document. The Input Tag Accessibility Attributes dialog box appears.



- 3 Complete the dialog box.

For more information, click the Help button in the dialog box.

- 4 Click OK.

Note: If you press Cancel, the form object appears in the document, but Dreamweaver does not associate accessibility tags or attributes with it.

- 5 If Dreamweaver asks if you want to insert a form tag, click Yes.

The form object appears in the document.

To edit accessibility values for a form object:

- 1 In the Document window, select the object.
- 2 Do one of the following:
 - Edit the appropriate attributes in Code view.
 - Right-click (Windows) or Control-click (Macintosh), and then choose Edit Tag Code.

To make an accessible form object dynamic:

- 1 In the Document window, select the form object you want to make dynamic.
- 2 Right-click (Windows) or Control-click (Macintosh), and then choose Edit Tag Code.
- 3 Edit attributes as necessary to make the object dynamic.

Note: You cannot use the Bindings panel to add an accessible dynamic form object.

Inserting accessible frames

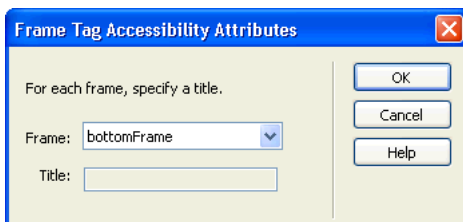
When you insert a frameset, and you've selected the Images option in Accessibility preferences, Dreamweaver prompts you to enter a name for each frame to make it accessible.

To insert an accessible frameset:

- 1 Place the insertion point in the document.

- 2 Do one of the following to insert a frameset:
 - Choose Insert > Frames, then select a frameset.
 - Click the Frames tab in the Insert bar, then click the appropriate frameset button.
 - Click the Frames tab in the Insert bar, then drag the appropriate frameset icon into the document.

The frameset appears in your document and the Frame Tag Accessibility Attributes dialog box appears.



- 3 Select a frame from the pop-up menu, and then enter a name for the frame.
For more information, click the Help button in the dialog box.
Tip: Choose Windows > Others > Frames to view a diagram of the frames you are naming.
- 4 Repeat the previous step until you name each frame.
- 5 Click OK.

Note: If you press Cancel, the frameset appears in the document, but Dreamweaver does not associate accessibility tags or attributes with it.

To edit accessibility values for a frame:

- 1 Display Code view or Split view for your document, if you're currently viewing design.
- 2 Choose Windows > Others > Frames to open the Frames panel.
- 3 Select a frame by placing the insertion point in one of the frames.
Dreamweaver highlights the frame tag in the code.
- 4 Do one of the following:
 - Edit the code in Code view.
 - Right-click (Windows) or Control-click (Macintosh), and then choose Edit Tag Code.

Inserting accessible media objects

When you insert a media object, and you've selected the Media option in Accessibility preferences, Dreamweaver prompts you to enter information to make the media object accessible.

To insert an accessible media object:

- 1 Place the insertion point in the Document window where you want the media object to appear.

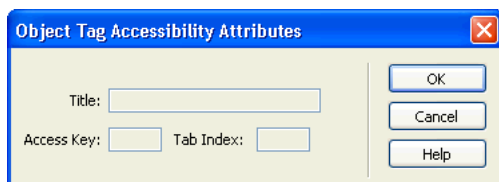
2 Do one of the following:

- Choose Insert > Media, then select an item.
- Click the Media tab in the Insert bar, then click an object button.
- Click the Media tab in the Insert bar, then drag the appropriate icon into the document.
A Select File or an Insert Flash dialog box appears.

Note: For Active X objects, the accessibility dialog box appears. For these objects, skip the next step.

3 Complete the Select File or Insert Flash dialog box, and then click OK.

The Object Tag Accessibility Attributes dialog box appears.



4 Complete the dialog box.

For more information, click the Help button in the dialog box.

5 Click OK.

The media object appears in the document.

Note: If you press Cancel, a media object placeholder appears in the document, but Dreamweaver does not associate accessibility tags or attributes with it.

To edit accessibility values for a media object:

1 In the Document window, select the object.

2 Do one of the following:

- Edit the appropriate attributes in Code view.
- Right-click (Windows) or Control-click (Macintosh), and then choose Edit Tag Code.

Inserting accessible tables

When you insert a table, and you've selected the Tables option in Accessibility preferences, Dreamweaver prompts you to enter a name for each table to make it accessible.

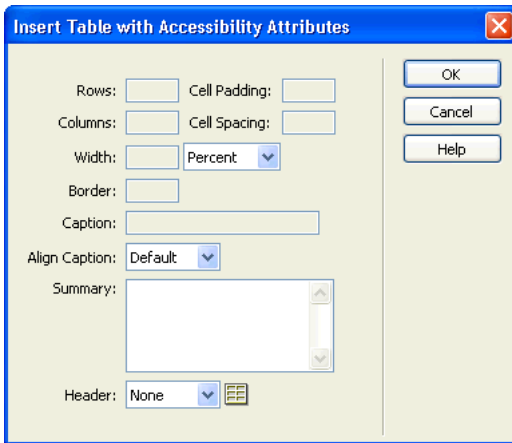
To insert an accessible table:

1 Place the insertion point in the Document window where you want the table to appear.

2 Do one of the following:

- Choose Insert > Table.
- Click the Table button on the Common tab in the Insert bar.

- Drag the Table icon from the Common tab in the Insert bar into the document.
The Insert Table with Accessibility Attributes dialog box appears.



3 Complete the dialog box.

For more information, click the Help button in the dialog box.

4 Click OK.

The table appears in the document.

Note: If you press Cancel, the table does not appear in the document.

To edit accessibility values for a table:

- 1 In the Document window, select the table.
- 2 Do one of the following:
 - Edit the appropriate table attributes in Code view.
 - Right-click (Windows) or Control-click (Macintosh), and then choose Edit Tag Code.

Testing your website for accessibility

The best way to ensure that your site is accessible to all users is to design your site according to the Section 508 accessibility guidelines of the 1998 Rehabilitation Act. Then, use the Dreamweaver reports feature to check that you have implemented the guidelines in your site.

You can run an Accessibility report on the current document, selected files, a folder, or an entire site.

To run an Accessibility report on the current document:

- 1 Open the document you want to check.
- 2 Choose File > Check Page > Check Accessibility.

The report results appear in the Site Reports panel (in the Results panel group).

To run an Accessibility report on selected content:

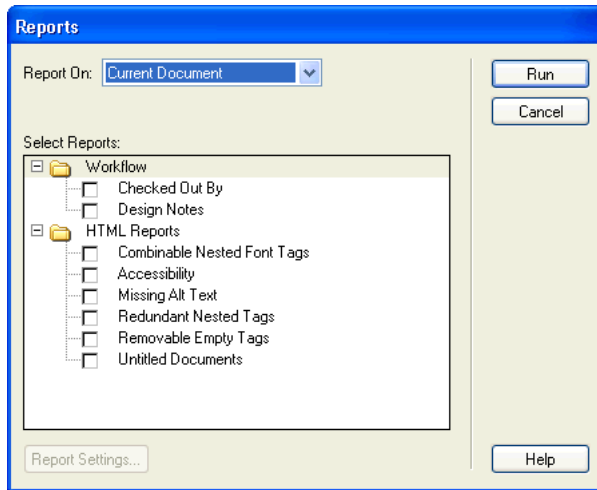
1 Do one of the following to select the content you want to check:

- Open a document.
- Select files in the Site panel.
- Select a site from the Site pop-up menu in the Site panel.

Note: To run an Accessibility report on a folder, you do not need to select the folder now; you select the folder in the Reports dialog box.

2 Choose Site > Reports

The Reports dialog box appears.



3 Select the content you want to report on from the Report On pop-up menu, and then select Accessibility.

Note: If you select Folder from the Report On pop-up menu, click the Browse to select a folder.

4 Click Run.

The report results appear in the Site Reports panel (in the Results panel group).

To view and edit reported problems:

1 Run an Accessibility report on selected website content, if you haven't already.

The report results appear in the Site Reports panel (in the Results panel group). Select any line in the report, and click the More Info button on the left side of the Site Reports panel for an description of the problem and suggestions for fixing the problem.

The information appears in the Reference panel (in the Code panel group).

2 Double click any line in the report to view the corresponding code in the Document window.

Note: If you are in Design view, Dreamweaver changes the display to split view to show the reported problem in code.

- 3 In the Document window, in Code view, edit or add attributes to the appropriate tags to make your content comply with accessibility guidelines.

To use the tag editor to assist in coding, select code, right-click (Windows) or Control-click (Macintosh), and choose tag editor. For more information about tags and attributes, select O'Reilly HTML Reference in the Reference panel (in the Code panel group), and then select a tag.

To save an accessibility report:

Click the Save button on the left side of the Site Reports panel.

About this feature

The accessibility validation feature in Dreamweaver MX uses technology from UsableNet. UsableNet is an industry leader in developing easy-to-use software to automate usability and accessibility testing and repair. For additional assistance with accessibility testing, try the UsableNet LIFT for Macromedia Dreamweaver, a complete solution for developing usable and accessible websites. UsableNet Lift for Macromedia Dreamweaver includes fix wizards for complex tables, forms, and images; a global ALT editor; customizable reporting; and a new active monitoring mode that ensures content is accessible as pages are being built. Request a demo of Lift for Macromedia Dreamweaver at <http://www.usablenet.com>.

UsableNet™

Part VI

Working with Behaviors and Animations

Many web pages contain only text and images, with no interactive elements. Use JavaScript behaviors and layer-animation timelines in Dreamweaver to provide interactivity and animation to capture the interest of your visitors.

This part contains the following chapters:

- Chapter 24, “Using JavaScript Behaviors”
- Chapter 25, “Animating Layers”

CHAPTER 24

Using JavaScript Behaviors

Dreamweaver behaviors place JavaScript code in documents to allow visitors to interact with a web page to change the page in various ways, or to cause certain tasks to be performed. A behavior is a combination of an event with an action triggered by that event. In the Behaviors panel, you add a behavior to a page by specifying an action and then specifying the event that triggers that action.

Note: Behavior code is client-side JavaScript code; that is, it runs in browsers, not on servers.

Events are, effectively, messages generated by browsers indicating that a visitor to your page has done something. For example, when a visitor moves the pointer over a link, the browser generates an `onMouseOver` event for that link; the browser then checks to see whether there's some JavaScript code (specified in the page being viewed) that the browser is supposed to call when that event is generated for that link. Different events are defined for different page elements; for example, in most browsers `onMouseOver` and `onClick` are events associated with links, whereas `onLoad` is an event associated with images and with the `body` section of the document.

An action consists of prewritten JavaScript code that performs a specific task, such as opening a browser window, showing or hiding a layer, playing a sound, or stopping a Shockwave movie. The actions provided with Macromedia Dreamweaver MX are carefully written by Dreamweaver engineers to provide maximum cross-browser compatibility.

After you attach a behavior to a page element, whenever the event you've specified occurs for that element, the browser calls the action (the JavaScript code) that you've associated with that event. (The events that you can use to trigger a given action vary from browser to browser.) For example, if you attach the Popup Message action to a link and specify that it will be triggered by the `onMouseOver` event, then whenever someone points to that link with the mouse pointer in the browser, your message pops up in a dialog box.

A single event can trigger several different actions, and you can specify the order in which those actions occur.

Dreamweaver MX provides about two dozen behavior actions; additional actions can be found on the Macromedia Exchange website as well as on third-party developer sites. (See "Downloading and installing third-party behaviors" on page 354.) You can write your own behavior actions if you are proficient in JavaScript. For more information on writing behavior actions, see *Extending Dreamweaver*.

Note: The terms *behavior* and *action* are Dreamweaver terms, not HTML terms. From the browser's point of view, an action is just like any other piece of JavaScript code.

This chapter contains these sections:

- “Using the Behaviors panel” on page 350
- “About events” on page 350
- “Applying a behavior” on page 351
- “About behaviors and text” on page 352
- “Attaching a behavior to a timeline” on page 352
- “Changing a behavior” on page 353
- “Updating a behavior” on page 353
- “Creating new actions” on page 353
- “Downloading and installing third-party behaviors” on page 354
- “Using the behavior actions that come with Dreamweaver” on page 354

Using the Behaviors panel

Use the Behaviors panel to attach behaviors to page elements (more specifically to tags) and to modify parameters of previously attached behaviors.

To open the Behaviors panel, choose Window > Behaviors.

Behaviors that have already been attached to the currently selected page element appear in the behavior list (the main area of the panel), listed alphabetically by event. If there are several actions for the same event, the actions will execute in the order in which they appear on the list. If no behaviors appear in the behavior list, then no behaviors have been attached to the currently selected element.

For more information about the options in the Behaviors panel, choose Help from the Options menu in the panel group's title bar.

About events

Each browser provides a set of events that you can associate with the actions listed in the Behaviors panel's Actions (+) pop-up menu. When a visitor to your web page interacts with the page—for example, by clicking an image—the browser generates events; those events can be used to call JavaScript functions that cause an action to occur. (Events can also be generated without user interaction, such as when you set a page to automatically reload every 10 seconds.) Dreamweaver supplies many common actions that you can trigger using these events.

For names and descriptions of the events provided by each browser, see the Dreamweaver Support Center at <http://www.macromedia.com/support/dreamweaver/>.

Note that most events can be used only with certain page elements. To find out what events a given browser supports for a given page element, insert the page element in your document and attach a behavior to it, then look at the Events pop-up menu in the Behaviors panel. For a detailed advanced look at precisely which tags can be used with a given event in a given browser, search for the event in one of the files in your Dreamweaver/Configuration/Behaviors/Events folder.

Applying a behavior

You can attach behaviors to the entire document (that is, to the `body` tag) or to links, images, form elements, or any of several other HTML elements. The target browser you choose determines which events are supported for a given element. Internet Explorer 4.0, for example, has a much wider array of events for each element than Navigator 4.0 or any 3.0 browser.

Note: You can't attach a behavior to plain text. For more information, see "About behaviors and text" on page 352.

You can specify more than one action for each event. Actions occur in the order in which they're listed in the Actions column of the Behaviors panel. For information on changing the order of actions, see "Changing a behavior" on page 353.

To attach a behavior:

- 1 Select an element on the page, such as an image or a link.

To attach a behavior to the entire page, click the `<body>` tag in the tag selector at the bottom left of the Document window.

- 2 Choose Window > Behaviors to open the Behaviors panel.
- 3 Click the plus (+) button and choose an action from the Actions pop-up menu.

Actions that are dimmed in the menu can't be chosen. They may be dimmed because a required object doesn't exist in the current document. For example, the Play Timeline action is dimmed if the document has no timelines, and the Control Shockwave or Flash action is dimmed if the document contains no Shockwave or Flash movies. If no events are available for the selected object, all actions are dimmed.

When you choose an action, a dialog box appears, displaying parameters and instructions for the action.

- 4 Enter parameters for the action, and click OK.

All actions provided in Dreamweaver work in 4.0 and later browsers. Some actions do not work in older browsers. See "Using the behavior actions that come with Dreamweaver" on page 354.

- 5 The default event to trigger the action appears in the Events column. If this is not the trigger event you want, choose another event from the Events pop-up menu. (To open the Events pop-up menu, select an event or action in the Behaviors panel, and click the downward-pointing black arrow that appears between the event name and the action name.)

Different events appear in the Events pop-up menu depending on the selected object and on the browsers specified in the Show Events For submenu. Events may be dimmed if the relevant objects do not yet exist on the page or if the selected object cannot receive events. If the expected events don't appear, make sure the correct object is selected, or change the target browsers in the Show Events For pop-up menu.

If you're attaching a behavior to an image, some events (such as `onMouseOver`) appear in parentheses. These events are available only for links. When you choose one of them, Dreamweaver wraps a tag around the image to define a null link. The null link is represented by `javascript:;` in the Property inspector's Link text box. You can change the link value if you want to turn it into a real link to another page, but if you delete the JavaScript link without replacing it with another link, you will remove the behavior.

About behaviors and text

You can't attach a behavior to plain text. Tags like `p` and `span` don't generate events in browsers, so there's no way to trigger an action from those tags.

However, you can attach a behavior to a link. Therefore, to attach a behavior to text, the easiest approach is to add a null link (that doesn't point to anything) to the text, then attach the behavior to the link. Note that if you do this, your text will appear as a link. You can change the link color and remove the underlining if you really don't want it to look like a link, though site visitors may then be unaware that there's a reason to click that text.

To attach a behavior to the selected text:

- 1 In the Property inspector, enter `javascript:;` in the Link field. Be certain to include both the colon and the semicolon.

Note: You can instead use a number sign (`#`) in the Link field if you want. The problem with using a number sign is that when a visitor clicks the link, some browsers may jump to the top of the page. Clicking the JavaScript null link has no effect at all on the page, so the JavaScript approach is generally preferable.

- 2 With the text still selected, open the Behaviors panel (Window > Behaviors).
- 3 Choose an action from the Actions pop-up menu, enter parameters for the action, and choose an event to trigger the action. For details, see "Applying a behavior" on page 351.

To change the appearance of linked text to make it look like it isn't a link:

- 1 Open the Document window's Code view by choosing View > Code.
- 2 Find the link.
- 3 In the link's `a href` tag, insert this attribute: `style="text-decoration:none; color:black"`.

This attribute setting disables underlining and sets the color of the text to black. (Of course, if the surrounding text is a different color, use that color instead of black.)

Note that this attribute is an inline CSS style. An inline style applied to a single link overrides other CSS styles that apply to that link, but has no effect outside of that link. To change the appearance of linked text everywhere on a page or across your entire site, use CSS styles to create a new style for links. For details, see Chapter 19, "Inserting and Formatting Text," on page 271.

Attaching a behavior to a timeline

To trigger a behavior at a certain frame in a timeline (rather than having a visitor's interaction trigger it), place the behavior in the timeline. (For information on creating a timeline, see "Animating your layers" on page 388.) For example, you can start a sound playing at frame 15 of a timeline.

Only one kind of event can trigger an action in a timeline: the animation reaching a certain frame number (an `onFrame7` event, for example).

The behavior can affect any object on the page, not just objects in the timeline. Preview the timeline in a browser to see the behavior working. You cannot preview behaviors inside Dreamweaver.

To place a behavior in a timeline:

- 1 Click in a frame in the Behaviors channel in the Timelines panel.
- 2 Use the Behaviors panel to choose an action to perform at that frame.

The action appears in the Behaviors panel, with an event indicating the frame number at which the action is triggered. A minus sign (–) appears in the Behaviors channel of the timeline frame.

Changing a behavior

After attaching a behavior, you can change the event that triggers the action, add or remove actions, and change parameters for actions.

To change a behavior:

- 1 Select an object with a behavior attached.
- 2 Choose Window > Behaviors to open the Behaviors panel.

Behaviors appear in the panel alphabetically by event. If there are several actions for the same event, the actions appear in the order in which they will execute.

- 3 Choose from the following options:
 - To edit an action's parameters, double-click the behavior name, or select it and press Enter (Windows) or Return (Macintosh); then change parameters in the dialog box and click OK.
 - To change the order of actions for a given event, select an action and click the up or down arrow button.
 - To delete a behavior, select it and click the minus (–) button or press Delete.

Updating a behavior

If your pages contain behaviors created with Dreamweaver 1 or Dreamweaver 2, those behaviors are not updated automatically when you open the pages in the current version of Dreamweaver. However, when you update one occurrence of a behavior in a page (by following the procedure below), all other occurrences of that behavior in that page are also updated. Behaviors created in Dreamweaver 3 work without modification in Dreamweaver 4.

To update a behavior in a page:

- 1 Select an element that has the behavior attached to it.
- 2 Open the Behaviors panel.
- 3 Double-click the behavior.
- 4 Click OK in the behavior's dialog box.

All occurrences of that behavior in that page are updated.

Creating new actions

Actions consist of JavaScript and HTML code. If you are proficient with JavaScript, you can write new actions and add them to the Actions pop-up menu in the Behaviors panel. For more information, see *Extending Dreamweaver*.

Downloading and installing third-party behaviors

One of the most useful features of Dreamweaver is its extensibility—that is, it offers users who are proficient in JavaScript the opportunity to write JavaScript code that extends the capabilities of Dreamweaver. Many of these users have chosen to share their extensions with others by submitting them to the Macromedia Exchange for Dreamweaver website.

To download and install new behaviors from the Exchange site:

- 1 Open the Behaviors panel and choose Get More Behaviors from the Actions (+) pop-up menu.

Your primary browser opens, and the Exchange site appears. (You must be connected to the web to download behaviors.)

- 2 Browse or search for packages.
- 3 Download and install the extension package you want.

For details, see “Adding extensions to Dreamweaver” on page 53.

Using the behavior actions that come with Dreamweaver

The behavior actions included with Dreamweaver have been written to work in Netscape Navigator 4.0 and later, and in Internet Explorer 4.0 and later. Most of these behavior actions also work in Netscape Navigator version 3.0 and later. (The layer-related behaviors do not work in Navigator 3.0.) Most of these behavior actions fail silently in Internet Explorer version 3.0.

Note: The Dreamweaver actions have been carefully written to work in as many browsers as possible. If you remove code from a Dreamweaver action by hand, or replace it with your own code, you may lose cross-browser compatibility.

Although the Dreamweaver actions were written to maximize cross-browser compatibility, some actions do not work in older browsers. Also, some browsers do not support JavaScript at all, and many people who browse the web keep JavaScript turned off in their browsers. For best cross-platform results, provide alternative interfaces enclosed in `noscript` tags so that people without JavaScript can still use your site.

Call JavaScript

The Call JavaScript action lets you use the Behaviors panel to specify that a custom function or line of JavaScript code should be executed when an event occurs. (You can write the JavaScript yourself, or you can use code provided by various freely available JavaScript libraries on the web.)

To use the Call JavaScript action:

- 1 Select an object and open the Behaviors panel.
- 2 Click the plus (+) button and choose Call JavaScript from the Actions pop-up menu.
- 3 Type the exact JavaScript to be executed, or type the name of a function.

For example, to create a Back button, you might type
`if (history.length > 0){history.back()}. If you have encapsulated your code in a function, type only the function name (for example, hogback()).`

- 4 Click OK.
- 5 Check that the default event is the one you want.

If it isn't, choose another event from the pop-up menu. If the events you want are not listed, change the target browser in the Show Events For pop-up menu.

Change Property

Use the Change Property action to change the value of one of an object's properties (for example, the background color of a layer or the action of a form). The properties you can affect are determined by the browser; many more properties can be changed by this behavior in Internet Explorer 4.0 than in IE 3.0 or Navigator 3.0 or 4.0. For example, you can set the background color of a layer dynamically.

Note: Use this action only if you are very familiar with HTML and JavaScript.

To use the Change Property action:

- 1 Select an object and open the Behaviors panel.
- 2 Click the plus (+) button and choose Change Property from the Actions pop-up menu.
- 3 From the Type of Object pop-up menu, choose the type of object whose property you want to change.

The Named Object pop-up menu now lists all the named objects of the type you chose.

- 4 Choose an object from the Named Object pop-up menu.
- 5 Choose a property from the Property pop-up menu, or enter the name of the property in the text field.

To see the properties that can be changed in each browser, choose different browsers or browser versions from the browser pop-up menu. If you are typing a property name, be sure to use the exact JavaScript name of the property (and remember that JavaScript properties are case-sensitive).

- 6 Enter the new value for the property in the New Value field, and click OK.
- 7 Check that the default event is the one you want. (When the event occurs, the action will execute and the property will change.)

If it isn't, choose another event from the pop-up menu. If the events you want are not listed, change the target browser in the Show Events For pop-up menu.

Check Browser

Use the Check Browser action to send visitors to different pages depending on their browser brands and versions. For example, you might want visitors to go to one page if they have Navigator 4.0 or later, to go to another page if they have Internet Explorer 4.0 or later, and to stay on the current page if they have any other kind of browser.

It's useful to attach this behavior to the `body` tag of a page that is compatible with practically any browser (and that does not use any other JavaScript); this way, visitors who come to the page with JavaScript turned off will still see something.

Another option is to attach this behavior to a null link (such as ``) and have the action determine the link's destination based on the visitor's browser brand and version.

To use the Check Browser action:

- 1 Select an object and open the Behaviors panel.
- 2 Click the plus (+) button and choose Check Browser from the Actions pop-up menu.

- 3 Determine how you want to separate your visitors: by browser brand, by browser version, or both.
For example, do you want everyone with a 4.0 browser to see one page, and all others to see a different page? Or perhaps you want Netscape Navigator users to see one page and Internet Explorer users to see another.
- 4 Specify a version of Netscape Navigator.
- 5 In the adjacent pop-up menus, choose options for what to do if the browser is the Netscape Navigator version you specified or later and what to do otherwise.
The options are Go to URL, Go to Alt URL, and Stay on This Page.
- 6 Specify a version of Internet Explorer.
- 7 In the adjacent pop-up menus, choose options for what to do if the browser is the Internet Explorer version you specified or later and what to do otherwise.
The options are Go to URL, Go to Alt URL, and Stay on This Page.
- 8 Choose an option from the Other Browsers pop-up menu to specify what to do if the browser is neither Navigator nor Internet Explorer. (For example, the visitor may be using a text-based browser like Lynx.)
Stay on This Page is the best option for browsers other than Navigator and IE because most do not support JavaScript—and if they cannot read this behavior, they will stay on the page anyway.
- 9 Enter the paths and filenames of the URL and the alternate URL in the text fields at the bottom of the dialog box. If you enter a remote URL, you must enter the http:// prefix in addition to the www address.
- 10 Click OK.
- 11 Check that the default event is the one you want.
If it isn't, choose another event from the pop-up menu. If the events you want are not listed, change the target browser in the Show Events For pop-up menu. Remember that the purpose of this behavior is to check for different browser versions, so it's best to choose an event that works on 3.0 and later browsers.

Check Plugin

Use the Check Plugin action to send visitors to different pages depending on whether they have the specified plug-in installed. For example, you might want visitors to go to one page if they have Shockwave and another page if they do not.

Note: You cannot detect specific plug-ins in Internet Explorer using JavaScript. However, selecting Flash or Director will add the appropriate VBScript code to your page to detect those plug-ins in IE on Windows. Plug-in detection is impossible in Internet Explorer on the Macintosh.

To use the Check Plugin action:

- 1 Select an object and open the Behaviors panel.
- 2 Click the plus (+) button and choose Check Plugin from the Actions pop-up menu.

- 3 Choose a plug-in from the Plugin pop-up menu, or click Enter and type the exact name of the plug-in in the adjacent field.

You must use the exact name of the plug-in as specified in bold on the About Plug-ins page in Navigator. (In Windows, choose Navigator's Help > About Plug-ins command; on the Macintosh, choose About Plug-ins from the Apple menu.)

- 4 In the If Found, Go To URL field, specify a URL for visitors who have the plug-in.

If you specify a remote URL, you must include the `http://` prefix in the address.

To make visitors with the plug-in stay on the same page, leave this field blank.

- 5 In the Otherwise, Go To URL field, specify an alternative URL for visitors who don't have the plug-in.

To make visitors without the plug-in stay on the same page, leave this field blank.

- 6 Plug-in detection is not possible in Internet Explorer on the Macintosh, and most plug-ins cannot be detected in Internet Explorer on Windows. By default, when detection is impossible, the visitor is sent to the URL listed in the Otherwise field. To instead send the visitor to the first (If Found) URL, select the Always go to first URL if detection is not possible option. When selected, this option effectively means "assume that the visitor has the plug-in, unless the browser explicitly indicates that the plug-in is not present."

In general, if the plug-in content is integral to your page, select the "Always go to first URL if detection is not possible" option; visitors without the plug-in will often be prompted by the browser to download the plug-in. If the plug-in content is not essential to your page, leave this option unselected.

This option applies only to Internet Explorer; Navigator can always detect plug-ins.

- 7 Click OK.

- 8 Check that the default event is the one you want.

If it isn't, choose another event from the pop-up menu. If the events you want are not listed, change the target browser in the Show Events For pop-up menu.

Control Shockwave or Flash

Use the Control Shockwave or Flash action to play, stop, rewind, or go to a frame in a Macromedia Shockwave or Macromedia Flash movie.

To use the Control Shockwave or Flash action:

- 1 Choose Insert > Media > Shockwave or Insert > Media > Flash to insert a Shockwave or Flash movie, respectively.
- 2 Choose Window > Properties and enter a name for the movie in the upper leftmost field (next to the Shockwave or Flash icon). You must name the movie to control it with the Control Shockwave or Flash action.
- 3 Select the item you want to use to control the Shockwave or Flash movie. For example, if you have an image of a "Play" button that will be used to make the movie play, select that image.
- 4 Open the Behaviors panel (Window > Behaviors).
- 5 Click the plus (+) button and choose Control Shockwave or Flash from the Actions pop-up menu. A parameters dialog box appears.

- 6 Choose a movie from the Movie pop-up menu.
Dreamweaver automatically lists the names of all Shockwave and Flash movies in the current document. (Specifically, Dreamweaver lists movies with filenames ending in .dcr, .dir, .swf, or .spl that are in `object` or `embed` tags.)
- 7 Choose to play, stop, rewind, or go to a frame in the movie. The Play option plays the movie starting from the frame where the action occurs.
- 8 Click OK.
- 9 Check that the default event is the one you want.
If it isn't, choose another event from the pop-up menu. If the events you want are not listed, change the target browser in the Show Events For submenu of the Events pop-up menu.

Drag Layer

The Drag Layer action lets the visitor drag a layer. Use this action to create puzzles, slider controls, and other movable interface elements.

You can specify in which direction the visitor can drag the layer (horizontally, vertically, or in any direction), a target to which the visitor should drag the layer, whether to snap the layer to the target if the layer is within a certain number of pixels of the target, what to do when the layer hits the target, and more.

Because the Drag Layer action must be called before the layer can be dragged by the visitor, make sure the event that triggers the action occurs before the visitor attempts to drag the layer. It's best to attach Drag Layer to the `body` object (with the `onLoad` event), though you can also attach it to a link that fills the entire layer (such as a link around an image) using the `onMouseOver` event.

To use the Drag Layer action:

- 1 Choose Insert > Layer or click the Draw Layer button on the Insert bar and draw a layer in the Document window's Design view.
- 2 Select the `body` tag by clicking `<body>` in the tag selector at the bottom of the Document window.
- 3 Open the Behaviors panel.
- 4 Click the plus (+) button and choose Drag Layer from the Actions pop-up menu.
If Drag Layer is unavailable, you probably have a layer selected. Because layers do not accept events in both 4.0 browsers, you must select a different object—such as the `body` tag or a link (a `tag`)—or change the target browser to IE 4.0 in the Show Events For pop-up menu.
- 5 In the Layer pop-up menu, select the layer that you want to make draggable.
- 6 Choose either Constrained or Unconstrained from the Movement pop-up menu.
Unconstrained movement is appropriate for puzzles and other drag-and-drop games. For slider controls and moveable scenery such as file drawers, curtains, and mini-blinds, choose constrained movement.
- 7 For constrained movement, enter values (in pixels) in the Up, Down, Left, and Right fields.
Values are relative to the starting position of the layer. To constrain movement within a rectangular region, enter positive values in all four fields. To allow only vertical movement, enter positive values for Up and Down and 0 for Left and Right. To allow only horizontal movement, enter positive values for Left and Right and 0 for Up and Down.

- 8 Enter values (in pixels) for the drop target in the Left and Top fields.

The drop target is the spot to which you want the visitor to drag the layer. A layer is considered to have reached the drop target when its left and top coordinates match the values you enter in the Left and Top fields. Values are relative to the top left corner of the browser window. Click Get Current Position to automatically fill the fields with the current position of the layer.

- 9 Enter a value (in pixels) in the Snap if Within field to determine how close the visitor must get to the drop target before the layer snaps to the target.

Larger values make it easier for the visitor to find the drop target.

- 10 For simple puzzles and scenery manipulation, you can stop here. To define the drag handle for the layer, track the movement of the layer while it is being dragged, and trigger an action when the layer is dropped, click the Advanced tab.

- 11 To specify that the visitor must click a particular area of the layer to drag the layer, choose Area Within Layer from the Drag Handle pop-up menu; then enter the left and top coordinates and the width and height of the drag handle.

This option is useful when the image inside the layer has an element that suggests dragging, such as a title bar or drawer handle. Do not set this option if you want the visitor to be able to click anywhere in the layer to drag it.

- 12 Choose any While Dragging options that you want to use:

- Select Bring Layer to Front if the layer should move to the front of the stacking order while it is being dragged. If you select this option, use the pop-up menu to choose whether to leave the layer in front or restore it to its original position in the stacking order.
- Enter JavaScript code or a function name (for example, `monitorLayer()`) in the Call JavaScript field to repeatedly execute the code or function while the layer is being dragged. For example, you could write a function that monitors the coordinates of the layer and displays hints such as “you’re getting warmer” or “you’re nowhere near the drop target” in a text field.

- 13 Enter JavaScript code or a function name (for example, `evaluateLayerPos()`) in the second Call JavaScript field to execute the code or function when the layer is dropped. Select Only if Snapped if the JavaScript should be executed only if the layer has reached the drop target.

- 14 Click OK.

- 15 Check that the default event is the one you want.

If it isn’t, choose another event from the pop-up menu. If the events you want are not listed, change the target browser in the Show Events For pop-up menu. Remember that layers are not supported by 3.0 browsers.

Note: You cannot attach the Drag Layer action to an object with the `onMouseDown` or `onClick` events.

Gathering information about the draggable layer

When you attach the Drag Layer action to an object, Dreamweaver inserts the `MM_dragLayer()` function into the head section of your document. In addition to registering the layer as draggable, this function defines three properties for each draggable layer—`MM_LEFTRIGHT`, `MM_UPDOWN`, and `MM_SNAPPED`—that you can use in your own JavaScript functions to determine the relative horizontal position of the layer, the relative vertical position of the layer, and whether the layer has reached the drop target.

Note: The information provided here is intended for the use of experienced JavaScript programmers only.

For example, the following function displays the value of the `MM_UPDOWN` property (the current vertical position of the layer) in a form field called `curPosField`. (Form fields are useful for displaying continuously updated information because they are dynamic—that is, you can change their contents after the page has finished loading—in both Navigator and Internet Explorer.)

```
function getPos(layername){
    var layerRef = MM_findObj(layername);
    var curVertPos = layerRef.MM_UPDOWN;
    document.tracking.curPosField.value = curVertPos;
}
```

Instead of displaying the value of `MM_UPDOWN` or `MM_LEFTRIGHT` in a form field, you could write a function that displays a message in the form field depending on how close the value is to the drop zone, or you could call another function to show or hide a layer depending on the value. How you react to the value of `MM_UPDOWN` or `MM_LEFTRIGHT` is limited only by your imagination and your JavaScript skills.

It is especially useful to read the `MM_SNAPPED` property when you have several layers on the page, all of which must reach their targets before the visitor can advance to the next page or task. For example, you could write a function to count how many layers have an `MM_SNAPPED` value of `true` and call it whenever a layer is dropped. When the snapped count reaches the desired number, you could send the visitor to the next page or display a message of congratulations.

If you have used the `onMouseOver` event to attach the Drag Layer action to links within several layers, you must make a minor change to the `MM_dragLayer()` function to prevent the `MM_SNAPPED` property of a snapped layer from being reset to `false` if the mouse pointer rolls over the layer. (This can happen if you have used Drag Layer to create a picture puzzle, because the visitor is likely to roll the mouse pointer over snapped pieces while positioning others.) The `MM_dragLayer()` function does not prevent this behavior, because it is sometimes desirable—for example, if you want to set multiple drop targets for a single layer.

To prevent re-registration of snapped layers:

- 1 Make a backup copy of your document before making any changes to the code. (You can do this in the Site panel in Dreamweaver, or in Windows Explorer (Windows) or the Finder (Macintosh).)
- 2 Choose Edit > Find.
- 3 Choose HTML Source from the Find What pop-up menu.
- 4 Type `(!curDrag)` in the adjacent text field.
- 5 Click Find Next.

If Dreamweaver asks if you want to continue searching from the beginning of the document, click Yes. Dreamweaver finds a statement that reads:

```
if (!curDrag) return false;
```

- 6 Close the Find dialog box and then modify the statement in the Document window's Code view or in the Code inspector so that it reads:

```
if (!curDrag || curDrag.MM_SNAPPED != null) return false;
```

The two pipes (`||`) mean “or,” and `curDrag` is a variable that represents the layer that is being registered as draggable. In English the statement means “If `curDrag` is not an object, or if it already has an `MM_SNAPPED` value, don't bother executing the rest of the function.”

Go to URL

The Go to URL action opens a new page in the current window or in the specified frame. This action is particularly useful for changing the contents of two or more frames with one click. It can also be called in a timeline to jump to a new page after a specified time interval.

To use the Go To URL action:

- 1 Select an object and open the Behaviors panel.
- 2 Click the plus (+) button and choose Go to URL from the Actions pop-up menu.
- 3 Choose a destination for the URL from the Open In list.

The Open In list automatically lists the names of all frames in the current frameset as well as the main window. If there are no frames, the main window is the only option.

Note: This action may produce unexpected results if any frame is named top, blank, self, or parent. Browsers sometimes mistake these names for reserved target names.

- 4 Click Browse to select a document to open, or enter the path and filename of the document in the URL field.
- 5 Repeat steps 3 and 4 to open additional documents in other frames.
- 6 Click OK.
- 7 Check that the default event is the one you want.

If it isn't, choose another event from the pop-up menu. If the events you want are not listed, change the target browser in the Show Events For pop-up menu.

Jump Menu

When you create a jump menu by using Insert > Form Objects > Jump Menu, Dreamweaver creates a menu object and attaches the Jump Menu (or Jump Menu Go) behavior to it. There is usually no need to attach the Jump Menu action to an object by hand. For information about jump menus and how to create them, see "Inserting a jump menu" on page 413.

You can edit an existing jump menu in either of two ways:

- You can edit and rearrange menu items, change the files to jump to, and change the window in which those files open, by double-clicking an existing Jump Menu action in the Behaviors panel.
- You can edit the items in the menu just as you would edit items in any menu, by selecting the menu and using the List Values button in the Property inspector. See "Creating pop-up menus" on page 588 for details.

To edit a jump menu using the Behaviors panel:

- 1 Create a jump menu object if there isn't one already in your document.
- 2 Select the jump menu object and open the Behaviors panel.
- 3 Double-click Jump Menu in the Actions column.
- 4 Make changes as desired in the Jump Menu dialog box, then click OK.

Jump Menu Go

The Jump Menu Go action is closely associated with the Jump Menu action; Jump Menu Go lets you associate a Go button with a jump menu. (Before you use this action, a jump menu must already exist in the document.) Clicking the Go button opens the link that's selected in the jump menu. A jump menu doesn't normally need a Go button; choosing an item from a jump menu generally causes a URL to load without any need for further user action. But if the visitor chooses the same item that's already chosen in the jump menu, the jump doesn't occur. In general, that doesn't matter, but if the jump menu appears in a frame, and the jump menu items link to pages in other frames, a Go button is often useful, to allow visitors to re-choose an item that's already selected in the jump menu.

To add a Jump Menu Go action:

- 1 Select an object to use as the Go button (generally a button image), and open the Behaviors panel.
- 2 Click the plus (+) button and choose Jump Menu Go from the Actions pop-up menu.
- 3 In the Choose Jump Menu pop-up menu, choose a menu for the Go button to activate.
- 4 Click OK.

Open Browser Window

Use the Open Browser Window action to open a URL in a new window. You can specify the properties of the new window, including its size, attributes (whether it is resizable, has a menu bar, and so on), and name. For example, you can use this behavior to open a larger image in a separate window when the visitor clicks a thumbnail image; with this behavior, you can make the new window the exact size of the image.

If you specify no attributes for the window, it opens at the size and with the attributes of the window that launched it. Specifying any attribute for the window automatically turns off all other attributes that are not explicitly turned on. For example, if you set no attributes for the window, it might open at 640 x 480 pixels and have a navigation bar, location toolbar, status bar, and menu bar. If you explicitly set the width to 640 and the height to 480 and set no other attributes, the window opens at 640 x 480 pixels and has no navigation bar, no location toolbar, no status bar, no menu bar, no resize handles, and no scroll bars.

To use the Open Browser Window action:

- 1 Select an object and open the Behaviors panel.
- 2 Click the plus (+) button and choose Open Browser Window from the Actions pop-up menu.
- 3 Click Browse to select a file, or enter the URL you want to display.
- 4 Set any of the following options:

Window Width specifies the width of the window in pixels.

Window Height specifies the height of the window in pixels.

Navigation Toolbar is the row of browser buttons that includes Back, Forward, Home, and Reload.

Location Toolbar is the row of browser options that includes the location field.

Status Bar is the area at the bottom of the browser window in which messages (such as the load time remaining and the URLs associated with links) appear.

Menu Bar is the area of the browser window (Windows) or the desktop (Macintosh) where menus such as File, Edit, View, Go, and Help appear. You should explicitly set this option if you want visitors to be able to navigate from the new window. If you do not set this option, the user can only close or minimize the window (Windows) or close the window or quit the application (Macintosh) from the new window.

Scrollbars as Needed specifies that scroll bars should appear if the content extends beyond the visible area. If you do not explicitly set this option, scroll bars do not appear. If the Resize Handles option is also turned off, visitors have no easy way of seeing content that extends beyond the original size of the window. (Though they may be able to make the window scroll by dragging off the edge of the window.)

Resize Handles specifies that the user should be able to resize the window, either by dragging the lower right corner of the window or by clicking the maximize button (Windows) or size box (Macintosh) in the upper right corner. If this option is not explicitly set, the resize controls are unavailable and the lower right corner is not draggable.

Window Name is the name of the new window. You should name the new window if you want to target it with links or control it with JavaScript. This name cannot contain spaces or special characters.

5 Click OK.

6 Check that the default event is the one you want.

If it isn't, choose another event from the pop-up menu. If the events you want are not listed, change the target browser in the Show Events For pop-up menu.

Play Sound

Use the Play Sound action to play a sound. For example, you might want to play a sound effect whenever the mouse pointer rolls over a link, or you might want to play a music clip when the page loads.

Note: Browsers may require some kind of additional audio support (such as an audio plug-in) to play sounds. Therefore, different browsers with different plug-ins often play sounds differently. It's difficult to reliably predict how visitors to your site will experience the sounds you provide.

To use the Play Sound action:

1 Select an object and open the Behaviors panel.

2 Click the plus (+) button and choose Play Sound from the Actions pop-up menu.

3 Click Browse to select a sound file, or enter the path and filename in the Play Sound field.

4 Click OK.

5 Check that the default event is the one you want.

If it isn't, choose another event from the pop-up menu. If the events you want are not listed, change the target browser in the Show Events For pop-up menu.

Popup Message

The Popup Message action displays a JavaScript alert with the message you specify. Because JavaScript alerts have only one button (OK), use this action to provide information rather than to present the user with a choice.

You can embed any valid JavaScript function call, property, global variable, or other expression in the text. To embed a JavaScript expression, place it inside braces ({}). To display a brace, precede it with a backslash (\).

Example

The URL for this page is {window.location}, and today is {new Date()}.

Note: You can't control how the JavaScript alert looks; that's determined by the visitor's browser. If you want more control over the appearance of your message, consider using the Open Browser Window behavior. For details, see "Open Browser Window" on page 362.

To use the Popup Message action:

- 1 Select an object and open the Behaviors panel.
- 2 Click the plus (+) button and choose Popup Message from the Actions pop-up menu.
- 3 Enter your message in the Message field.
- 4 Click OK.
- 5 Check that the default event is the one you want.

If it isn't, choose another event from the pop-up menu. If the events you want are not listed, change the target browser in the Show Events For pop-up menu.

Preload Images

The Preload Images action loads images that do not appear on the page right away (such as those that will be swapped in with timelines, behaviors, or JavaScript) into the browser cache. This prevents delays caused by downloading when it is time for the images to appear.

Note: The Swap Image action automatically preloads all highlight images when you select the Preload Images option in the Swap Image dialog box, so you do not need to manually add Preload Images when using Swap Image.

To use the Preload Images action:

- 1 Select an object and open the Behaviors panel.
- 2 Click the plus (+) button and choose Preload Images from the Actions pop-up menu.
- 3 Click Browse to select an image file to preload, or enter the path and filename of an image in the Image Source File field.
- 4 Click the plus (+) button at the top of the dialog box to add the image to the Preload Images list.

Note: If you do not click the plus button before entering the next image, the image you have just chosen will be replaced in the list with the image you choose next.

- 5 Repeat steps 3 and 4 for all remaining images that you want to preload on the current page.
- 6 To remove an image from the Preload Images list, select the image in the list and click the minus (-) button.
- 7 Click OK.

- 8 Check that the default event is the one you want.

If it isn't, choose another event from the pop-up menu. If the events you want are not listed, change the target browser in the Show Events For pop-up menu.

Set Nav Bar Image

Use the Set Nav Bar Image action to turn an image into a navigation bar image, or to change the display and actions of images in a navigation bar. (For more information, see “Inserting a navigation bar” on page 415.)

Use the Basic tab of the Set Nav Bar Image dialog box to create or update a navigation bar image or set of images, to change which URL is displayed when a navigation-bar button is clicked, and to select a different window in which to display a URL.

Use the Advanced tab of the Set Nav Bar Image dialog box to change the state of other images in a document based on the current button's state. By default, clicking an element in a navigation bar automatically causes all other elements in the navigation bar to return to their Up states; use the Advanced tab if you want to set a different state for an image when the selected image is in its Down or Over state.

To edit a Set Nav Bar Image action:

- 1 Select an image in the navigation bar to edit, and open the Behaviors panel.
- 2 In the Behaviors panel, in the Actions column, double-click the Set Nav Bar Image action associated with the event you're altering.
- 3 In the Basic tab of the Set Nav Bar Image dialog box, select image edit options.

To set multiple images for a navigation bar button:

- 1 Select an image in the navigation bar to edit, and open the Behaviors panel.
- 2 In the Behaviors panel, in the Actions column, double-click the Set Nav Bar Image action associated with the event you're altering.
- 3 Click the Advanced tab of the Set Nav Bar Image dialog box.
- 4 In the When Element Is Displaying pop-up menu, choose an image state. For information about image states, see “Creating navigation bars” on page 414.
 - Choose Down Image if you want to change the display of another image after a user has clicked the selected image.
 - Choose Over Image or Over While Down Image if you want to change the display of another image when the pointer is over the selected image.
- 5 In the Also Set Image list, select another image on the page to set.
- 6 Click Browse to select the image file to display, or type the path of the image file in the To Image File field.
- 7 If you selected Over Image or Over While Down Image in step 4, you have an additional option. In the If Down, To Image File text field, click Browse to select the image file, or type the path to the image file to display.

Set Text of Frame

The Set Text of Frame action allows you to dynamically set the text of a frame, replacing the content and formatting of a frame with the content you specify. The content can include any valid HTML code. Use this action to dynamically display information.

Although the Set Text of Frame action replaces the formatting of a frame, you can select Preserve Background Color to preserve the page background and text color attributes.

You can embed any valid JavaScript function call, property, global variable, or other expression in the text. To embed a JavaScript expression, place it inside braces ({}). To display a brace, precede it with a backslash (\).

Example

The URL for this page is {window.location}, and today is {new Date()}.

To create a frameset:

Choose Modify > Frameset > Split Frame Left, Right, Up, or Down.

For more information, see “Creating frames and framesets” on page 259.

To use the Set Text of Frame action:

- 1 Select an object and open the Behaviors panel.
- 2 Click the plus (+) button and choose Set Text > Set Text of Frame from the Actions pop-up menu.
- 3 In the Set Text of Frame dialog box, choose the target frame from the Frame pop-up menu.
- 4 Click the Get Current HTML button to copy the current contents of the target frame’s body section.
- 5 Enter a message in the New HTML field, then click OK.
- 6 Check that the default event is the one you want. If it isn’t, choose another event from the pop-up menu.
If you don’t see the events you want, change the target browser in the Show Events For pop-up menu.

Set Text of Layer

The Set Text of Layer action replaces the content and formatting of an existing layer on a page with the content you specify. The content can include any valid HTML source code.

Set Text of Layer replaces the content and formatting of the layer, but retains layer attributes, including color. Format the content by including HTML tags in the New HTML field of the Set Text of Layer dialog box.

You can embed any valid JavaScript function call, property, global variable, or other expression in the text. To embed a JavaScript expression, place it inside braces ({}). To display a brace, precede it with a backslash (\).

Example

The URL for this page is {window.location}, and today is {new Date()}.

To create a layer:

- 1 Choose Insert > Layer.

For more information, see “Creating layers on your page” on page 378.

- 2 In the Property inspector, type a name for the layer.

To attach a Set Text of Layer action:

- 1 Select an object and open the Behaviors panel.
- 2 Click the plus (+) button and choose Set Text > Set Text of Layer from the Actions pop-up menu.
- 3 In the Set Text of Layer dialog box, use the Layer pop-up menu to choose the target layer.
- 4 Enter a message in the New HTML field, then click OK.
- 5 Check that the default event is the one you want. If it isn't, choose another event from the pop-up menu.

If you don't see the events you want, change the target browser in the Show Events For pop-up menu.

Set Text of Status Bar

The Set Text of Status Bar action shows a message in the status bar at the bottom left of the browser window. For example, you can use this action to describe the destination of a link in the status bar instead of showing the URL associated with it. To see an example of a status message, roll your mouse over any of the navigation buttons in Dreamweaver Help. Note, however, that visitors often ignore or overlook messages in the status bar (and not all browsers provide full support for setting the text of the status bar); if your message is important, consider displaying it as a pop-up message or as the text of a layer.

You can embed any valid JavaScript function call, property, global variable, or other expression in the text. To embed a JavaScript expression, place it inside braces ({}). To display a brace, precede it with a backslash (\{).

Example

The URL for this page is {window.location}, and today is {new Date()}.

To use the Set Text of Status Bar action:

- 1 Select an object and open the Behaviors panel.
- 2 Click the plus (+) button and choose Set Text > Set Text of Status Bar from the Actions pop-up menu.
- 3 In the Set Text of Status Bar dialog box, type your message in the Message field.
Keep the message concise. The browser truncates the message if it doesn't fit in the status bar.
- 4 Click OK.
- 5 Check that the default event is the one you want.

If it isn't, choose another event from the pop-up menu. If the events you want are not listed, change the target browser in the Show Events For pop-up menu.

Set Text of Text Field

The Set Text of Text Field action replaces the content of a form's text field with the content you specify. You can embed any valid JavaScript function call, property, global variable, or other expression in the text. To embed a JavaScript expression, place it inside braces ({}). To display a brace, precede it with a backslash (\{).

Example

The URL for this page is {window.location}, and today is {new Date()}.

To create a named text field:

- 1 Choose Insert > Form Objects > Text Field.
If Dreamweaver prompts you to add a form tag, click Yes. For more information, see “Creating Interactive Forms” on page 573.
- 2 In the Property inspector, type a name for the text field. Make sure the name is unique on the page (don't use the same name for multiple elements on the same page, even if they're in different forms).

To use the Set Text of Text Field action:

- 1 Select a text field and open the Behaviors panel.
- 2 Click the plus (+) button and choose Set Text > Set Text of Text Field from the Actions pop-up menu.
- 3 In the Set Text of Text Field dialog box, choose the target text field from the Text Field pop-up menu.
- 4 Enter text in the New Text field, then click OK.
- 5 Check that the default event is the one you want. If it isn't, choose another event from the pop-up menu.

If you don't see the events you want, change the target browser in the Show Events For pop-up menu.

Show-Hide Layers

The Show-Hide Layers action shows, hides, or restores the default visibility of one or more layers. This action is useful for showing information as the user interacts with the page. For example, as the user rolls the mouse pointer over an image of a plant, you could show a layer that gives details about the plant's growing season and region, how much sun it needs, how large it can grow, and so on.

Show-Hide Layers is also useful for creating a preload layer—that is, a large layer that obscures the contents of the page at first and then disappears when all the page components have finished loading.

To use the Show-Hide Layers action:

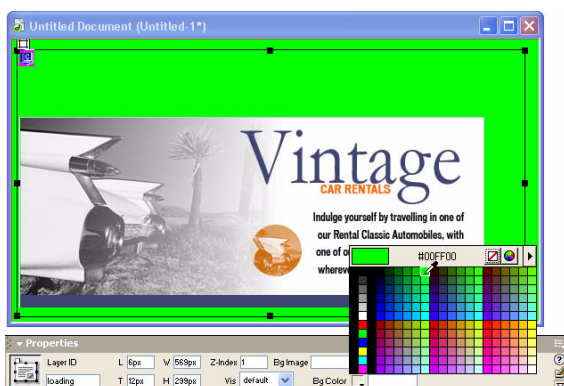
- 1 Choose Insert > Layer or click the Layer button in the Insert bar, and draw a layer in the Document window.
Repeat this step to create additional layers.
- 2 Click in the Document window to deselect the layer, then open the Behavior panel.

- 3 Click the plus (+) button and choose Show-Hide Layers from the Actions pop-up menu.
If Show-Hide Layers is unavailable, you probably have a layer selected. Because layers do not accept events in both 4.0 browsers, you must select a different object—such as the `body` tag or a link (`a` tag)—or change the target browser to IE 4.0 in the Show Events For pop-up menu.
- 4 From the Named Layers list, select the layer whose visibility you want to change.
- 5 Click Show to show the layer, Hide to hide the layer, or Default to restore the layer's default visibility.
- 6 Repeat steps 4 and 5 for all remaining layers whose visibility you want to change at this time. (You can change the visibility of multiple layers with a single behavior.)
- 7 Click OK.
- 8 Check that the default event is the one you want.
If it isn't, choose another event from the pop-up menu. If the events you want are not listed, change the target browser in the Show Events For pop-up menu.

When viewed in a Netscape Navigator browser window, layers may shrink to fit the content. To keep this from happening, add text or images to layers, or set layer clip values.

To create a preload layer:

- 1 Click the Draw Layer button in the Common category of the Insert bar and draw a large layer in the Document window's Design view.
Be sure the layer covers all the content on the page.
- 2 In the Layers panel, drag the layer name to the top of the list of layers to specify that the layer should be at the front of the stacking order.
- 3 Select the layer if it's not selected, and name it *loading*, using the leftmost field in the layer Property inspector.
- 4 With the layer still selected, set the background color of the layer to the same color as the page background in the Property inspector.



- 5 Click inside the layer (which should now be obscuring the rest of the page contents) and type a message, if desired.
For example, “Please wait while the page loads” or “Loading...” are messages that tell visitors what is happening so that they know the page contains content.
- 6 Click the `<body>` tag in the tag selector in the bottom left corner of the Document window.
- 7 In the Behaviors panel, choose Show-Hide Layers from the Actions pop-up menu.
- 8 Select the layer called `loading` from the Named Layers list.
- 9 Click Hide.
- 10 Click OK.
- 11 Make sure that the event listed next to the Show-Hide Layers action in the behaviors list is `onLoad`. (If it isn't, select the event and click the downward-pointing triangle that appears between the event and the action. Choose `onLoad` from the list of events in the pop-up menu.)

Show Pop-Up Menu

You use the Show Pop-Up Menu behavior to create or edit a Dreamweaver pop-up menu or to open and modify a Fireworks pop-up menu you've inserted in a Dreamweaver document.

You set options in the Show Pop-Up Menu dialog box to create a horizontal or vertical pop-up menu. You can use this dialog box to set or modify the color, text, and position of a pop-up menu.

Note: You must use the Edit button in the Dreamweaver Property inspector to edit images in a Fireworks image-based pop-up menu. You can, however, use the Show Pop-Up Menu command to change the text in an image-based pop-up menu. For information about editing images in Fireworks MX see, Chapter 21, “Dreamweaver Integration with Other Applications,” on page 305.

To view a pop-up menu in a document, you must open the document in a browser window, then roll the pointer over the triggering image or link.

To use the Show Pop-up Menu action:

- 1 Select an object to attach the behavior to and open the Behaviors panel (Shift+F3).
- 2 Click the plus (+) button and choose Show Pop-Up Menu from the Actions pop-up menu.
- 3 In the Show Pop-Up Menu dialog box that appears, use the following tabs to set options for the pop-up menu:

Contents allows you to set the name, structure, URL, and target of individual menu items.

Appearance enables you to set the appearance of the menu's Up State and Over State and to set font choices for menu item text.

Advanced allows you to set the properties of the menu cells. For example, you can set cell width and height, cell color and border width, text indentation, and the length of delay before the menu appears after the user moves the pointer over the trigger.

Position lets you set where the menu is positioned relative to the triggering image or link.

Adding, removing, and changing the order of pop-up menu items

You use the Contents tab in the Show Pop-up Menu dialog box to create menu items. You can also use this tab to remove existing items, or to change the order in which they appear in a menu.

To add pop-up menu items:

- 1 In the Contents tab, create a pop-up menu item by doing the following:

In the Text field, select the default text (New Item), then type the text you want to appear in the pop-up menu.

- 2 Set additional options, as desired:

If you want the menu item to open another file when clicked, in the Link field, type the file path or click the Folder icon and browse to the document you want to open.

If you want to set a location in which the document opens, for example in a new window or in a specific frame, in the Target pop-up menu choose the desired location.

Note: If the frame you want to target doesn't appear on the Target pop-up menu, close the Show Pop-Up Menu dialog box, then in the Document window select and name the frame.

- 3 Click the plus (+) button to add additional entries to the Show Pop-Up Menu preview list.

When you finish adding menu items, click OK to accept the default settings or select another Show Pop-Up Menu tab to set additional options.

To create a submenu item:

In the Show Pop-Up Menu list, select the item you want to make into a submenu item, then do one of the following:

- To indent an item in the menu list, click the Indent Item button.
- To remove an indent, click the Outdent Item button.

Note: You cannot make the first menu item in a list a submenu item.

To change the order of an item in the menu:

In the Show Pop-Up Menu list, select the item you want to move up or down, then click the Up or Down arrow to move the item where you want it to appear.

To remove an item from the menu:

- 1 In the Contents tab, select the menu entry you want to remove in the Show Pop-Up Menu list.
- 2 Click the minus (-) button.

Setting the appearance of a pop-up menu

After you create the menu items, use the Show Pop-Up Menu's Appearance tab to set the orientation, font attributes, and button state attributes for the pop-up menu.

Note: The Appearance tab's preview pane provides an approximate rendering of the options you set in this tab.

To set the appearance of a pop-up menu:

- 1 In the pop-up menu at the top of the Appearance tab, choose Vertical Menu or Horizontal Menu to set the menu's orientation.
- 2 Set the text formatting options you want:

In the Font pop-up menu, select the font you want to apply to the menu items.

Note: If the font you want to apply is not in the font list, use the Edit Font List option, to add the desired font to the font list. To ensure the menu appears as desired, you should choose a font that site visitor's are likely to have.

Set the font size, style attributes, and text alignment or justification options for the menu item text.

- 3 In the Up State and Over States boxes, use the color picker to set the text and cell colors of the menu item buttons.
- 4 When you finish setting appearance options, click OK or select another Show Pop-Up Menu tab to set additional options.

Setting advanced appearance options

Use options in the Advanced tab to specify additional attributes of the menu cells. For example, you can set the width, height, cell spacing or padding of the menu button, indent text, and set border attributes.

To set advanced formatting attributes for a pop-up menu:

- 1 Click the Advanced tab, then set the options you want to apply to the menu items:

Cell Width sets a specific width, in pixels, for the menu buttons. Cell width is set automatically based on the widest item; to increase the cell width, select Pixels in the pop-up menu and enter a value larger than the one that appears in the Cell Width text box.

Cell Height sets a specific height, in pixels, for the menu buttons. To increase the cell height, select Pixels in the pop-up menu and enter a value larger than the one that appears in the Cell Height text box.

Cell Padding specifies the number of pixels between a cell's content and its boundaries.

Cell Spacing specifies the number of pixels between adjacent cells.

Text Indent allows you to specify, in pixels, how far text in a menu item is indented within the cell.

Menu Delay sets the length of time between when the user moves the pointer over the triggering image or link, and when the menu appears. Values are in milliseconds so the default setting, 1000, equals 1 second. For every second of delay you want, add 000; for example, for a 3 second delay, type 3000.

Pop-up Borders determines whether a border appears around the items in the menu. If you want a border to appear around the menu items, make sure the Show Borders checkbox is checked.

Border Width sets the border's width, in pixels.

Shadow, Border Color, and Highlight allow you to pick a color for these border options. Shadow and highlight are not reflected in the preview.

- 2 When you finish setting advanced appearance options, click OK or select another Show Pop-Up Menu tab to set additional options.

Setting a pop-up menu's position in a document

Use position options to set where the pop-up menu displays relative to the triggering image or link. You can also set whether the menu hides or not when the user moves the pointer away from the trigger.

To set pop-up menu position options:

- 1 In the Show Pop-Up Menu dialog box, click the Position tab.
- 2 Set the location of the pop-up menu by doing one of the following:
 - Choose one of the preset options.
 - Set customized position coordinates by typing a number in the X text box to set the horizontal coordinate and by typing a number in the Y text box to set the vertical coordinate. Coordinates count from the top left corner of the menu.
- 3 To hide the pop-up menu when the pointer is not over it, make sure Hide Menu onMouseOut Event is checked. To leave the menu displayed, deselect this option.
- 4 When you finish creating or modifying the pop-up menu, click OK.

Modifying a pop-up menu

The Show Pop-Up Menu behavior allows you to edit or update the contents of a pop-up menu. You can add, delete, or change menu items, rearrange them, and set where a menu is positioned relative to the triggering image or link.

To open an existing HTML-based pop-up menu:

- 1 In the Dreamweaver document, select the link or image that triggers the pop-up menu.
- 2 Open the Behaviors panel (Shift + F3), if it isn't already open, then in the Actions list, double-click Show Pop-Up Menu.

The Show Pop-Up Menu dialog box appears.

- 3 Make the changes you want to make to the pop-up menu.
- 4 When you finish modifying the pop-up menu, click OK.

For detailed information about setting pop-up menu options, see “Show Pop-Up Menu” on page 370.

Swap Image

The Swap Image action swaps one image for another by changing the `src` attribute of the `img` tag. Use this action to create button rollovers and other image effects (including swapping more than one image at a time). Inserting a rollover image automatically adds a Swap Image behavior to your page.

Note: Because only the `src` attribute is affected by this action, you should swap in an image that has the same dimensions (height and width) as the original. Otherwise, the image you swap in appears compacted or expanded to fit the original image's dimensions.

To use the Swap Image action:

- 1 Choose Insert > Image or click the Image button on the Insert bar to insert an image.
- 2 In the Property inspector, enter a name for the image in the leftmost text box.

The Swap Image action still works if you do not name your images; it names unnamed images automatically when you attach the behavior to an object. However, it is easier to distinguish images in the Swap Image dialog box if all of the images are named beforehand.

- 3 Repeat steps 1 and 2 to insert additional images.
- 4 Select an object (generally the image you're going to swap) and open the Behaviors panel.
- 5 Click the plus (+) button and choose Swap Image from the Actions pop-up menu.
- 6 From the Images list, select the image whose source you want to change.
- 7 Click Browse to select the new image file, or enter the path and filename of the new image in the Set Source To text box.
- 8 Repeat steps 6 and 7 for any additional images you want to change. Use the same Swap Image action for all the images you want to change at once; otherwise, the corresponding Swap Image Restore action won't restore all of them.
- 9 Select the Preload Images option to load the new images into the browser's cache when the page is loaded.
This prevents delays caused by downloading when it is time for the images to appear.
- 10 Click OK.
- 11 Check that the default event is the one you want.
If it isn't, choose another event from the pop-up menu. If the events you want are not listed, change the target browser in the Show Events For pop-up menu.

Swap Image Restore

The Swap Image Restore action restores the last set of swapped images to their previous source files. This action is automatically added whenever you attach the Swap Image action to an object; if you left the Restore option selected while attaching Swap Image, you should never need to select the Swap Image Restore action manually.

Go to Timeline Frame

The Go to Timeline Frame action moves the playback head to the specified frame. You can use this action in the Behavior channel of the Timelines panel to make portions of the timeline loop a specific number of times, to create a Rewind link or button, or to let the user jump to different parts of the animation.

To use the Go To Timeline Frame action:

- 1 Choose Window > Timeline to open the Timelines panel, and make sure that your document contains a timeline.
If you don't see any purple animation bars in the Timelines panel, your document does not contain a timeline. See "Moving a layer using a timeline animation" on page 390.
- 2 Select an object to attach the behavior to.
To attach the behavior to a frame in the Timeline, click in the Behavior channel at the desired frame.
- 3 Open the Behaviors panel.
- 4 Click the plus (+) button and choose Timeline > Go to Timeline Frame from the Actions pop-up menu. (If this action is dimmed, your document doesn't contain a timeline.)
- 5 Choose a timeline from the Timeline pop-up menu.

- 6 Enter a frame number in the Go to Frame text box.
- 7 If you are adding this action in the Behavior channel of a timeline and want a portion of the timeline to loop, enter the number of times the segment should loop in the Loop text box.
You must leave this text box blank if you are not attaching Go to Timeline Frame to a frame in a timeline.
- 8 Click OK.
- 9 Check that the default event is the one you want.
If it isn't, choose another event from the pop-up menu. If the events you want are not listed, change the target browser in the Show Events For pop-up menu.

Play Timeline and Stop Timeline

Use the Play Timeline and Stop Timeline actions to let visitors start and stop a timeline by clicking a link or button, or to start and stop a timeline automatically when the user rolls over a link, image, or other object. The Play Timeline action is automatically attached to the `body` tag with the `onLoad` event when you select Autoplay in the Timelines panel.

To use the Play Timeline and Stop Timeline actions:

- 1 Choose Window > Timeline to open the Timelines panel, and make sure that your document contains a timeline.
If you don't see any purple animation bars in the Timelines panel, your document does not contain a timeline. See "Moving a layer using a timeline animation" on page 390.
- 2 Select an object and open the Behaviors panel.
- 3 Click the plus (+) button and choose Play Timeline or Stop Timeline from the Actions pop-up menu.
- 4 Select the timeline you want to play or stop, or choose to stop all timelines, from the pop-up menu.
- 5 Click OK.
- 6 Check that the default event is the one you want.
If it isn't, choose another event from the pop-up menu. If the events you want are not listed, change the target browser in the Show Events For pop-up menu.

Validate Form

The Validate Form action checks the contents of specified text fields to ensure that the user has entered the correct type of data. Attach this action to individual text fields with the `onBlur` event to validate the fields as the user is filling out the form, or attach it to the form with the `onSubmit` event to evaluate several text fields at once when the user clicks the Submit button. Attaching this action to a form prevents the form from being submitted to the server if any of the specified text fields contains invalid data.

To use the Validate Form action:

- 1 Choose Insert > Form or click the Form button in the Insert bar to insert a form.
- 2 Choose Insert > Form Objects > Text Field or click the Text Field button in the Insert bar to insert a text field.
Repeat this step to insert additional text fields.
- 3 Do one of the following:
 - To validate individual fields as the user fills out the form, select a text field and choose Window > Behaviors.
 - To validate multiple fields when the user submits the form, click the `<form>` tag in the tag selector in the bottom left corner of the Document window and choose Window > Behaviors.
- 4 Choose Validate Form from the Actions pop-up menu.
- 5 Do one of the following:
 - If you are validating individual fields, select the same field that you have selected in the Document window from the Named Fields list.
 - If you are validating multiple fields, select a text field from the Named Fields list.
- 6 Select the Required option if the field must contain some data.
- 7 Choose from one of the following Accept options:
 - Use Anything if the field is required but need not contain any particular kind of data. (If the Required option is not selected, the Anything option is meaningless—that is, it is the same as if the Validate Form action were not attached to the field.)
 - Use E-mail address to check that the field contains an @ symbol.
 - Use Number to check that the field contains only numerals.
 - Use Number From to check that the field contains a number in a specific range.
- 8 If you are validating multiple fields, repeat steps 6 and 7 for any additional fields that you want to validate.
- 9 Click OK.
If you are validating multiple fields when the user submits the form, the `onSubmit` event automatically appears in the Events pop-up menu.
- 10 If you are validating individual fields, check that the default event is `onBlur` or `onChange`.
If it isn't, select `onBlur` or `onChange` from the pop-up menu. Both of these events trigger the Validate Form action when the user moves away from the field. The difference between them is that `onBlur` occurs whether or not the user has typed in the field, and `onChange` occurs only if the user changed the contents of the field. The `onBlur` event is preferred when you have specified that the field is required.

CHAPTER 25

Animating Layers

A layer is an HTML page element that you can position anywhere on your page. Layers can contain text, images, or any other content that you can place in the body of an HTML document.

With Dreamweaver, you can place and animate layers without having to do any JavaScript or HTML coding. You can place layers in front of and behind each other, hide some layers while showing others, and move layers across the screen. You can place a background image in one layer, then place a second layer, containing text with a transparent background, in front of that. And you can animate layers fading in and out.

Layers provide a great deal of flexibility in placing content. However, web browsers older than Microsoft Internet Explorer 4.0 and Netscape Navigator 4.0 can't display layers, and the version 4 browsers aren't completely consistent in how they display layers. To ensure that everyone can view your web page, you can design your page layout using layers, then convert the layers to tables. For more information, see "Using tables and layers for layout" on page 386. If your audience is likely to be using more recent browsers, however, you can design layouts entirely with layers, without converting them to tables.

Note: If you're unfamiliar with using layers and CSS, we recommend that you use tables or Layout view for page layout (see "Presenting Content with Tables" on page 227 and "Laying Out Pages in Layout View" on page 241). Layout view is an easy way to implement your page layout with tables as the underlying structure.

This chapter contains the following sections:

- "About HTML code for layers" on page 378
- "Creating layers on your page" on page 378
- "Nesting layers" on page 380
- "Manipulating layers" on page 381
- "Adding content to layers" on page 383
- "Viewing and setting layer properties" on page 384
- "Using tables and layers for layout" on page 386
- "Animating your layers" on page 388
- "Animating layers using behavior actions" on page 395

About HTML code for layers

When you place a layer in a document, Dreamweaver inserts the HTML tag for that layer in your code. You can choose to have Dreamweaver use either the `div` tag or the `span` tag for your layers. By default, Dreamweaver creates layers using the `div` tag.

Note: There are two other tags that you can use to create layers: `layer` and `ilayer`. However, these tags are supported only in Netscape Navigator 4; Internet Explorer does not support these tags, and Netscape discontinued support for these tags in later browsers. Dreamweaver recognizes the `layer` and `ilayer` tags, but does not create layers using these tags.

To change the default tag, see “Setting Layers preferences” on page 380.

The difference between the `div` and `span` tags is that browsers that don’t support layers place extra line breaks before and after the `div` tag; that is, the `div` tag is a block-level element, while the `span` tag is an inline element. In most cases, it’s better for layer content to appear in a paragraph of its own in browsers that don’t support layers, so in most cases it’s better to use `div` than `span`. To further improve readability in older browsers, be careful where you place the code for a layer.

The code that defines a layer can be anywhere in the body of an HTML file. When you draw a layer in Dreamweaver, the layer is displayed where you drew it, but Dreamweaver inserts the layer’s code at the beginning of the page, just after the `body` tag. (If you use the Insert Layer command instead of drawing a layer, the layer code is inserted at the insertion point.) If you create a nested layer, Dreamweaver inserts the code inside the tag that defines the parent layer.

Note: Regardless of which tag you use, versions of both Internet Explorer and Netscape Navigator prior to 4.0 display the contents of a layer, but do not position the layer. The layer’s contents appear at the point in the page where the layer’s code is; for example, if the layer’s code is at the beginning of the page, the layer’s contents will appear at the beginning of the page in browsers that don’t support layers.

The following is sample HTML code for a layer:

```
<div id="Layer1" style="position:absolute; visibility:inherit; width:200px;
  height:115px; z-index:1">
</div>
```

The following is sample HTML code for a layer nested inside another layer:

```
<div id="Parent" style="position:absolute; left:56px; top:54px; width:124px;
  height:158px; z-index:1;">
Content inside the parent layer.
<div id="Nested" style="position:absolute; left:97px; top:114px; width:54px;
  height:69px; z-index:1;">
  Content inside the nested layer.
</div>
</div>
```

You can set properties for layers on your page, including *x* and *y* coordinates, *z*-index (also called the stacking order), and visibility. For more information, see “Viewing and setting layer properties” on page 384.

Creating layers on your page

Dreamweaver lets you create layers on your page easily and position them precisely.

To create a layer, do one of the following:

- To draw a layer, click the Draw Layer button in the Insert bar, then drag in the Document window’s Design view to draw the layer.

- To insert a layer's code at a particular place in the document, place the insertion point in the Document window and then choose Insert > Layer.

If you have Invisible Elements showing, a layer-code marker appears in the Design view each time you place a layer on the page. If layer-code markers aren't visible and you want to see them, choose View > Visual Aids > Invisible Elements.

Note: When the Invisible Elements option is turned on, the elements on your page may appear to shift position. However, invisible elements don't appear in browsers, so when you view your page in a browser, all the visible elements appear in the correct positions.

To draw multiple layers consecutively:

- 1 Click the Draw Layer button in the Insert bar.
- 2 Control-drag (Windows) or Command-drag (Macintosh) to draw each layer.

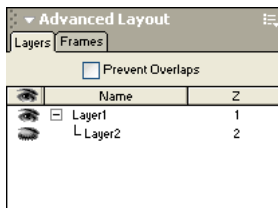
You can continue to draw new layers as long as you do not release the Control or Command key.

Related topics

- “About the Layers panel” on page 379
- “Setting Layers preferences” on page 380
- “Preventing layer overlaps” on page 387

About the Layers panel

The Layers panel is a way to manage the layers in your document. To open the Layers panel, choose Window > Others > Layers. Layers are displayed as a list of names, in order of z-index; the first created layer appears at the bottom of the list, and the most recently created layer at the top of the list. Nested layers are displayed as names connected to parent layers. Click the plus (+) or minus (-) icons (Windows) or the expander arrow (Macintosh) to show or hide nested layers.



Use the Layers panel to prevent overlaps, to change the visibility of layers, to nest or stack layers, and to select one or more layers.

Related topics

- “Creating layers on your page” on page 378
- “Setting Layers preferences” on page 380
- “Nesting layers” on page 380
- “Selecting layers” on page 381
- “Viewing and setting layer properties” on page 384
- “Changing the stacking order of layers” on page 385
- “Changing layer visibility” on page 385
- “Preventing layer overlaps” on page 387

Setting Layers preferences

Use the Layers category in the Preferences dialog box to specify the default settings for new layers you create.

To view or set layer preferences:

- 1 Choose Edit > Preferences or Dreamweaver MX > Preferences (Mac OS X).
- 2 Select Layers from the Category list.
- 3 Make changes as necessary.
For more information, click the Help button in the dialog box.
- 4 Click OK to close the dialog box.

Related topics

“Nesting layers” on page 380

“Selecting layers” on page 381

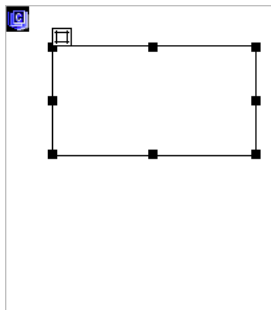
“Changing the stacking order of layers” on page 385

“Changing layer visibility” on page 385

“Preventing layer overlaps” on page 387

Nesting layers

A nested layer is a layer whose code is contained in another layer. Nesting is often used to group layers together. A nested layer moves with its parent layer and can be set to inherit visibility from its parent.



To create a nested layer, insert or draw a layer inside another layer, or use the Layers panel. To force layers to automatically nest when you draw a layer starting inside another layer, select the Nesting option in the Layer preferences. For more information, see “Setting Layers preferences” on page 380.

Note: Use the Netscape Resize Fix when you use nested layers in your page (Commands > Add/Remove Netscape Resize Fix). Otherwise, layers lose their positions when a visitor resizes a browser window in Netscape Navigator 4 versions. You can set a preference option to always insert the Netscape Resize Fix; for more information, see Dreamweaver Help (Help > Using Dreamweaver).

To create a nested layer, do one of the following:

- To insert a nested layer, place the insertion point inside an existing layer and choose Insert > Layer.
- To draw a nested layer, click the Draw Layer button in the Insert bar, then drag to draw a layer inside an existing layer. If Nesting is turned off in Layers preferences, Alt-drag (Windows) or Option-drag (Macintosh) to draw a layer inside an existing layer.

Tip: Nested layers may appear differently in different browsers. When you create nested layers, check their appearance in various browsers frequently during the design process.

To nest an existing layer inside another layer using the Layers panel:

- 1 Choose Window > Others > Layers to open the Layers panel.
- 2 Select a layer in the Layers panel, then Control-drag (Windows) or Command-drag (Macintosh) the layer to the target layer in the Layers panel. Release the mouse button when the name of the target layer is highlighted.

Manipulating layers

As you work with your page layout, you can select, move, resize, and align layers. You must select a layer before you can move, resize, or align it.

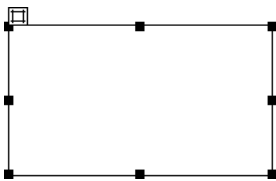
To prevent layers from overlapping each other as you move and resize them, use the Prevent Overlap option (see “Preventing layer overlaps” on page 387).

Selecting layers

Select one or more layers to manipulate them or change their properties.

To select a layer, do one of the following:

- Click the name of the layer in the Layers panel.
- Click a layer’s selection handle. If the selection handle isn’t visible, click anywhere inside the layer to make the handle visible.



- Click a layer’s border.
- Control-Shift-click (Windows) or Command-Shift-click (Macintosh) inside a layer. If multiple layers are selected, this deselects all other layers and selects only the one that you clicked.
- Click the layer-code marker (in the Design view) that represents the layer’s location in the HTML code. If the layer-code marker isn’t visible, choose View > Visual Aids > Invisible Elements.

To select multiple layers, do one of the following:

- Shift-click two or more layer names in the Layers panel.
- Shift-click inside or on the border of two or more layers.

When multiple layers are selected, the handles of the last selected layer are highlighted in black. The resize handles of the other layers are highlighted in white.

Resizing layers

You can resize an individual layer, or simultaneously resize multiple layers to make them the same width and height.

If the Prevent Overlaps option is on, you will not be able to resize a layer so that it overlaps with another layer (see “Preventing layer overlaps” on page 387).

To resize the selected layer, do one of the following:

- To resize by dragging, drag any of the layer’s resize handles.
- To resize one pixel at a time, hold down Control (Windows) or Option (Macintosh) while pressing an arrow key.

Note: The arrow keys move the right and bottom borders of the layer; you can’t resize using the top and left borders with this technique.

- To resize by the grid snapping increment, hold down Shift-Control (Windows) or Shift-Option (Macintosh) while pressing an arrow key. For information about setting the grid snapping increment, see “Snapping layers to the grid” on page 383.
- In the Property inspector, type values for width (W) and height (H).

Resizing a layer changes the width and height of the layer. It does not define how much of the layer’s content is visible. To define the visible region within a layer, see “Viewing and setting layer properties” on page 384.

To resize multiple layers at once:

1 In the Design view, select two or more layers.

2 Do one of the following:

- Choose Modify > Align > Make Same Width or Modify > Align > Make Same Height.

The first selected layers conform to the width or height of the last selected layer (highlighted in black).

- In the Property inspector, under Multiple Layers, enter width and height values. The values are applied to all selected layers.

Moving layers

You can move layers in the Design view in much the same way that you move objects in most basic graphics applications.

If the Prevent Overlaps option is on, you will not be able to move a layer so that it overlaps another layer. See “Preventing layer overlaps” on page 387.

To move one or more selected layers, do one of the following:

- To move by dragging, drag the selection handle of the last selected layer (highlighted in black).
- To move one pixel at a time, use the arrow keys. Hold down Shift while pressing an arrow key to move the layer by the current grid snapping increment. For information about setting the grid snapping increment, see “Snapping layers to the grid” on page 383.

Aligning layers

Use the layer alignment commands to align one or more layers with a border of the last layer selected.

When you align layers, child layers that aren't selected may move because their parent layer is selected and moved. To prevent this, don't use nested layers.

To align two or more layers:

- 1 Select the layers.
- 2 Choose **Modify > Align**, then select an alignment option.

For example, if you select **Top**, all of the layers move so that their top borders are in the same vertical position as the top border of the last selected layer (highlighted in black).

Snapping layers to the grid

Use the grid as a visual guide for drawing, positioning, or resizing layers in the Document window's Design view. You can make page elements automatically snap to the grid as you move them, and change the grid or control the snapping behavior by specifying grid settings. Snapping works whether or not the grid is visible.

To show or hide the grid:

Choose **View > Grid > Show Grid**.

To enable or disable snapping:

Choose **View > Grid > Snap to Grid**.

To change grid settings:

- 1 Choose **View > Grid > Grid Settings**.
The Grid Settings dialog box appears.
- 2 Set options as desired.
For more information, click the Help button in the dialog box.
- 3 Click **OK**.

Adding content to layers

Before you place objects in a layer, you must place the insertion point in the layer.

To place the insertion point in a layer:

Click anywhere inside the layer's borders.

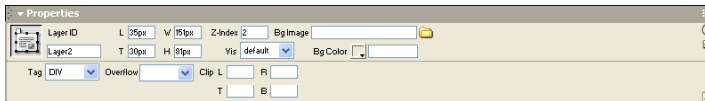
The layer's borders highlight, and the selection handle appears, but the layer itself is not selected. For information on selecting layers, see "Selecting layers" on page 381.

Viewing and setting layer properties

You can view and set various attributes of a layer in the Property inspector.

To view all layer properties:

- 1 Select a layer. For information on selecting layers, see “Selecting layers” on page 381.
- 2 Open the Property inspector by choosing Window > Properties.
- 3 If the Property inspector isn’t expanded, click the expander arrow in the lower right corner to see all properties.



To set layer properties with the Property inspector:

- 1 Select a layer. For information on selecting layers, see “Selecting layers” on page 381.
- 2 Open the Property inspector by choosing Window > Properties.
- 3 Change the layer’s attributes by setting properties.

For more information, click the Help button in the Property inspector.

Viewing and setting properties for multiple layers

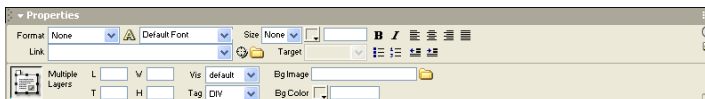
When you select two or more layers, the layer Property inspector displays text properties and a subset of the full layer properties, allowing you to modify several layers at once.

To select multiple layers:

Hold down Shift while selecting layers. For information on selecting layers, see “Selecting layers” on page 381.

To view all properties for multiple layers:

- 1 Select multiple layers.
- 2 Open the Property inspector by choosing Window > Properties.
- 3 If the Property inspector isn’t expanded, click the expander arrow in the lower right corner to see all properties.



To set properties for multiple layers at once with the Property inspector:

- 1 Select multiple layers.
- 2 Open the Property inspector by choosing Window > Properties.

3 Change the layers' attributes by setting properties.

For more information, click the Help button in the Property inspector.

Related topics

“Viewing and setting layer properties” on page 384

“Selecting layers” on page 381

Changing the stacking order of layers

Use the Property inspector or the Layers panel to change the stacking order of layers. The layer at the top of the Layers panel list is at the top of the stacking order, and appears in front of the other layers.

In HTML code, the stacking order, or *z*-index, of the layers determines the order in which they are drawn in a browser. You can change the *z*-index for each layer using the Layers panel or using the Property inspector.

To change the stacking order of layers in the Layers panel:

Choose Window > Others > Layers to open the Layers panel. Then do one of the following:

- Select and drag a layer up or down to the desired stacking order. A line appears as you move the layer, indicating where the layer will appear. Release the mouse button when the placement line appears in the desired place in the stacking order.
- In the Z column, click the number of the layer you are changing. Type a higher number than the existing number to move the layer up in the stacking order, or type a lower number to move the layer down in the stacking order.

To change the stacking order of layers using the Property inspector:

- 1 Choose Window > Others > Layers to open the Layers panel to see the current stacking order.
- 2 Select a layer in the Layers panel or in the Document window.
- 3 In the layer Property inspector, type a number in the Z-index text box.
 - Type a higher number to move the layer up in the stacking order.
 - Type a lower number to move the layer down in the stacking order.

Changing layer visibility

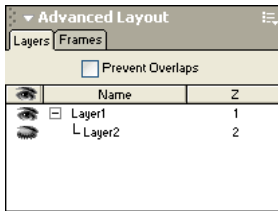
While working on your document, you can show and hide layers manually, using the Layers panel, to see how the page will appear under different conditions.

Note: The currently selected layer always becomes visible and appears in front of other layers while it's selected.

To change layer visibility:

- 1 Choose Window > Others > Layers to open the Layers panel.
- 2 Click in the eye-icon column for a layer to change its visibility.
 - An open eye means the layer is visible.
 - A closed eye means the layer is invisible.

- If there is no eye icon, usually the layer inherits visibility from its parent. (When layers are not nested, the parent is the document body, which is always visible.) Also, no eye icon appears when no visibility is specified (which appears in the Property inspector as Default visibility).



To change the visibility of all layers at once:

- 1 Choose Window > Others > Layers to open the Layers panel.
- 2 Click the header eye icon at the top of the column.

Note: This procedure can set all layers to visible or hidden, but not to inherit.

Related topic

For information on setting the default visibility for new layers, see “Setting Layers preferences” on page 380.

Using tables and layers for layout

Instead of using tables or Layout view to create your layout (see “Presenting Content with Tables” on page 227 and “Laying Out Pages in Layout View” on page 241), some web designers prefer to work with layers and tables. Dreamweaver allows you to create your layout using layers, and then (if you wish) convert them into tables, since 3.0 browsers don’t support layers. You can convert back and forth between layers and tables to adjust the layout and optimize page design.

You can’t convert layers to tables or tables to layers in a template document or in a document to which a template has been applied. Instead, create your layout in a non-template document, and convert it before saving it as a template.

A single table or layer cannot be converted. You can convert layers to tables and tables to layers only for an entire page.

If you want to generate separate 3.0 browser-compatible files from a file that uses layers, use the Convert option in the File menu (see “Converting to 3.0 browser compatibility” on page 387).

Note: Converting from layers to tables may result in tables with a large number of empty cells.

Converting between layers and tables

You can create your layout using layers, then convert the layers to tables so that your layout can be viewed in older browsers.

To convert layers to a table:

- 1 Choose Modify > Convert > Layers to Table.
- 2 In the dialog box that appears, select the desired options.
For more information, click the Help button in the dialog box.
- 3 Click OK.

The layers are converted to a table.

To convert tables to layers:

- 1 Choose Modify > Convert > Tables to Layers.
- 2 In the dialog box that appears, select the desired options.
For more information, click the Help button in the dialog box.
- 3 Click OK.

The tables are converted to layers. Empty cells are not converted to layers unless they have background colors.

Note: Page elements that were outside of tables are also placed in layers.

Preventing layer overlaps

Because table cells cannot overlap, Dreamweaver cannot create a table from overlapping layers. If you plan to convert the layers in a document to tables for compatibility with 3.0 browsers, use the Prevent Overlap option to constrain layer movement and positioning so that layers don't overlap.

To prevent layers from overlapping, do one of the following:

- Select the Prevent Overlaps option in the Layers panel.
- Choose Modify > Arrange > Prevent Layer Overlaps.

When this option is on, a layer can't be created in front of, moved or resized over, or nested within an existing layer. If you activate this option after creating overlapping layers, drag each overlapping layer to move it away from other layers. Dreamweaver does not automatically fix existing overlapping layers in the page when you enable Prevent Layer Overlaps.

When this option and snapping are enabled, a layer won't snap to the grid if it would cause two layers to overlap. Instead, it will snap to the edge of the closest layer.

Note: Certain actions allow you to overlap layers even when the Prevent Overlaps option is enabled. If you insert a layer using the Insert menu, enter numbers in the Property inspector, or reposition layers by editing the HTML source code, you can cause layers to overlap or nest while this option is enabled. If overlaps happen, drag overlapping layers in the Design view to separate them.

Converting to 3.0 browser compatibility

You can convert a page that uses layers or CSS to using tables and HTML markup, to make the page compatible with version 3.0 browsers. Dreamweaver creates a separate, converted document, while preserving the original document.

In general, you should convert a document only when you are completely satisfied with your original document's layout; otherwise, you must convert the document each time you change the original.

To convert a file for use with 3.0 browsers:

- 1 Choose File > Convert > 3.0 Browser Compatible.
- 2 In the dialog box that appears, choose whether to convert layers to tables, CSS styles to HTML markup (character styles), or both.
- 3 Click OK.

Dreamweaver opens the converted file in a new, untitled window. If you selected Convert Layers to Table or Both, all layers are replaced with a single table that preserves the original positioning of content.

Note: Overlapping layers cannot be converted, nor can layers that are off the page to the left or top.

If you selected Convert CSS Styles to HTML Markup or Both, CSS markup is replaced, where possible, with HTML character styles. Any CSS markup that cannot be converted to HTML is removed. For information on which styles are converted and which are removed, see the “CSS to HTML markup conversion table” on page 293.

Timeline code that animates layers is removed. Timeline code that is unrelated to layers (for example, behaviors or changes to an image’s source file) will continue to execute as specified. The timeline is automatically rewound to frame 1. For more information on timelines, see “Moving a layer using a timeline animation” on page 390.

Animating your layers

Dynamic HTML, or DHTML, refers to the combination of HTML with a scripting language that allows you to change style or positioning properties of HTML elements. Timelines, in Dreamweaver, use dynamic HTML to change the properties of layers and images over time. Use timelines to create animations that do not require any ActiveX controls, plug-ins, or Java applets (but do require JavaScript).

Note: The word *dynamic* can mean different things in different web-related contexts. Don’t confuse Dynamic HTML with the idea of a dynamic web page, which means a web page generated dynamically by server-side code before being served to a visitor. For more information about creating dynamic pages, see Chapter 30, “The Dreamweaver Workflow for Dynamic Page Design,” on page 483.

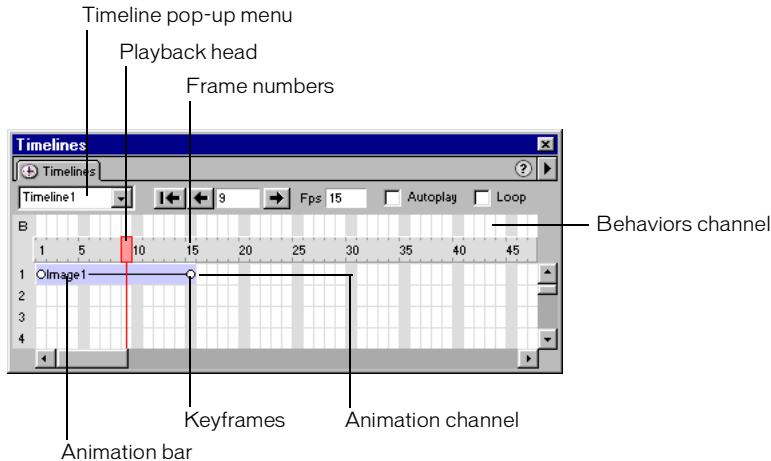
Timelines allow you to change the position, size, visibility, and stacking order of a layer. (The layer functions of timelines work only in 4.0 or later browsers.) Timelines are also useful for other actions that you want to occur after a page loads. For example, timelines can change the source file of an image tag so different images appear in the page over time.

To see the JavaScript code generated by a timeline, open the Document window’s Code view. The timeline code is in the `MM_initTimelines` function, inside a `script` tag in the head section of the document.

When editing the HTML of a document containing timelines, make sure you do not move, rename, or delete anything that a timeline refers to.

Using the Timelines panel

The Timelines panel shows how the properties of layers and images change over time. Choose Window > Others > Timelines to open the Timelines panel.



Timeline pop-up menu specifies which of the document's timelines is currently displayed in the Timelines panel.

Playback head shows which frame of the timeline is currently displayed in the Document window.

Frame numbers indicate the sequential numbering of frames. The number between the Back and Play buttons is the current frame number. You control the duration of animation by setting the total number of frames and the number of frames per second (fps). The default setting of 15 frames per second is a good average rate to use for most browsers running on common Windows and Macintosh systems.

Note: Faster rates may not improve performance. Browsers always play every frame of the animation, even if they cannot attain the specified frame rate. The frame rate is ignored if it is higher than the browser can manage.

Context menu contains various timeline-related commands.

Behaviors channel is the channel for behaviors that should be executed at a particular frame in the timeline.

Animation bars show the duration of each object's animation. A single row can include multiple bars representing different objects. Different bars cannot control the same object in the same frame.

Keyframes are frames in a bar where you have specified properties (such as position) for the object. Dreamweaver calculates intermediate values for frames in between keyframes. Small circles mark keyframes.

Animation channels display bars for animating layers and images.

Playback options

The following are the playback options for viewing the animation.



Rewind moves the playback head to the first frame in the timeline.

Back moves the playback head one frame to the left. Click Back and hold down the mouse button to play the timeline backward.

Play moves the playback head one frame to the right. Click Play and hold down the mouse button to play the timeline forward.

Autoplay makes a timeline begin playing automatically when the current page loads in a browser. Autoplay attaches a behavior to the page's `body` tag that executes the Play Timeline action when the page loads.

Loop makes the current timeline loop indefinitely while the page is open in a browser. Loop inserts the Go to Timeline Frame behavior in the Behaviors channel after the last frame of the animation. Double-click the behavior's marker in the Behaviors channel to edit the parameters for this behavior and change the number of loops.

Moving a layer using a timeline animation

The most common kind of timeline animation involves moving a layer along a path. Timelines can move only layers. To make images or text move, create a layer using the Draw Layer button on the Insert bar and then insert images, text, or any other type of content in the layer (see "Creating layers on your page" on page 378).

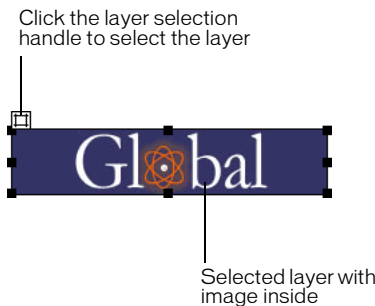
Timelines can also change other attributes of layers and images; for more information, see "Changing image and layer properties with timelines" on page 393.

To animate a layer using a timeline:

- 1 Move the layer to where it should be when the animation begins.
- 2 Choose Window > Others > Timelines.

3 Select the layer you want to animate.

Make sure you have selected the desired element. Click the layer marker or the layer selection handle, or use the Layers panel to select a layer. For more information, see “Manipulating layers” on page 381. When a layer is selected, handles appear around it as shown in the following illustration.



Clicking inside the layer places a blinking insertion point inside the layer, but it does not select the layer.

4 Choose **Modify > Timeline > Add Object to Timeline** or simply drag the selected layer into the Timelines panel.

A bar appears in the first channel of the timeline. The name of the layer appears in the bar.

5 Click the keyframe marker at the end of the bar.

6 Move the layer on the page to where it should be at the end of the animation.

A line appears showing the path of the animation in the Document window.

7 If you want the layer to move in a curve, select its animation bar and **Control-click** (Windows) or **Command-click** (Macintosh) a frame in the middle of the bar to add a keyframe at the frame you clicked, or click a frame in the middle of the animation bar and choose **Add Keyframe** from the context menu.

Repeat this step to define additional keyframes.

8 Hold down the **Play** button to preview the animation on the page.

Repeat the procedure to add additional layers and images to the timeline and to create a more complex animation.

Creating a timeline by dragging a path

If you want to create an animation with a complex path, it may be more efficient to record the path as you drag the layer rather than creating individual keyframes.

To create a timeline by dragging a path:

- 1 Select a layer.
- 2 Move the layer to where it should be when animation begins.
Make sure you have selected the layer; if the insertion point is in the layer, the layer is not selected. To select a layer, click the layer marker or the layer selection handle, or use the Layers panel. For more information, see “Manipulating layers” on page 381.
- 3 Choose **Modify > Timeline > Record Path of Layer**.
- 4 Drag the layer around the page to create a path.
- 5 Release the layer at the point where the animation should stop.
Dreamweaver adds an animation bar to the timeline, containing the appropriate number of keyframes.
- 6 In the Timelines panel, click the Rewind button; then hold down the Play button to preview your animation.

Modifying timelines

After defining a timeline’s basic components, you can make changes such as adding and removing frames, changing the start time of the animation, and so on.

To modify a timeline, do any of the following:

To make the animation play longer, drag the end frame marker to the right. All the keyframes in the animation shift so that their relative positions remain constant. To prevent the other keyframes from moving, **Alt-drag** (Windows) or **Option-drag** (Macintosh) the end frame marker.

- To make the layer reach a keyframe position earlier or later, move the keyframe marker left or right in the bar.
- To change the start time of an animation, select one or more of the bars associated with the animation (press **Shift** to select more than one bar at a time) and drag left or right.
- To shift the location of an entire animation path, select the entire bar and then drag the object on the page. Dreamweaver adjusts the position of all keyframes. Making any type of change with an entire bar selected changes all the keyframes.
- To add or remove frames in the timeline, choose **Modify > Timeline > Add Frame** or **Modify > Timeline > Remove Frame**.
- To make the timeline play automatically when the page opens in a browser, click **Autoplay**. Autoplay attaches a behavior to the page that executes the Play Timeline action when the page loads.
- To make the timeline loop continuously, click **Loop**. Loop inserts the Go To Timeline Frame action in the Behaviors channel after the last frame of the animation. You can edit the parameters for this behavior to define the number of loops.

Changing image and layer properties with timelines

In addition to moving layers with timelines, you can change the visibility, size, and stacking order of a layer; you can also change the source file of an image.

To change image and layer properties with a timeline:

- 1 In the Timelines panel, do one of the following:
 - Select an existing keyframe in the bar controlling the object you want to change. (The start and end frames are always keyframes.)
 - Create a new keyframe by clicking a frame in the middle of the animation bar and choosing **Modify > Timeline > Add Keyframe**. You can instead create a new keyframe by **Control-clicking (Windows)** or **Command-clicking (Macintosh)** a frame in the animation bar.
- 2 Define new properties for the object by doing one of the following:
 - To change the source file of an image, click the folder icon next to the **Src** text box in the Property inspector, then browse to and select a new image.
 - To change the visibility of a layer, choose **inherit**, **visible**, or **hidden** from the pop-up menu in the **Vis** text box of the Property inspector. Alternatively, use the eye icons in the Layers panel. See “Changing layer visibility” on page 385.
 - To change the size of a layer, drag the layer’s resize handles or enter new values in the **Width** and **Height** text boxes in the Property inspector. Not all browsers can dynamically change the size of a layer.
 - To change the stacking order of a layer, enter a new value in the **Z-Index** text box or use the Layers panel to change the stacking order of the current layer (see “Changing the stacking order of layers” on page 385).
- 3 Hold down the Play button to see the animation.

Using multiple timelines

Instead of trying to control all the action on a page with one timeline, it’s easier to work with separate timelines that control discrete parts of the page. For example, a page might include interactive elements that each trigger a different timeline.

To manage multiple timelines, do any of the following:

- To create a new timeline, choose **Modify > Timeline > Add Timeline**.
- To remove the selected timeline, choose **Modify > Timeline > Remove Timeline**. This permanently removes all animations from the selected timeline.
- To rename the selected timeline, choose **Modify > Timeline > Rename Timeline** or enter a new name in the Timeline pop-up menu in the Timelines panel.
- To view a different timeline in the Timelines panel, choose a new timeline from the Timeline pop-up menu in the Timelines panel.

Copying and pasting animations

Once you have an animation sequence that you like, you can copy it and paste it into another area of the current timeline, into another timeline in the same document, or into a timeline in another document. You can also copy and paste multiple sequences at once.

To cut or copy and paste animation sequences:

- 1 Click an animation bar to select a sequence. To select multiple sequences, Shift-click multiple animation bars; to select all sequences, press Control+A (Windows) or Command+A (Macintosh).
- 2 Copy or cut the selection.
- 3 Do one of the following:
 - Move the playback head to another spot in the current timeline.
 - Select another timeline from the Timeline pop-up menu.
 - Open another document, or create a new one, and then click in the Timelines panel.
- 4 Paste the selection into the timeline.

Animation bars for the same object cannot overlap, because a layer cannot be in two places at one time (nor can an image have two different sources at a time). If the animation bar you are pasting would overlap another animation bar for the same object, Dreamweaver automatically shifts the selection to the first frame that doesn't overlap.

There are two principles to keep in mind when pasting animation sequences into another document:

- If you copy an animation sequence for a layer and the new document contains a layer with the same name, Dreamweaver applies the animation properties to the existing layer in the new document.
- If you copy an animation sequence for a layer and the new document does not contain a layer with the same name, Dreamweaver pastes the layer and its contents from the original document along with the animation sequence. To apply the pasted animation sequence to another layer in the new document, choose Change Object from the context menu and select the name of the second layer from the pop-up menu. Delete the pasted layer if desired.

Applying an animation sequence to a different object

To save time, you can create an animation sequence once and apply it to each of the remaining layers in your document.

To apply an existing animation sequence to other objects:

- 1 In the Timelines panel, select the animation sequence and copy it.
- 2 Click any frame of the Timelines panel and paste the sequence at that frame.
- 3 Right-click (Windows) or Control-click (Macintosh) the pasted animation sequence and choose Change Object from the context menu.
- 4 In the dialog box that appears, choose another object from the pop-up menu and click OK.
- 5 Repeat steps 2 through 4 for any remaining objects that you want to follow the same animation sequence.

You can also change your mind about which layer should be animated after creating an animation sequence; simply follow steps 3 and 4 above (no copying or pasting is necessary).

Renaming timelines

To rename the timeline currently displayed in the Timelines panel:

- 1 Choose Modify > Timeline > Rename Timeline.
- 2 In the Rename Timeline dialog box, enter a new name.

If your document contains the Play Timeline behavior action (for example, if it contains a button that the visitor must click to start the timeline), you must edit the behavior to reflect the new timeline name.

Animation tips for timelines

The following suggestions can improve the performance of your animations and make creating animations easier:

- Show and hide layers instead of changing the source file for multiple-image animations. Switching the source file of an image can slow down the animation, because the new image must be downloaded. There will be no noticeable pauses or missing images if all images are downloaded at once in hidden layers before the animation runs.
- Extend animation bars to create smoother motion. If animation looks choppy and images jump between positions, drag the end frame of the layer's animation bar to extend the motion over more frames. Making the animation bar longer creates more data points between the start and end point of the movement and also makes the object move more slowly. Try increasing the number of frames per second (fps) to improve speed, but be aware that most browsers running on average systems cannot animate much faster than 15 fps. Test the animation on different systems with different browsers to find the best settings.
- Don't animate large bitmaps. Animating large images results in slow animations. Instead, create composites and move small parts of the image. For example, show a car moving by animating only the wheels.
- Create simple animations. Do not create animations that demand more than current browsers can provide. Browsers always play every frame in a timeline animation, even when system or Internet performance decreases.

Animating layers using behavior actions

Attach the following behavior actions to a link, button, or other object to control timelines and layers. To create interesting effects, you can place behaviors containing these actions in the Behaviors channel. For example, you can make a timeline stop itself. For more information, see “Attaching a behavior to a timeline” on page 352 and “Using JavaScript Behaviors” on page 349.

Drag Layer lets the visitor drag a layer. Use this action to create puzzles, slider controls, and other moveable user interface elements. See “Drag Layer” on page 358.

Show-Hide Layers shows, hides, or restores the default visibility of one or more layers. This action is useful for showing information as the visitor interacts with the page. See “Show-Hide Layers” on page 368.

Play Timeline and **Stop Timeline** let visitors start and stop a timeline by clicking a link or button. These actions can also start and stop a timeline automatically when the visitor points to a link, image, or other object. See “Play Timeline and Stop Timeline” on page 375.

Go To Timeline Frame causes the timeline to jump to a certain frame. The Loop check box in the Timelines panel adds the Go To Timeline Frame action after the last frame of the animation, causing it to go to frame 1 and start the animation again. See “Go to Timeline Frame” on page 374.

Set Text of Layer replaces the content and formatting of an existing layer on a page with the content you specify. The content can include any valid HTML. See “Set Text of Layer” on page 366.

Part VII

Working with Multiple Pages

Link your pages together, and reuse design elements and layouts from one page to another. Then test your site.

This part contains the following chapters:

- Chapter 26, “Linking and Navigation”
- Chapter 27, “Managing Site Assets, Libraries, and Templates”
- Chapter 28, “Testing a Site”

CHAPTER 26

Linking and Navigation

Once you've set up a Dreamweaver site to store your website documents and have created HTML pages, you'll want to create connections from your documents to other documents.

Macromedia Dreamweaver MX provides several ways to create hypertext links to documents, images, multimedia files, or downloadable software. You can establish links to any text or image anywhere within a document, including text or images located in a heading, list, table, layer, or frame.

For a visual representation of how your files are linked together, use the site map. In the site map you can add new documents to your site, create and remove document links, and check links to dependent files (see "Using the site map" on page 81).

There are several different ways of creating and managing links. Some web designers prefer creating links to nonexistent pages or files as they work, while others prefer creating all the files and pages first, then adding the links. Another way to manage links is to create "placeholder" pages which stand in for the final file and let you add links quickly and check them before you have actually completed all the pages. For more information about checking links, see "Checking links in a page or site" on page 465.

This chapter contains the following sections:

"About document locations and paths" on page 399

"Creating links" on page 402

"Managing links" on page 409

"Creating jump menus" on page 412

"Creating navigation bars" on page 414

"Creating image maps" on page 416

"Attaching behaviors to links" on page 418

About document locations and paths

Understanding the file path between the document you're linking from and the document you're linking to is essential to creating links.

Each web page has a unique address, called a Uniform Resource Locator (URL). (For more information on URLs, see the World Wide Web Consortium page on naming and addressing.) However, when you create a local link (a link from one document to another on the same site), you generally don't specify the entire URL of the document you're linking to; instead, you specify a relative path from the current document or from the site's root folder.

The three types of link paths are as follows:

- Absolute paths (such as `http://www.macromedia.com/support/dreamweaver/contents.html`). See “Absolute paths” on page 400.
- Document-relative paths (such as `dreamweaver/contents.html`). See “Document-relative paths” on page 400.
- Site root-relative paths (such as `/support/dreamweaver/contents.html`). See “Site root-relative paths” on page 401.

Using Dreamweaver, you can easily select the type of document path to create for your links. See “Linking files and documents” on page 403.

Note: It is best to use the type of linking you prefer and are most comfortable with—either site or document relative. Browsing to links, as opposed to typing in the paths, ensures that you always enter the right path.

Absolute paths

Absolute paths provide the complete URL of the linked document, including the protocol to use (usually `http://` for web pages). For example, `http://www.macromedia.com/support/dreamweaver/contents.html` is an absolute path.

You must use an absolute path to link to a document on another server. While you can also use absolute-path links for local links (to documents in the same site), that approach is discouraged—if you move the site to another domain, all of your local absolute-path links will break. Using relative paths for local links also provides greater flexibility if you need to move files within your site.

Note: When inserting images (not links): if you use an absolute path to an image that resides on a remote server and is not available on the local hard drive, you will not be able to view the image in your document window. Instead, you must preview the document in a browser to see it. If possible, use document or site root-relative paths for images. For more information, see “Inserting an image” on page 298.

Document-relative paths

Document-relative paths are the most appropriate paths to use for local links in most websites. They’re particularly useful when the current document and the linked document are in the same folder and are likely to remain together. You can also use a document-relative path to link to a document in another folder by specifying the path through the folder hierarchy from the current document to the linked document.

The basic idea of document-relative paths is to omit the part of the absolute URL that is the same for both the current document and the linked document, providing only the portion of the path that differs.

- To link to another file in the same folder as the current document, simply enter the filename.
- To link to a file in a subfolder of the current document’s folder, provide the name of the subfolder, then a forward slash (`/`), and then the filename.
- To link to a file in the parent folder of the current document’s folder, precede the filename with `../` (where “`..`” means “up one level in the folder hierarchy”).

For example, suppose you have a site with the following structure:

To link from contents.html to other files:

- To link from contents.html to hours.html (both files are in the same folder), the filename is the relative path: hours.html.
- To link to tips.html (in the subfolder named resources), use the relative path resources/tips.html.

Each forward slash (/) represents moving down one level in the folder hierarchy.

- To link to index.html (in the parent folder, one level above contents.html), use the relative path ../index.html.

Each ../ represents moving up one level the folder hierarchy.

- To link to catalog.html (in a different subfolder of the parent folder), use the relative path ../products/catalog.html.

The ../ moves up to the parent folder; the products/ moves down into the products subfolder.

Note: Always save a new file before creating a document-relative path; a document-relative path is not valid without a definite starting point. If you create a document-relative path before saving the file, Dreamweaver temporarily uses an absolute path beginning with file:// until the file is saved; when you save the file, Dreamweaver converts the file:// path into a relative path.

When you move a group of files as a group—for example, when you move an entire folder, so that all the files inside that folder retain the same relative paths to each other—you don't need to update document-relative links between those files. However, when you move an individual file that contains document-relative links, or an individual file that's linked to by a document-relative link, you do need to update those links. (If you move or rename files using the Site panel, Dreamweaver updates all relevant links automatically.)

Site root-relative paths

Site root-relative paths provide the path from the site's root folder to a document. You may want to use these types of paths if you are working on a large website that uses several servers, or one server that hosts several different sites. However, if you are not familiar with this type of path, you may want to stick to document-relative paths.

A site root-relative path begins with a leading forward slash, which stands for the site root folder. For example, /support/tips.html is a site root-relative path to a file (tips.html) in the support subfolder of the site's root folder.

A site root-relative path often provides the best way to specify links in a website in which you need to frequently move HTML files from one folder to another. When you move a document that contains root-relative links, you don't need to change the links; for example, if your HTML files use root-relative links for dependent files (such as images), then if you move an HTML file, its dependent-file links are still valid. However, when you move or rename the documents linked to with root-relative links, you do need to update those links, even if the documents' paths relative to each other haven't changed.

For example, if you move a folder, all root-relative links to files within that folder must be updated. (If you move or rename files using the Site panel, Dreamweaver updates all relevant links automatically.)

To use site root-relative paths, first define a local folder in Dreamweaver by choosing a local root folder to serve as the equivalent of the document root on a server (see “Setting up a Dreamweaver site” on page 59).

Dreamweaver uses this folder to determine the site root-relative paths to files.

Note: Root-relative links are interpreted by servers, not by browsers, so if you open a local page that uses root-relative links in your browser (without using Preview in Browser from within Dreamweaver), the links don't work. When you use the Preview in Browser command to preview a document that uses root-relative links, Dreamweaver temporarily converts those links (in the previewed file only) to use document-relative paths. However, you can preview only one page that uses root-relative links at a time—if you follow a link from the previewed page, the next page's root-relative links are not converted, and the browser can't follow such links. Previewing pages in framesets that use root-relative links results in similar problems.

To preview a set of pages that use root-relative links, do one of the following:

- Put the files on a remote server and view them from there.
- (Windows only) Choose Edit > Preferences, select Preview in Browser from the category list on the left, and then select Preview Using Local Server.

Note: To use this option, you must be running a web server on your local computer.

Creating links

The HTML tag for creating a hypertext link is called an anchor tag or an a tag. Dreamweaver creates an anchor tag for objects, text, or images you create links from. You can create links to other documents and files, and links to specific places in a single document using the a href tag.

For example, if you select the text Home Page in the Document window, then create a link to a file named home.htm, the HTML source code for the link looks like this:

```
<a href="home.htm">Home Page</a>
```

If you are creating a link to a specific place in a document, first create a named anchor (for example, a name="MainMenu"). Then create a link within the page that refers to that named anchor (for example, a href="#MainMenu").

Before creating links, make sure you understand how document-relative paths, site root-relative paths, and absolute paths work (see "About document locations and paths" on page 399).

You can create several types of links in a document:

- A link to another document or to a file, such as a graphic, movie, PDF, or sound file. See "Linking files and documents" on page 403.
- A named anchor link, which jumps to a specific location within a document. See "Linking to a specific place in a document" on page 406.
- An e-mail link, which creates a new blank e-mail message with the recipient's address already filled in. See "Creating a hyperlink" on page 407.
- Null and script links, which enable you to attach behaviors to an object or to create a link that executes JavaScript code. See "Creating null and script links" on page 408.

Using Dreamweaver, there are several ways to create local links (links between documents in the same site):

- Use the site map to view, create, change, or delete links.
- In the Document window, select text or a page element, and then use Modify > Make Link to select a file to link to.
- Use the Property inspector, by selecting text or a page element in the document, and then using the Property inspector folder icon or Point-to-File icon to select a file to link to or typing the path of the file.

Note: Typing URLs or paths to a file can lead to incorrect paths and links that don't work. To ensure that the path is correct, use the folder icon to browse to your link.

- In the Document window, select text or a page element, choose Make Link from the context menu, and then select a file to link to.

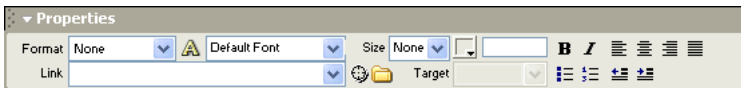
To create an external link (to a document on another site), you must type the absolute path (with proper protocol) in the Property inspector. Make sure that you enter the entire path (including `http://`) when creating external links.

Linking files and documents

You can use the Property inspector and the Point-to-File icon to create links from an image, an object, or text to another document or file. For information about using the site map to create links, see “Creating and modifying links in the site map” on page 410.

To link documents using the Property inspector’s folder icon or link text box:

- 1 Select text or an image in the Document window’s Design view.
- 2 Open the Property inspector (Window > Properties), and then do one of the following:



- Click the folder icon to the right of the Link text box to browse to and select a file.

The path to the linked document appears in the URL field. Use the Relative To pop-up menu in the Select HTML File dialog box to indicate whether to make the path document-relative or root-relative, then click Select.

Note: When you change the path type in the Relative To field, Dreamweaver uses your choice as the default path type for any future links until you change the path type again.

- Type the path and filename of the document in the Link text box.

To link to a document in your site, enter a document-relative or site root-relative path. To link to a document outside your site, enter an absolute path including the protocol (such as `http://`). You can use this approach to enter a link for a file that hasn’t been created yet.

- 3 From the Target pop-up menu, select a location in which to open the document.

To make the linked document appear somewhere other than in the current window or frame, select an option from the Target pop-up menu on the Property inspector:

- `_blank` loads the linked document in a new, unnamed browser window.
- `_parent` loads the linked document in the parent frame or parent window of the frame that contains the link. If the frame containing the link is not nested, then the linked document loads in the full browser window.
- `_self` loads the linked document in the same frame or window as the link. This target is the default, so you usually don’t have to specify it.
- `_top` loads the linked document in the full browser window, thereby removing all frames.

Tip: If all the links on your page will be set to the same target, you can specify this target once by choosing Insert > Head Tag > Base and selecting the target information. For information about targeting frames, see “Controlling frame content with links” on page 265.

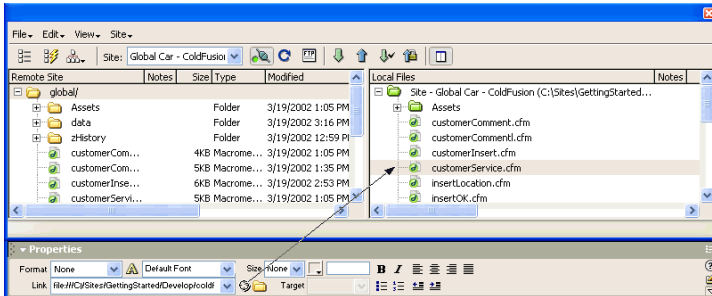
To link documents using the Point-to-File icon:

- 1 Select text or an image in the Document window's Design view.
- 2 Drag the Point-to-File icon at the right of the Link text box in the Property inspector and point to another open document, a visible anchor in an open document, or a document in the Site panel.

The Link text box updates to show the link.

Note: You can link to an open document only if your documents are not maximized in the Document window. When you point to an open document, that document moves to the foreground of your screen while you are making your selection.

- 3 Release the mouse button.



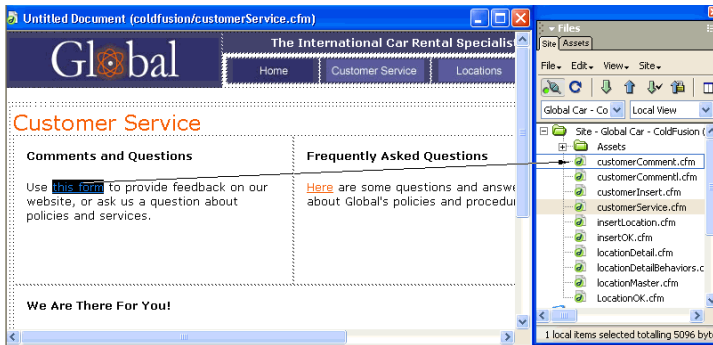
Dragging the Point-to-File icon from the Property inspector to a file in the Site panel

To create a link from a selection in an open document:

- 1 Select text in the Document window.
- 2 Shift-drag from the selection.
The Point-to-File icon appears as you drag.
- 3 Point to another open document, a visible anchor in an open document, or a document in the Site panel.

Note: You can link to an open document only if your documents are not maximized in the Document window. When you point to an open document, that document moves to the foreground of your screen while you are making your selection.

- 4 Release the mouse button.

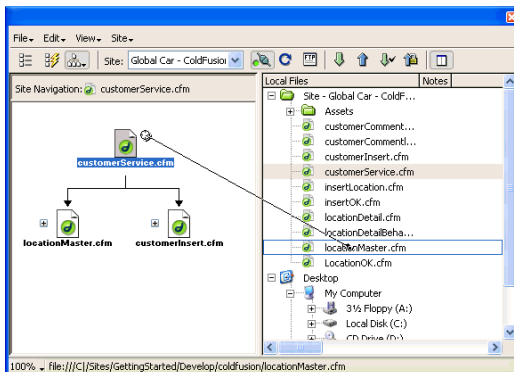


Shift-dragging the Point-to-File icon from text in the Document window to a file in the Site panel

To link documents using the site map and the Point-to-File icon:

- 1 Expand the Site panel, and then display both the Site Files and the Site Map views by holding down the Site Map icon and choosing Map and Files.
- 2 Select an HTML file in the site map.
The Point-to-File icon appears next to the file.
- 3 Drag the Point-to-File icon and point to another file in the site map or to a local file in the Site Files view.
- 4 Release the mouse button.

A hypertext link with the name of the linked file is placed at the bottom of the selected HTML file. This method works well when you are building your site and you want to create links across the site quickly.



Linking to a specific place in a document

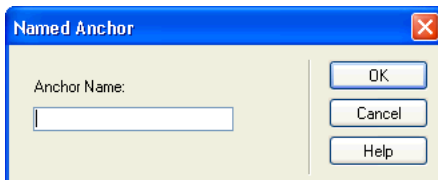
You can use the Property inspector to link to a particular section of a document by first creating named anchors. Named anchors let you set markers in a document, which are often placed at a specific topic or at the top of a document. You can then create links to these named anchors, which quickly take your visitor to the specified position.

Creating a link to a named anchor is a two-step process. First, create a named anchor; then create a link to the named anchor.

To create a named anchor:

- 1 In the Document window's Design view, place the insertion point where you want the named anchor.
- 2 Do one of the following:
 - Choose Insert > Named Anchor.
 - Press Control+Alt+A (Windows) or Command+Option+A (Macintosh).
 - Select the Common tab in the Insert bar and click the Named Anchor button.

The Named Anchor dialog box appears.



- 3 In the Anchor Name field, type a name for the anchor.

For more information, click the Help button in the dialog box.

The anchor marker appears at the insertion point.

Note: If you do not see the anchor marker, choose View > Visual Aids > Invisible Elements.

To link to a named anchor:

- 1 In the Document window's Design view, select text or an image to create a link from.
- 2 In the Link text box of the Property inspector, type a number sign (#) and the name of the anchor. For example:
 - To link to an anchor named “top” in the current document, type **#top**.
 - To link to an anchor named “top” in a different document in the same folder, type **filename.html#top**.

Note: Anchor names are case sensitive.

To link to a named anchor using the point-to-file method:

- 1 Open the document containing the named anchor you want.

Note: Choose View > Visual Aids > Invisible Elements to make the anchor visible if you don't see it.

- 2 In the Document window's Design view, select text or an image you want to link from. (If this is another open document, you must switch to it.)
- 3 Do one of the following:
 - Click the Point-to-File icon to the right of the Link text box in the Property inspector and drag it to the anchor you want to link to: either an anchor within the same document or an anchor in another open document.
 - Shift-drag in the Document window from the selected text or image to the anchor you want to link to: either an anchor within the same document or an anchor in another open document.

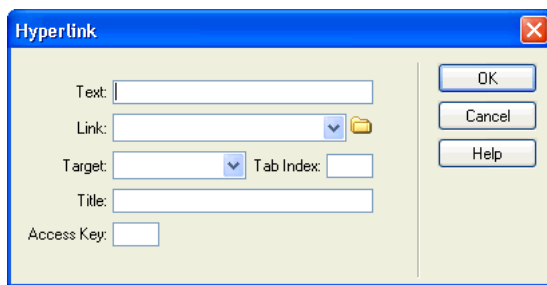
Creating a hyperlink

You can add hyperlink text to link to another file.

To add a hyperlink using the Hyperlink command:

- 1 Place the insertion point in the document where you want the hyperlink to appear.
- 2 Do one of the following to display the Insert Hyperlink dialog box:
 - Choose Insert > Hyperlink.
 - Select the Common tab in the Insert bar, and then click the Hyperlink button.

The Hyperlink dialog box appears.



- 3 Complete the dialog box.

For more information, click the Help button in the dialog box.
- 4 Click OK.

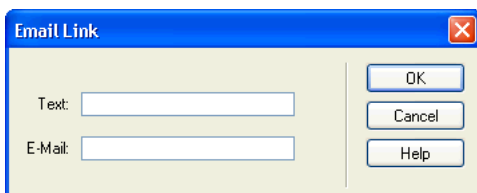
Creating an e-mail link

An e-mail link opens a new blank message window (using the mail program associated with the user's browser) when clicked. In the e-mail message window, the To field is automatically updated with the address specified in the e-mail link.

To create an e-mail link using the Insert E-Mail Link command:

- 1 In the Document window's Design view, position the insertion point where you want the e-mail link to appear, or select the text or image you want to appear as the e-mail link.

- 2 Do one of the following to insert the link:
 - Choose Insert > E-Mail Link.
 - Select the Common tab in the Insert bar, and then click the Insert E-Mail Link button.The E-Mail Link dialog box appears.



- 3 Complete the dialog box.
For more information, click the Help button in the dialog box.
- 4 Click OK.

To create an e-mail link using the Property inspector:

- 1 Select text or an image in the Document window's Design view.
- 2 In the Link text box of the Property inspector, type **mailto:** followed by an e-mail address.
Do not type any spaces between the colon and the e-mail address. For example, type **mailto:jdoe@macromedia2.com**.

Creating null and script links

The most familiar kinds of links are those to documents and named anchors (described in “Linking files and documents” on page 403 and “Linking to a specific place in a document” on page 406, respectively), but there are other types of links as well.

A null link is an undesignated link. Use null links to attach behaviors to objects or text on a page. Once you have created a null link, you can attach a behavior to it to swap an image or to display a layer when the pointer is moved over the link. For information about attaching behaviors to objects, see “Applying a behavior” on page 351.

Script links execute JavaScript code or call a JavaScript function and are useful for giving visitors additional information about an item without leaving the current web page. Script links can also be used to perform calculations, form validations, and other processing tasks when a visitor clicks a specific item.

To create a null link:

- 1 Select text, an image, or an object in the Document window's Design view.
- 2 In the Property inspector, type **javascript::** (the word *javascript*, followed by a colon, followed by a semicolon) in the Link text box.

To create a script link:

- 1 Select text, an image, or an object in the Document window's Design view.
- 2 In the Link text box of the Property inspector, type **javascript:** followed by some JavaScript code or a function call.

For example, typing **javascript:alert('This link leads to the index')** in the Link text box produces a link that, when clicked, displays a JavaScript alert box with the message `This link leads to the index`.

Note: Because the JavaScript code appears in the HTML between double quotation marks (as the value of the href attribute), you must use single quotation marks in the script code or "escape" any double quotation marks by preceding them with a backslash (for example, `\ "This link leads to the index\"`).

Managing links

Dreamweaver can update links to and from a document whenever you move or rename the document within a local site. This feature works best when you store your entire site (or an entire self-contained section of it) on your local disk. Dreamweaver does not change files in the remote folder until you put the local files on or check them in to the remote server.

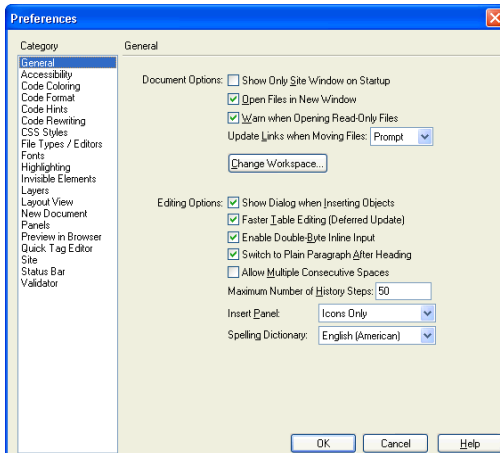
To turn on link management in Dreamweaver:

- 1 Choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X).

The Preferences dialog box appears.

- 2 Select General from the category list on the left.

The General preferences options appear.



- 3 In the Document Options section, select Always or Prompt from the Update Links when Moving Files pop-up menu.

If you choose Always, Dreamweaver automatically updates all links to and from a selected document whenever you move or rename it. (For specific instructions on what to do when you delete a file, see “Changing a link sitewide” on page 412.)

If you choose Prompt, Dreamweaver first displays a dialog box that lists all the files affected by the change. Click Update to update the links in these files, or click Don't Update to leave the files unchanged.

- 4 Click OK.

To make the updating process faster, Dreamweaver can create a cache file in which to store information about all the links in your local folder. This cache file is created when you select the Cache option in the Site Definition dialog box, and the cache file updates invisibly as you use Dreamweaver to add, change, or delete links to files on your local site.

To create a cache file for your site:

- 1 Choose Site > Edit Sites.

The Edit Sites dialog box appears.

- 2 Select a site, and then click Edit.

- 3 Select Local Info from the category list on the left.

The Site Definition dialog box displays the Local Info options.

- 4 In the Local Info category, select the Enable Cache checkbox.

The first time you change or delete links to files in your local folder after launching Dreamweaver, Dreamweaver prompts you to load the cache. If you click Yes, the cache loads and Dreamweaver updates all the links to the file you just changed. If you click No, the change is noted in the cache, but the cache does not load, and Dreamweaver does not update links.

It may take a few minutes for the cache to load on larger sites; most of this time is spent comparing the time stamps of the files on the local site with the time stamps recorded in the cache to see if the cache is out of date. If you have not changed any files outside Dreamweaver, you can safely click the Stop button when it appears.

To re-create the cache for your site:

In the Site panel, choose Site > Recreate Site Cache.

Creating and modifying links in the site map

You can modify the structure of the site in the site map by adding, changing, and removing links. Dreamweaver automatically updates the site map to display the changes to the site.

To add a link, do one of the following:

- Drag a page from the Windows Explorer or the Macintosh Finder onto a page in the site map.

Note: Make sure the Site panel is docked, and then click the Expand arrow. Hold down the Site Map button, and then select Files and Map.

- Select an HTML page in the site map, and then choose Site > Link to Existing File (Windows) or Site > Site Map View > Link to Existing File (Macintosh), or choose Link to Existing File from the context menu.
- Select an HTML page in the site map, and then choose Site > Link to New File (Windows) or Site > Site Map View > Link to New File (Macintosh), or choose Link to New File from the context menu.
- Select an HTML page in the site map; the Point-to-File icon appears. Drag the icon to the object you want to link to: either a file in the Site Files view, an open Dreamweaver document, or a named anchor in a document open on the desktop.

Note: For more information, see “Linking files and documents” on page 403.

To change a link:

- 1 In the site map, select a page to which you want to change the link (so that the document that currently links to this page will point to another page), and then do one of the following:
 - Choose Site > Change Link (Windows) or Site > Site Map View > Change Link (Macintosh).
Note: Use the Site panel’s Site menu.
 - Right-click (Windows) or Control-click (Macintosh), and choose Change Link from the context menu.
- 2 Browse to the file you want the link to point to or type a URL.
- 3 Click OK.

To remove a link, do one of the following:

- Select the page in the site map, and then choose Site > Remove Link (Windows) or Site > Site Map View > Remove Link (Macintosh).
Note: Use the Site panel’s Site menu.
- Select the page in the site map, and right-click (Windows) or Control-click (Macintosh), and choose Remove Link from the context menu.

Removing a link does not delete the file, but removes the link from the HTML source on the page that is pointing to the link.

To open the source of a link:

Select a file in the site map, and do one of the following:

- Choose Site > Open Source of Link (Windows) or Site > Site Map View > Open Source of Link (Macintosh).
- Right-click (Windows) or Control-click (Macintosh), and choose Open Source of Link from the context menu.

The Property inspector and the source file containing the link open in the Document window, with the link highlighted.

Changing a link sitewide

In addition to having Dreamweaver update links automatically whenever you move or rename a file, you can manually change all links (including e-mail, FTP, null, and script links) to point somewhere else.

You can use this option at any time (for example, you might have the words “this month’s movies” linked to /movies/july.html throughout your site, and on August 1 you must change those links to point to /movies/august.html), but it’s particularly useful when you want to delete a file that other files link to.

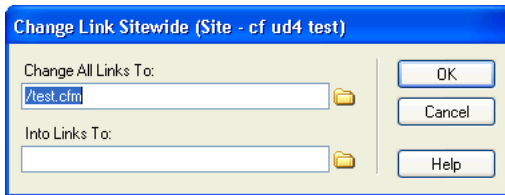
To change a link sitewide:

- 1 Select a file in the Local view of the Site panel.

Note: If you are changing an e-mail, FTP, null, or script link, you do not need to select a file.

- 2 Choose Site > Change Link Sitewide.

The Change Link Sitewide dialog box appears.



- 3 Complete the dialog box

For more information, click the Help button in the dialog box.

- 4 Click OK.

Dreamweaver updates any documents that link to the selected file, making them point to the new file, using the path format already used in the document (for example, if the old path was document relative, the new path is also document relative). The type of the link, whether document or root-relative, doesn’t matter. Dreamweaver automatically updates the link.

After you change a link sitewide, the selected file becomes an orphan (that is, no files on your local disk point to it). You can safely delete it without breaking any links in your local Dreamweaver site.

Note: Because these changes occur locally, you must manually delete the corresponding orphan file in the remote folder and put or check in any files in which links were changed; otherwise, visitors to your site won’t be able to see the changes.

Creating jump menus

A jump menu is a pop-up menu in a document, visible to your site visitors, listing options that link to documents or files. You can create links to documents in your website, links to documents on other websites, e-mail links, links to graphics, or links to any file type that can be opened in a browser.

A jump menu can contain three basic components:

- A menu selection prompt, such as a category description for the menu items, or instructions, such as “Choose one:” (Optional)

- A list of linked menu items: a user chooses an option and a linked document or file opens. (Required)
- A Go button. (Optional)

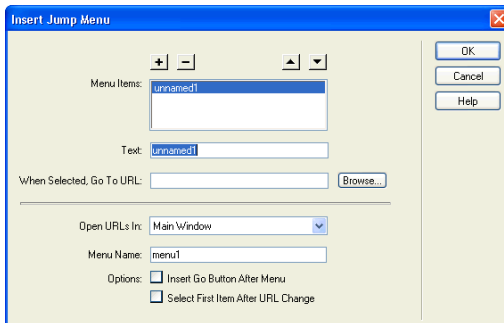
Inserting a jump menu

To insert a jump menu in your document, use the Jump Menu form object.

To create a jump menu:

- 1 Open a document, and then place the insertion point in the Document window.
- 2 Do one of the following:
 - Choose Insert > Form Object > Jump Menu.
 - Select the Form tab in the Insert bar, and then click the Jump Menu button.

The Insert Jump Menu dialog box appears.



- 3 Complete the dialog box.

For more information, click the Help button in the dialog box.

- 4 Click OK.

The jump menu appears in your document.

Editing jump menu items

To make changes to jump menu items, use the Property inspector or the Behaviors panel. You can change the list order or the file an item links to, or you can add, delete, or rename an item.

To change the location in which a linked file opens, or to add or change a menu selection prompt, you must use the Behaviors panel (see “Jump Menu” on page 361).

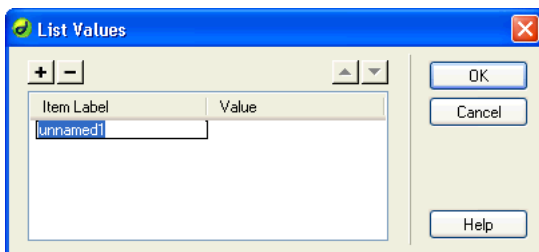
To edit a jump menu item using the Property inspector:

- 1 Choose Window > Properties to open the Property inspector, if it isn't open.
- 2 In the Document window's Design view, click the jump menu object to select it.

The List/Menu icon appears in the Property inspector.

- 3 In the Property inspector, click the List Values button.

The List Values dialog box appears.



- 4 Make changes to the menu items as necessary, and then click OK.

Troubleshooting jump menus

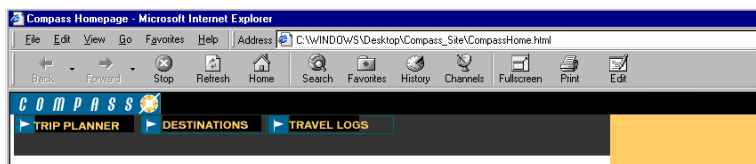
Once users choose a jump menu item, there is no way to reselect that menu item if they navigate back to that page, or if the Open URL In field specifies a frame. There are two ways to work around this problem:

- Use a menu selection prompt, such as a category, or a user instruction, such as “Choose one:”. The menu selection prompt is reselected automatically after each menu selection.
- Use a Go button, which allows a user to revisit the currently chosen link.

Note: Select only one of these options per jump menu, in the Insert Jump Menu dialog box, because they apply to an entire jump menu.

Creating navigation bars

A navigation bar consists of an image (or set of images) whose display changes based on the actions of a user. Navigation bars often provide an easy way to move between pages and files on a site.



Before using the Insert Navigation Bar command, create a set of images for the display states of each navigation element. (It can be helpful to think of a navigation bar element as a button, because when clicked, it takes the user to another page.)

A navigation bar element can have four states:

- Up: the image that appears when the user hasn't yet clicked or interacted with the element.

For example, the element in this state looks like it hasn't been clicked.



- **Over:** the image that appears when the pointer is moved over the Up image. The element's appearance changes (for example, it may get lighter) to let users know they can interact with it.



- **Down:** the image that appears after the element has been clicked.

For example, when a user clicks an element, a new page loads and the navigation bar is still displayed, but the clicked element is darkened to show that it's been selected.

- **Over While Down:** the image that appears when the pointer is rolled over the Down image after the element has been clicked.

For example, the element appears dimmed or gray. You can use this state as a visual clue to users that this element cannot be clicked again while they are in this part of the site.

You don't have to include navigation bar images for all four of these states; for example, you may just want Up and Down states.

You can create a navigation bar, copy it to other pages in your site, use it with frames, and edit the page behaviors to show different states as pages are accessed.

Inserting a navigation bar

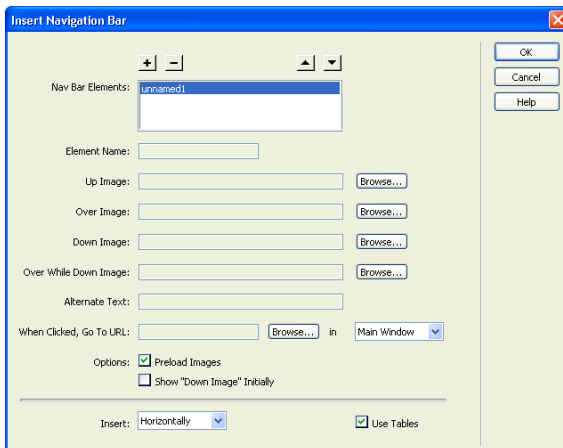
When you insert a navigation bar, you name the navigation bar elements and select images to use for them.

To create a navigation bar:

1 Do one of the following:

- Choose Insert > Interactive Images > Navigation Bar.
- Select the Common tab of the Insert bar, and then click the Insert Navigation Bar button.

The Insert Navigation Bar dialog box appears.



- 2 Complete the dialog box.

For more information, click the Help button in the dialog box.

- 3 Click OK.

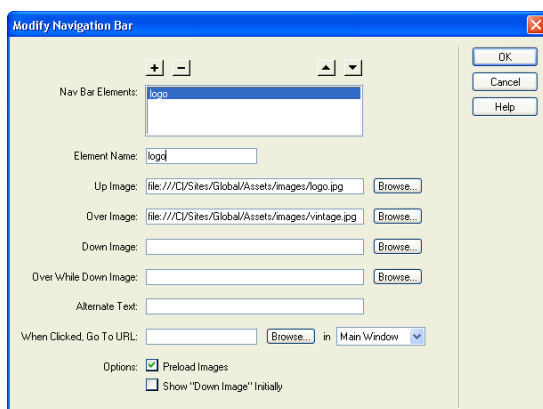
Modifying a navigation bar

Once you create a navigation bar for a document, you can add images to or remove them from the navigation bar by using the Modify Navigation Bar command. You can use this command to change an image or set of images, to change which file opens when an element is clicked, to select a different window or frame in which to open a file, and to reorder the images.

To modify a navigation bar:

- 1 Choose Modify > Navigation Bar.

The Modify Navigation Bar dialog box appears.



- 2 In the Nav Bar Elements list, select the element you want to edit.

- 3 Make changes as necessary.

For more information, click the Help button in the dialog box.

- 4 Click OK.

Creating image maps

An image map is an image that has been divided into regions, or “hotspots”; when a user clicks a hotspot, an action occurs (for example, a new file opens). Use the image Property inspector to graphically create and edit client-side image maps.

Client-side image maps store the hypertext link information in the HTML document—not in a separate map file as server-side image maps do. When a site visitor clicks a hotspot in the image, the associated URL is sent directly to the server. This makes client-side image maps faster than server-side image maps, because the server does not need to interpret where the visitor clicked. Client-side image maps are supported by Navigator 2.0 and later versions, NCSA Mosaic 2.1 and 3.0, and all versions of Internet Explorer.

Dreamweaver does not alter references to server-side image maps in existing documents; you can use both client-side image maps and server-side image maps in the same document. However, browsers that support both types of image maps give priority to client-side image maps. To include a server-side image map in a document, you must write the appropriate HTML code.

Inserting client-side image maps

When you insert a client-side image map, you create a hotspot area, and then define a link that opens when a user clicks the hotspot area.

Note: You can create multiple hotspot areas, but they are part of the same image map.

To create a client-side image map:

- 1 In the Document window, select the image.
- 2 In the Property inspector, click the expander arrow in the lower right corner to see all properties.
- 3 In the Map Name field, type a unique name for the image map.

Note: If you are using multiple image maps in the same document, make sure each map has a unique name.

- 4 To define the image map areas, do one of the following:
 - Select the circle tool and drag the pointer over the image to create a circular hotspot.
 - Select the rectangle tool and drag the pointer over the image to create a rectangular hotspot.
 - Select the polygon tool and define an irregularly shaped hotspot by clicking once for each corner point. Click the arrow tool to close the shape.

After you create the hotspot, the hotspot Property inspector appears.

- 5 Complete the hotspot Property inspector.

For more information, click the Help button in the Property inspector.

- 6 When you finish mapping the image, click a blank area in the document to change the Property inspector.

Modifying an image map

You can easily edit the hotspots you create in an image map. You can move a hotspot area, resize hotspots, or move a hotspot forward or back in layer.

You can also copy an image with hotspots from one document to another, or copy one or more hotspots from an image and paste them on another image; hotspots associated with the image are also copied to the new document.

To select multiple hotspots in an image map:

- 1 Use the pointer hotspot tool to select a hotspot.
- 2 Do one of the following:
 - Shift-click the other hotspots you want to select.
 - Press Control+A (Windows) or Command+A (Macintosh) to select all of the hotspots.

To move a hotspot:

- 1 Use the pointer hotspot tool to select the hotspot you want to move.
- 2 Do one of the following:
 - Drag the hotspot to a new area.
 - Use the Shift + arrow keys to move a hotspot by 10 pixels in the selected direction.
 - Use the arrow keys to move a hotspot by 1 pixel in the selected direction.

To resize a hotspot:

- 1 Use the pointer hotspot tool to select the hotspot you want to resize.
- 2 Drag a hotspot selector handle to change the size or shape of the hotspot.

Attaching behaviors to links

You can attach a behavior to any link in a document (see “Applying a behavior” on page 351). Consider using the following behaviors when you insert linked elements into documents:

Set Text of Status Bar determines the text of a message and display it in the status bar at the bottom left of the browser window. For example, you can use this action to describe the destination of a link in the status bar instead of showing the URL associated with it. (See “Set Text of Status Bar” on page 367.)

Open Browser Window opens a URL in a new window. You can specify the properties of the new window, including its size, attributes (whether it is resizable, has a menu bar, and so on), and name. (See “Open Browser Window” on page 362.)

Jump Menu edits a jump menu. You can change the menu list, specify a different linked file, or change the browser location in which the linked document opens. (See “Jump Menu” on page 361.)

Set Nav Bar Image changes how a navigation bar behaves. Use this behavior to customize how the images in a navigation bar display. For example, when the pointer is over part of the navigation bar, the display of other images in the navigation bar or in the document change. (See “Set Nav Bar Image” on page 365.)

CHAPTER 27

Managing Site Assets, Libraries, and Templates

As you develop websites, you accumulate a growing number of assets. In some cases, you might use the same assets across multiple sites, or perhaps you have a set of favored assets that you use in all your sites. In Macromedia Dreamweaver MX, you can use the Assets panel to help manage your site assets. The Assets panel lets you easily keep track of and preview several kinds of assets that you have stored in your site, such as images, movies, colors, scripts, and links. You can easily drag an asset from the Assets panel directly to your current document to insert it in a page.

The Assets panel also provides access to two special types of assets—libraries and templates. Library items and templates are linked assets: editing a library item or template updates all documents in which these assets have been applied. Use library items and templates when you want to reuse specific content or design elements throughout a site or across multiple sites. Library items are intended for individual design elements, such as a site's copyright information or a logo. Templates let you control a larger design area. A template author designs a page and defines which areas of the page can accept either design or content edits.

The following topics are covered in this chapter:

- “Using the Assets panel” on page 420
- “Managing the Assets panel” on page 428
- “Working with Library items” on page 430
- “Creating, managing, and editing library items” on page 431
- “About Dreamweaver templates” on page 435
- “Creating a Dreamweaver template” on page 440
- “About defining template regions” on page 436
- “Creating a template-based document” on page 449
- “Creating a nested template” on page 456
- “Editing and updating templates” on page 457
- “Exporting and importing template XML content” on page 459

Using the Assets panel

Assets covers a variety of elements that you store in a site, such as an image or movie file.

You can obtain assets from various sources. For example, you might create assets in an application such as Macromedia Fireworks or Macromedia Flash, receive them from a co-worker or client, copy them from a clip-art CD, or copy an image from a graphics web site. For more information about planning and gathering assets, see “About site planning and design” on page 55.

You can use the Assets panel in two ways: as an easily accessible list of the assets in your site (the Site list), or as a way to organize the assets that you use most frequently (the Favorites list). The Assets panel automatically puts your site’s assets in the Site list. You add whatever assets you want to the Favorites list.

Most of the Assets panel operations work the same in the Site list and in the Favorites list. There are a few tasks, however, that you can perform only in the Favorites list (see “Setting up a Favorites list of assets” on page 428).

Opening the Assets panel

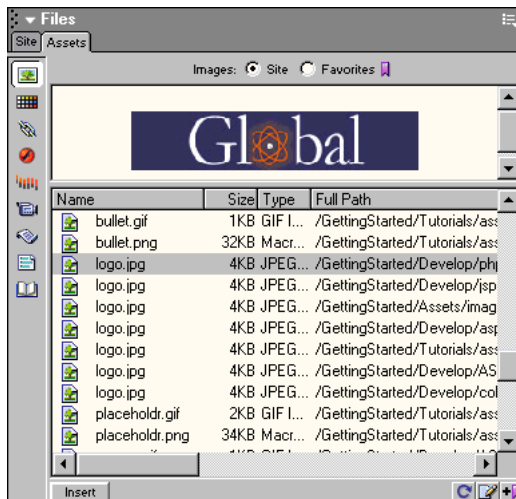
Use the Assets panel to view assets in the current site or to view assets that you’ve saved to a Favorites list.

You must define a local site before you can view assets in the Assets panel.

To open the Assets panel:

Choose **Window > Assets**.

The Assets panel appears. By default, the Images category is selected.



Selecting an assets category

The Assets panel helps you manage and organize your site's assets more easily than you can in the Site panel. Assets are categorized and display in a list.

To view assets in a particular category, click the appropriate icon. Dreamweaver displays the following categories of assets:



Images are image files in GIF, JPEG, or PNG formats. For information about images, see “Inserting Images” on page 297.



Colors are the colors that are used in documents and style sheets in your site, including text colors, background colors, and link colors.



URLs are the external links in your current site documents. This category includes the following types of links: FTP, gopher, HTTP, HTTPS, JavaScript, e-mail (`mailto`), and local file (`file://`).



Flash movies are files in any version of Macromedia Flash format. The Assets panel displays only SWF files (compressed Flash files), but not FLA (Flash source) files. See “About Flash content” on page 322.



Shockwave movies are files in any version of Macromedia Shockwave format. See “Inserting Shockwave movies” on page 326.



Movies are files in QuickTime or MPEG format. See “Inserting Media” on page 319.



Scripts are JavaScript or VBScript files. Note that scripts in HTML files (rather than in independent JavaScript or VBScript files) do not appear in the Assets panel. For information on working with JavaScript in Macromedia Dreamweaver, see “Using the JavaScript debugger” on page 209.



Templates provide an easy way to reuse the same page layout on multiple pages and to modify the layout on all pages attached to a template when a template is modified. See “About Dreamweaver templates” on page 435.



Library items are elements that you use in multiple pages; when you modify a library item, all pages that contain that item are updated. See “Working with Library items” on page 430

Note: Only files that fit into these categories show in the Assets panel. There are other types of files that are sometimes called assets, but they aren't shown in the panel.

Viewing assets in the Assets panel

The Assets panel provides two ways to view assets: the Site list, which shows all of the assets in your site, and the Favorites list, which shows only the assets you've explicitly chosen.

In both lists, assets are divided into categories: You select which category of assets to list by clicking one of the category buttons on the left side of the panel. The Site list and the Favorites list are both available for all categories of assets except templates and library items.

The Site list displays all assets that exist as files in your site, as well as all colors and URLs that are used in any document in your site.

Most of the following procedures can be performed in either the Site list or the Favorites list.

To view the Site list:

Click Site at the top of the Assets panel.

Note: In the Templates and Library categories, the Site and Favorites choices are unavailable.

To view the Favorites list:

Click Favorites at the top of the Assets panel.

The Favorites list is empty until you explicitly add assets to it.

To display assets in a particular category:

Click the appropriate category icon, and click either Site or Favorites. For example, to display all of the images in your site, click the Images icon and then click Site.

To preview an asset:

Select the asset in the Assets panel.

The preview area at the top of the panel shows a visual preview of the asset. For example, when you select a movie asset, the preview area shows an icon. To view the movie, click the Play button (the green triangle) in the upper right corner of the preview area.

To change the size of the preview area:

Drag the splitter bar (between the preview area and the list of assets) up or down.

Refreshing the Assets panel


It can take a few seconds for the Assets panel to create the Site list, because it has to read the site cache to create the list.

Certain changes don't appear immediately in the Assets panel:

- When you add or remove an asset from your site, the changes won't appear in the Assets panel until you refresh the Site list by clicking the Refresh Site List button. If you add or remove an asset outside of Dreamweaver—using Windows Explorer or the Finder, for example—you must rebuild the site cache to update the Assets panel.
- When you remove the only instance of a particular color or URL in your site, or when you save a new file that contains a color or URL that isn't already used in the site, the changes won't appear in the Assets panel until you refresh the Site list.

To manually refresh the Site list:

1 Click Site on the Assets panel to see the Site list

 2 Click the Refresh Site List button at the bottom of the Assets panel.

The site cache is created or updated as necessary, and the Assets panel updates to show the site assets.

To manually rebuild the site cache and refresh the Site list:

Control-click (Windows) or Command-click (Macintosh) in the Assets Panel List, then select Refresh Site List.

Adding an asset to a page

You can insert most kinds of assets into a document by dragging them into the Code view or Design view in the Document window, or by using the Insert button. You can either insert colors and URLs or apply them to selected text in the Design view. (URLs can also be applied to other elements in the Design view, such as images.) Templates are applied to the entire document.

To insert an asset in a document:


- 1 Place the insertion point in the Design view where you want the asset to appear.
- 2 Choose Window > Assets to open the Assets panel (if it's not already open).
- 3 Select the asset category for the type of asset you want to insert.
Select any category except Templates. A template can be applied only to an entire document; it can't be inserted into a document.
- 4 Select either Site or Favorites, and locate the desired asset.
There are no Site or Favorites lists for library items; skip this step if you're inserting a library item.
- 5 Do one of the following:
 - Drag the asset from the panel to the document. You can drag scripts into the head content area of the Document window; if that area isn't showing, choose View > Head Content.
 - Select the asset in the panel and click Insert.
The asset is inserted into the document. (If the asset is a color, it applies starting at the insertion point; that is, subsequent typing appears in that color.)

Using the Assets panel to apply a color

The colors in the Assets panel represent colors you've applied to various elements in your site, such as text, table border, background color, and so on. You can use the color swatches in the Color category to consistently apply your color choices to objects on a page.

For information about adding a color to the Color category, see “Creating a new color or URL to add to the Favorites list” on page 429.


To change the color of selected text in a document:

- 1 Select the text in the document.
-  2 Select the Colors category in the Assets panel.
- 3 Select the desired color.
- 4 Click Apply.


Using the Assets panel to add a URL link in a document

You can use the Assets panel to insert a URL link in a document or to apply a URL link to selected text.

To add a link to the current selection in a document:

- 1 Select the text or image where you want to apply the URL.
-  2 Select the URLs category in the Assets panel, and locate the desired URL.
- 3 Do one of the following:
 - Drag the URL from the panel to the selection in the Design view.
 - Select the URL, and click the Insert button.

To add a link in a document:

- 1 Place the insertion point in the document where you want to add the URL.
-  2 Select the URLs category in the Assets panel, and locate the desired URL.
- 3 Do one of the following:
 - Drag the URL from the panel to the selection in the Design view.
 - Select the URL, and click the Insert button.

Selecting and editing assets

The Assets panel allows you to select multiple assets at once. It also provides a quick way to begin editing assets.

To select multiple assets:

Select one of the assets by clicking it. Then select the other assets in one of the following ways:

- Shift-click to select a consecutive series of assets.
- Control-click (Windows) or Command-click (Macintosh) to add an individual asset to the selection (whether or not it's adjacent to the existing selection). Control-click or Command-click a selected asset to deselect it.

To edit an asset:

- 1 Do one of the following:
 - Double-click the asset.
 - Select the asset, then click Edit.

For some kinds of assets, such as images, editing the asset starts an external editing application. For colors and URLs, editing the asset lets you change the asset's value in the Favorites list only. (You can't edit colors and URLs in the Site list.) For templates and library items, editing the asset allows you to make changes to the asset within Dreamweaver.

If an external editor doesn't open for an asset that's supposed to use one, choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X), select the File Types/Editors category, and make sure you have an external editor defined for the asset's file type. See "Launching an external editor for media files" on page 320.

- 2 Change the asset as desired.
- 3 When you're done editing the asset, do one of the following:
 - If the asset is a file-based asset (anything other than a color or URL), save it (using whatever editor you used to edit it) and close it.
 - If the asset is a URL, click OK when you're finished editing in the Edit URL dialog box.
If the asset is a color, the Dreamweaver color picker is dismissed automatically when you pick a color. (To dismiss the color picker without picking a color, press Escape.)

Using the Library category of the Assets panel

The Library category of the Assets panel displays all of the items in the current site's library. For information about changing or updating a library item see "Editing a library item" on page 432.

To create a new empty library item in the Assets panel:

- 1 Make sure nothing is selected in the Document window.
If something is selected, it will be placed in the new library item.
- 2 Click the New Library Item button at the bottom of the Assets panel.
A new, untitled library item is added to the list in the panel.
- 3 While the item is still selected, enter a name for it.

To add a library item in a document:

- 1 In the Document window, place the pointer where you want to insert the library item.
- 2 Select the Library category in the Assets panel, and locate the desired library item.
- 3 Do one of the following:
 - Drag the library item from the panel to the selection in the Design view.
 - Select the library item, and click the Insert button.

To edit a Library item:

Select a library item from the list and click the Edit button, or double-click the item's name in the list.

To delete a library item:

- 1 Select the library item in the Assets panel.
- 2 Click the Delete button or press the Delete key, and confirm that you want to delete the library item.
The library item is deleted from your site.

Related topic

"Using the Assets panel" on page 420

Using the Assets panel to work with templates

You can use the Templates category of the Assets panel to create new templates and manage existing templates.

For more details about creating and editing templates, see “Creating a Dreamweaver template” on page 440. For more information about using the Assets panel to manage your assets, see “Using the Assets panel” on page 420.

To use the Assets panel to create a new, blank template

- 1 Choose Window > Templates.

The Assets panel appears, with the Templates category selected.



- 2 In the Assets panel, click the New Template button at the bottom of the Assets panel.

A new, untitled template is added to the list of templates in the panel.

- 3 While the template is still selected, enter a name for the template.
- 4 This creates a new blank template in the Assets panel and in the Templates folder but does not create a .dwt file. To start defining editable regions, click the Edit button at the bottom of the Assets panel.

For information about defining editable template regions see “Inserting an editable region” on page 442.

To edit a template file:

- 1 Choose Window > Assets, then click the Templates icon to view templates.

The bottom of the Assets panel lists all of the available templates for your site. The top area displays a preview of the selected template.

- 2 In the Name list, do one of the following:

- Double-click the name of the template.
- Click the Edit Template button at the bottom of the Assets panel.



- 3 Edit the template in the Document window. Create editable regions in the template as desired (see “About defining template regions” on page 436).
- 4 Save the edited template by choosing File > Save.

Note: If you haven’t defined any editable regions, Dreamweaver warns you that the template contains no editable regions. You can save or modify the template even if it doesn’t contain editable regions, but you won’t be able to modify documents based on that template until it has an editable region.

To apply a template to the active document:

- 1 Open the document you’re applying the template to.



- 2 Select the Templates category in the Assets panel, and locate the template you want to apply.

- 3 Do one of the following:

- Drag the template from the panel to the Document window.
- Select the template and click the Apply button.

Note: You must assign the existing content in the document to an editable region of the template, or Dreamweaver discards the content.

To rename a template in the Assets panel:

- 1 Click the name of the template to select it.
- 2 Click the name again so that the text is selectable, then enter a new name.

Note: This method of renaming works in the same way as renaming a file in Windows Explorer (Windows) or the Finder (Macintosh). As with Windows Explorer and the Finder, be sure to pause briefly between clicks. Do not double-click the name; because that opens the template for editing.

- 3 Click in another area of the Asset panel, or press Enter (Windows) or Return (Macintosh) for the change to take affect.
- 4 Dreamweaver asks if you want to update documents that are based on this template.
 - To update all documents in the site that are based on this template, click Update.
 - To not update any documents that are based on this template, click Don't Update. A document that you choose not to update is still based on the original template (not detached). To update this document later, open the document and choose Modify > Templates > Update.

To delete a template file:

- 1 Select the template in the Assets panel.
- 2 Click the Delete button and confirm that you want to delete the template.

If you delete a template file, you can't retrieve it. The template file is deleted from your site. Documents that were based on this template are not detached from the template; they retain the structure and editable regions that the template file had before it was deleted.

To turn such a document into a normal HTML file without editable or locked regions, use Modify > Templates > Detach from Template. For more information, see “Detaching a document from a template” on page 457.

Working with assets and sites

You may want to locate the file in the Site panel that corresponds to an asset in the Assets panel. This can be useful, for example, when you want to transfer the asset to or from your remote site.

The Assets panel shows all the assets (of recognized types) in your current site. (The current site is the site that contains the active document.) To use an asset from the current site in another site, you must copy it to the other site. You can copy an individual asset, a set of individual assets, or an entire Favorites folder at once.

Note: The Site panel may show a different site from the one that the Assets panel shows. This is because the Assets panel is associated with the active document. To determine which site the Assets panel is showing, look at the panel's title bar.

To locate an asset's file in the Site panel:

Right-click (Windows) or Control-click (Macintosh) the asset's name or icon in the Assets panel, and choose Locate in Site from the context menu.

The Site panel appears, with the asset file selected.

Locate in Site is unavailable for colors and URLs, which do not correspond to files in the site.

Note that Locate in Site locates the file corresponding to the asset itself; it does not locate a file that uses that asset.

To copy assets from the Site list or the Favorites list to another site:

- 1 Right-click (Windows) or Control-click (Macintosh) the icon or name of the asset or assets to copy.
In the Favorites list, you can copy a Favorites folder as well as individual assets. (For information about Favorites folders, see “Grouping assets in a Favorites folder” on page 430.)

- 2 From the context menu, choose Copy to Site, and choose the target site name from the submenu. (The submenu lists all the sites you’ve defined.)

The assets are copied to the specified site, into locations corresponding to their locations in the current site. Dreamweaver creates new folders in the target site’s hierarchy as needed.

The assets are also added to the specified site’s Favorites list.

When you open a document in the target site, the Assets panel switches to that site, and shows the copied asset.

Note: If the asset you copied is a color or a URL, it appears only in the other site’s Favorites list, not in the other site’s Site list. This is because there is no file corresponding to the color or URL, so there’s no file to copy into the other site.

Managing the Assets panel

By default, assets in a given category are listed in alphabetical order by name. The asset name may be a filename (as for images), a hexadecimal number (as for colors), or a URL. You can instead list the assets by any of several other criteria.

To list assets in a different order:

Click one of the column headings.

For example, to sort the list of images by type (so that all the GIF images are together, all the JPEG images are together, and so on), click the Type column heading.

To change the width of a column:

- 1 Position the pointer over the line that separates two column headings.
- 2 Drag that line to change the column width.

Setting up a Favorites list of assets

Most of the operations of the Assets panel are the same in the Favorites list as in the Site list (see “Using the Assets panel” on page 420). However, there are several tasks that you can perform only in the Favorites list.

Because the Assets panel’s Site list always shows all of the recognized assets in the site, this list can become cumbersome for some large sites. If you add frequently used assets to your Favorites list, you can group related assets together, give them nicknames to remind you what they’re for, and find them easily in the Assets panel.

Note: Favorite assets are not stored as separate files on your disk; they’re references to the assets in the Site list. Dreamweaver keeps track of which assets from the Site list to display in the Favorites list.

Adding assets to and removing assets from the Favorites list

There are several ways to add assets to your site's Favorites list.

To add assets to your Favorites list, do one of the following:



- Select one or more assets in the Site list, and click the Add to Favorites button.
- Select one or more assets in the Site list, and choose Add to Favorites from the context menu.
- Right-click (Windows) or Control-click (Macintosh) an element in the Document window's Design view, and choose the context menu command to add the element to the appropriate Favorites category.

For example, you can select an image and choose Add to Image Favorites from the context menu. Note that the context menu for text contains either Add to Color Favorites or Add to URL Favorites, depending on whether the text has a link attached. Also note that only elements that fit one of the categories in the Assets panel can be added to the Favorites list.

- Select one or more files in the Site panel and choose Add to Favorites from the context menu. Files that don't fit a category in the Assets panel are ignored.

Whichever method you choose, the assets are added to your Favorites list, and the Assets panel displays the Favorites list.

Note: There are no Favorites lists for templates and library items.

To remove assets from your Favorites list:

- 1 Select one or more assets (or a Favorites folder) in the Favorites list.



- 2 Click the Remove From Favorites button at the bottom of the Assets panel.

The assets are removed from the Favorites list, but they still appear in the Site list. If you remove a Favorites folder, the folder and all of the assets in it are removed from the Favorites list.

Creating a new color or URL to add to the Favorites list

You can use the Assets panel to add new colors or URLs to your Favorites list. Note that you can't add new colors or URLs to the Site list; the Site list contains only assets that are already in use in your site.

To create a new color:



- 1 Click the Colors category button.
- 2 Click Favorites to show the Favorites list.



- 3 Click the New Color button.

- 4 Select a color using the color picker.

To exit from the color picker without choosing a color, press Escape or click the gray bar at the top of the color picker. (For more information on using the color picker, see "Working with colors" on page 114.)

- 5 Give the color a nickname if desired.

To create a new URL:



1 Click the URLs category button.

2 Click Favorites to show the Favorites list.



3 Click the New URL button.

4 Enter a URL and a nickname in the Add URL dialog box, and click OK.

Creating a nickname for a favorite asset

You can give nicknames to assets in the Favorites list. The nickname is displayed instead of the asset's filename or value. For example, if you have a color named #999900, you might use a more descriptive nickname, such as PageBackgroundColor or ImportantTextColor.

You can give nicknames only to assets in the Favorites list. In the Site list, the assets are listed by their real filenames (or values, in the case of colors and URLs).

To give a nickname to a favorite asset:

1 Click the category that contains your asset.

2 Click Favorites to display the Favorites list.

3 Do one of the following:

- Right-click (Windows) or Control-click (Macintosh) the asset's name or icon in the Assets panel, and choose Edit Nickname from the context menu.
 - Click the asset's name once, pause, then click it again. (Don't double-click.)
- 4 Type a nickname for the asset, then press Enter (Windows) or Return (Macintosh).

The nickname appears in the Nickname column.

Grouping assets in a Favorites folder

In the Favorites list, within a given category, you can create named groups of assets, called Favorites folders. For example, if you have a set of images that you use on numerous catalog pages in an e-commerce site, you could group them together in a folder called CatalogImages.

Note: Placing an asset in a Favorites folder does not change the location of the asset's file on your disk.

To create a Favorites folder:

1 Click Favorites (at the top of the Assets panel) to display the Favorites list.



2 Click the New Favorites Folder button.

3 Name the folder.

4 Drag assets into the folder.

Working with Library items

A library is a special Dreamweaver file that contains a collection of individual “assets” or copies of assets you have created for placement in your web pages. If you simply want your pages to have the same headers and footers, yet have different page layouts, use library items to store the headers and footers. Library items are stored page elements that you can reuse in multiple pages; you can update all the pages that use a library item whenever you change the item's contents.

You can store all sorts of page elements, such as images, tables, sounds, and Flash movies in a library.

When you use a library item, Dreamweaver doesn't insert the library item in the web page, rather it inserts a link to the library item. If you later need to make changes to a library item, for example change some text, or an image, updating the library item automatically updates the instance of the library in any page in which you've inserted the library item.

Related topics

“Creating a library item” on page 431

“Inserting a library item in a document” on page 432

“Editing a library item” on page 432

“Making library items editable in a document” on page 434

Creating, managing, and editing library items

Libraries are a way to store page elements such as images, text, and other objects that you want to reuse or update frequently throughout your web site. These elements are called library items.

When you place a library item in a document, Dreamweaver inserts a copy of the HTML source code for that item into the document, and adds an HTML comment containing a reference to the original, external item. The reference to the external library item makes it possible to update the content on an entire site all at once by changing the library item and then using the update commands in the Modify > Library submenu.

Dreamweaver stores library items in a Library folder within the local root folder for each site. Each site has its own library; to copy a library item from one site to another site, use the Assets panel's Copy to Site command. For more information, see “Using the Library category of the Assets panel” on page 425.

Note: If the library item contains links, the links may not work in the new site. Also, images in a library item aren't copied to the new site.

Here's an example of how you might use a library item: suppose you're building a large site for a company. The company has a slogan that it wants to appear on every page of the site, but the marketing department is still finalizing the text of the slogan. If you create a library item to contain the slogan and use that library item on every page, then when the marketing department provides the final slogan, you can change the library item and automatically update every page that uses it.

Library items can also contain behaviors, but there are special requirements for editing the behaviors in library items; see “Editing a behavior in a library item” on page 434. Library items cannot contain timelines or style sheets, because the code for those elements is part of the head section.

Creating a library item

You can create a library item from any element in the body section of a document, including text, tables, forms, Java applets, plug-ins, ActiveX elements, navigation bars, and images.

For linked items such as images, the library stores only a reference to the item. The original file must remain at the specified location for the library item to work correctly. It can still be useful to store an image in a library item, though; for example, you could store a complete `img` tag in a library item, which would allow you to easily change the image's `alt` text, or even its `src` attribute, throughout the site. (Don't use this technique to change an image's `width` and `height` attributes, though, unless you also use an image editor to change the actual size of the image.)

To create a library item:

- 1 Select a portion of a document to save as a library item.
- 2 Do one of the following:
 - Choose Window > Library and drag the selection into the Library category of the Assets panel.
 - Click the New Library Item button at the bottom of the Assets panel (in the Library category).
 - Choose Modify > Library > Add Object to Library.
- 3 Enter a name for the new library item.

Each library item is saved as a separate file (with the file extension .lbi) in the Library folder of the site's local root folder.

Inserting a library item in a document

When you add a library item to a page, the actual content is inserted in the document along with a reference to the library item.

To insert a library item in a document:

- 1 Place the insertion point in the Document window.
- 2 Choose Window > Library.
The Assets panel appears, showing the Library category.
- 3 Drag a library item from the Assets panel to the Document window, or select an item and click the Insert button.

To insert the contents of a library item without including a reference to the item in the document, press Control (Windows) or Option (Macintosh) while dragging an item out of the Library category of the Assets panel. If you insert an item this way, you can edit the item in the document, but the document won't update when you update pages that use that library item.

Editing a library item

When you change a library item, you can choose to update all documents that use that item. If you choose not to update, the documents remain associated with the library item; you can update them later by choosing Modify > Library > Update Pages.

Other kinds of changes to library items include renaming items to break their connection with documents or templates, deleting items from the site's library, and recreating a missing library item.

Note: The CSS Styles panel and Timelines panel are unavailable when you are editing a library item, since library items can contain only body elements, and Timeline and CSS style sheet code inserts in the head section of a document. The Page Properties dialog box is also unavailable, because a library item can't include a body tag or its attributes.

To edit a library item:

- 1 Choose Window > Library.
The Library category of the Assets panel appears.
- 2 Select a library item.

A preview of the library item appears at the top of the Assets panel. (You can't edit anything in the preview.)

- 3 Click the Edit button at the bottom of the Assets panel. Alternatively, double-click the library item.

Dreamweaver opens a new window for editing the library item. This window is much like a Document window, but its Design view has a gray background to indicate that you're editing a library item instead of a document.

- 4 Edit the library item and then save your changes.
- 5 In the dialog box that appears, choose whether to update the documents on the local site that use the edited library item:
 - Choose Update to update all documents in the local site with the edited library item.
 - Choose Don't Update to avoid changing any documents until you use Modify > Library > Update Current Page or Update Pages.

To update the current document to use the current version of all library items:

Choose Modify > Library > Update Current Page.

To update the entire site or all documents that use a particular library item:

- 1 Choose Modify > Library > Update Pages.

The Update Pages dialog box appears.

- 2 In the Look In pop-up menu, do one of the following:
 - Choose Entire Site, and then choose the site name from the adjacent pop-up menu. This updates all pages in the selected site to use the current version of all library items.
 - Choose Files That Use, and then choose a library item name from the adjacent pop-up menu. This updates all pages in the current site that use the selected library item.
- 3 Make sure Library Items is selected in the Update option. (To update templates at the same time, make sure Templates is also selected. For more information, see "Updating documents that are based on a template" on page 457.)
- 4 Click Start.

Dreamweaver updates the files as indicated. If you selected the Show Log option, Dreamweaver provides information about the files it attempts to update, including information on whether they were updated successfully.

To rename a library item in the Assets panel:

- 1 Click the name of the library item once to select it.
- 2 After a brief pause, click again.
- 3 When the name becomes editable, enter a new name.

Note: This method of renaming works the same way that renaming a file in Windows Explorer (Windows) or the Finder (Macintosh) does. As with Windows Explorer and the Finder, make sure to pause briefly between clicks. Do not double-click the name; that opens the library item for editing.

- 4 Click elsewhere, or press Enter (Windows) or Return (Macintosh).

- 5 Dreamweaver asks if you want to update documents that use the item.
 - To update all documents in the site that use the item, click Update.
 - To refrain from updating any documents that use the item, click Don't Update.

To delete a library item from a library:

- 1 Select the item in the Library category of the Assets panel.
- 2 Click the Delete button and confirm that you want to delete the item.

Be careful; if you delete a library item, you can't use Undo to retrieve it. You may be able to recreate it, however, as described in the next procedure.

Deleting a library item removes the item from the library, but doesn't change the contents of any documents that use the item.

To recreate a missing or deleted library item:

- 1 Select an instance of the item in one of your documents.
- 2 Click the Recreate button in the Property inspector.

Making library items editable in a document

If you've added a library item to your document and you want to edit the item specifically for that page, you must break the link between the item in the document and the library. Once you've made an instance of a library item editable, that instance cannot be updated when the library item changes.

To make a library item editable:

- 1 Select a library item in the current document.
- 2 Click Detach from Original in the Property inspector.

The selected instance of the library item loses its highlighting (if you had highlighting visible), and can no longer be updated when the original library item changes.

Editing a behavior in a library item

When you create a library item that includes an element with a Dreamweaver behavior attached to it, Dreamweaver copies the element and its event handler (the attribute that specifies which event triggers the action (such as `onClick`, `onLoad`, or `onMouseOver`), and which action to call when the event occurs) to the library item file. Dreamweaver does not copy the associated JavaScript functions into the library item. Instead, when you insert the library item into a document, Dreamweaver automatically inserts the appropriate JavaScript functions into the `head` section of that document (if they aren't already there).

Note: If you hand-code JavaScript (that is, if you create it without using Dreamweaver behaviors), you can make it part of a library item if you use the Call JavaScript behavior to execute the code. If you don't use a Dreamweaver behavior to execute the code, the code isn't retained as part of the library item.

To edit a behavior in a library item, you must first insert the item in a document, then make the item editable in that document. After you've made the desired changes, you can re-create the library item, replacing the item in the library with the edited item from your document.

For more information about behaviors, see "Using JavaScript Behaviors" on page 349.

To edit a behavior in a library item:

- 1 Open a document that contains the library item.
Note the name of the library item, as well as the exact tags it contains. You'll need this information in steps 8 and 9.
- 2 Select the library item and click Detach from Original in the Property inspector.
- 3 Select the element that has the behavior attached to it.
- 4 Choose Window > Behaviors to open the Behaviors panel. In the Behaviors panel, double-click the action you want to change.
- 5 In the dialog box that appears, make the necessary changes and click OK.
- 6 Choose Window > Library to open the Library category of the Assets panel.
- 7 Make sure you have recorded the exact name of the original library item; then select the original library item, and delete it by clicking the Delete button in the Assets panel.
- 8 In the Document window, select all of the elements that make up the library item.
Be careful to select exactly the same elements that were in the original library item.
- 9 In the Assets panel, click the New Library Item button, and give the new item the same name as the item you deleted in step 7.
Be sure to use exactly the same spelling and capitalization.
- 10 To update the library item in your site's other documents, choose Modify > Library > Update Pages.
- 11 In the Update Pages dialog box, in the Look in pop-up menu, select Files That Use.
- 12 In the adjacent pop-up menu, choose the name of the library item you just created.
- 13 In the Update option, make sure Library Items is selected, and then click Start.
- 14 When the updates are complete, click Close to exit the Update Pages dialog box.

About Dreamweaver templates

A Dreamweaver template is a special type of document that you use to design a “locked” page layout. A template author designs the page layout, and creates regions in the template that are editable in documents that are based on a template. In a template, the designer controls which page elements a template user— such as writers, graphic artists, or other web developers— can edit. For information about authoring a template see “Creating a Dreamweaver template” on page 440. For information about creating a page based on a template, see “Creating a template-based document” on page 449.

One of the most powerful uses of templates is the ability to update multiple pages at once. A document that is created from a template remains connected to that template (unless you detach the document later). You can modify a template and immediately update the design in all document based on it.

A template controls the editable and fixed regions of a document based on a template. There are several types of template regions that you can include in a document. For information about these types of template regions, see “About defining template regions” on page 436.

You can add a Cascading Style Sheet (CSS), timeline, or behavior to a document that is based on a template, because Dreamweaver automatically inserts an editable region in the head section of a document.

Template-related operations (such as adding editable regions) are enabled in both Code view and Design view. Some template customizing options are only available in Code view. For information about adding template expressions, see *Writing a template expression*.

If you want to store additional information about a template (such as who created it, when it was last changed, or why you made certain layout decisions), you can create a Design Notes file for the template. Documents based on a template do not inherit the template's Design Notes. (For more information, see "Saving file information in Design Notes" on page 99.)

About defining template regions

A template determines the basic structure for a document and contains elements such as text, images, page layout, styles, and editable regions.

Dreamweaver automatically locks most regions of a document when you save the document as a template. As a template author you define which regions of a template-based document will be editable by inserting editable regions or editable parameters in the template. As you create the template, you can make changes to both editable and locked regions. In a document based on the template, however, a template user can make changes only in the editable regions; the locked regions can't be modified.

There are four types of template regions: editable regions, repeating regions, optional regions, and editable tag attributes.

An editable region is an unlocked region in a template-based document; a section a template user can edit. A template author can define any area of a template as editable. For a template to be effective, it should contain at least one editable region; otherwise, pages based on the template can't be edited. For information about inserting an editable region, see "Inserting an editable region" on page 442.

A repeating region is a section in a document that is set to repeat. For example, you can repeat a table row. By repeating the table row, you allow the template user to create an expanding list, while keeping the design under the control of the template author. In a document based on the template you use repeat region control options to add or delete copies of the repeated region. There are two types of repeating regions you can insert in a template—repeating region and repeating table. For information about inserting a repeating region in a template, see "Creating a repeating region in a template" on page 444. For information about creating a repeating table, see "Inserting a repeating table" on page 444. For information about working with a repeating region in a template-based page, see "Adding, deleting, and changing the order of a repeating region entry" on page 453.

An optional region is a section of a template that a designer defines as optional, to hold content such as optional text or an image which may or may not appear in a document based on the template. In the template-based page, the content editor usually controls whether the content shows or not. For information about setting optional regions in a template, see "Inserting an optional region" on page 448. For information about editing optional regions in a template-based page, see "Modifying optional region template parameters" on page 452.

An **editable tag attribute** lets you unlock a tag attribute in a template, so the attribute can be edited in a template-based page. For example, you can “lock” which image appears in the document but let the page author set the alignment to left, right, or center. For information about setting editable tag attributes, see “Defining editable tag attributes in a template” on page 446. For information about editing the tags in a template-based page, see “Modifying template properties” on page 450.

About template parameters

You can use template parameters to define optional regions, editable tag attributes, or to set values you want to pass to an attached document.

Creating a template parameter lets you define values for controlling content in documents based on a template. For each parameter you define a name, a data type, and a default value. Each parameter must have a unique name and is case sensitive.

Template parameters are passed to the document as instance parameters, and can be accessed by the Modify > Template Properties command. In most cases, a template user can edit the parameter’s default value to customize what appears in a template-based document. In other cases, the template author might use a computed-text expression to determine what appears in the document, based on the value in the expression. For information, see Writing a template expression in Dreamweaver Help.

Editing code outside of HTML tags

Some server scripts are inserted at the very beginning or end of the document (before the `<html>` tag or after the `</html>` tag). Such scripts require special treatment in templates and template-based documents. Normally, if you make changes to script code before the `<html>` tag or after the `</html>` tag in a template, the changes will not be copied to documents based on that template. This can cause server errors if other server scripts, within the main body of the template, depended on the scripts that were not copied. As a result, Dreamweaver warns you if you make a change to scripts before the `<html>` tag or after the `</html>` tag in a template.

To avoid this problem, you can insert the following code in the head section of the template:

```
<!-- TemplateInfo codeOutsideHTMLIsLocked="true" -->
```

When this code is present in a template, changes to scripts before the `<html>` tag or after the `</html>` tag will be copied to documents based on that template. However, you will no longer be allowed to edit those scripts in documents based on the template. Thus, you can choose to be able to edit these scripts in the template, or in documents based on the template, but not both.

About links in templates

To create a link in a template file, use the folder icon or the Point-to-File icon in the Property inspector; don’t type in the name of the file to link to. If you type the name, the link might not work. This section explains how Dreamweaver handles links in templates.

When you create a template file from an existing page, then save that page as a template, Dreamweaver updates the links so they point to the same files as before. Because templates are saved in the Templates folder, the path for a document-relative link changes when you save the page as a template. In Dreamweaver, when you create a new document based on that template and save the new document, all the document-relative links are updated to continue to point to the correct files.

When you add a new document-relative link to a template file, however, if you type the path into the link text box in the Property inspector, it's easy to enter the wrong path name. The correct path is the path from the Templates folder to the linked document, not the path from the template-based document's folder to the linked document.

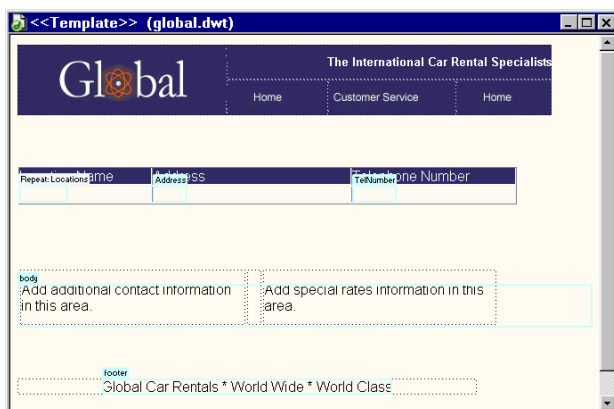
For information about linking using the Point-to-File icon, see “Linking files and documents” on page 403.

Note: In some cases, (such as file paths in event handlers in templates) you can't use the folder icon or the Point-to-File icon; in those cases, you must enter the correct path name.

Viewing templates in Design view

In template files and documents based on templates, editable regions appear in the Document window's Design view surrounded by rectangular outlines in a preset highlight color. (For information about setting highlighting color preferences, see Setting template highlighting preferences in Dreamweaver Help.) A small tab appears at the upper left corner of each region, showing the name of each defined region. Template regions also include a highlighted outline. You can make changes to both editable and locked content.

Template files can be identified by the <<Template>> title and .dwt file extension in the Document window.



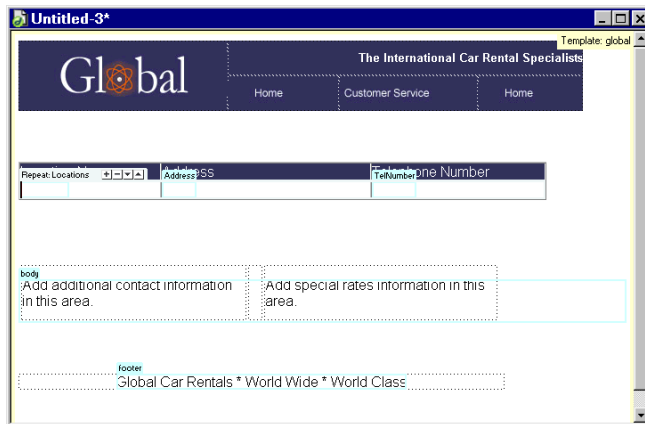
Viewing templates in Code view

In Code view, you can make changes to both editable and locked HTML source code in a template. Editable content is marked in HTML with the comments `<!-- TemplateBeginEditable>` and `<!-- TemplateEndEditable -->`. Everything between these comments is editable in documents based on the template. The HTML source code for the editable region from the previous example might look like this:

```
<table width="75%" border="1" cellspacing="0" cellpadding="0">
  <tr bgcolor="#333366">
    <td>Name</td>
    <td><font color="#FFFFFF">Address</font></td>
    <td><font color="#FFFFFF">Telephone Number</font></td>
  </tr>
  <!-- TemplateBeginEditable name="LocationList" -->
  <tr>
    <td>Enter name</td>
    <td>Enter Address</td>
    <td>Enter Telephone</td>
  </tr>
  <!-- TemplateEndEditable -->
</table>
```

Viewing template-based documents in Design view

In a document based on a template (a template-based document), in addition to the editable-region outlines, the entire page is surrounded by a different-colored outline, with a tab at the upper right giving the name of the template that the document is based on. This highlighted rectangle is there to remind you that the document is based on a template and that you can't change anything outside the editable regions.



Note: If you try to edit a locked region in a document based on a template when highlighting is turned off, the mouse pointer changes to indicate that you can't click in a locked region.

Viewing template-based documents in Code view

In Code view, editable regions of a document derived from a template are displayed in a different color than code in the non-editable regions; you can make changes only to code in the editable regions or editable parameters.

Editable content is marked in HTML with the Dreamweaver comments

`<!-- InstanceBeginEditable >` and `<!-- InstanceEndEditable -->`. Everything between these comments is editable.

The HTML source code for an editable region might look like this:

```
<!-- InstanceBeginEditable name="head" -->
<!-- InstanceEndEditable -->
</head>

<body bgcolor="#FFFFFF" leftmargin="0">
<table width="75%" border="1" cellspacing="0" cellpadding="0">
  <tr bgcolor="#333366">
    <td>Name</td>
    <td><font color="#FFFFFF">Address</font></td>
    <td><font color="#FFFFFF">Telephone Number</font></td>
  </tr>
  <!-- InstanceBeginEditable name="LocationList" -->
  <tr>
    <td>Enter name</td>
    <td>Enter Address</td>
    <td>Enter Telephone</td>
  </tr>
  <!-- InstanceEndEditable -->
</table>
</body>
<!-- InstanceEnd -->
```

The default setting for non-editable text is gray, you can define a different color for the editable and non-editable regions in the Preferences dialog box. Choose **Edit > Preferences** or **Dreamweaver > Preferences** (Mac OS X), and select **Code Coloring**. For more information, see **Setting code color preferences for a template document in Dreamweaver Help**.

Note: Dreamweaver prevents you from typing in locked regions in the Code view of the Document window.

Creating a Dreamweaver template

You can create a template from an existing document (such as an HTML, Macromedia ColdFusion, or Microsoft Active Server Pages document) or you can create a template from a new, blank document.

The New Document dialog box (**File > New**) provides several options for creating new templates, and creating template-based pages. For information, see “Working with the New Document dialog box” on page 108.

Dreamweaver saves templates with the file extension `.dwt`. Templates are saved in a special **Templates** folder in the local root folder of the site. If the **Templates** folder does not already exist in the site, Dreamweaver automatically creates it when you save a new template.

Note: Do not move your templates out of the **Templates** folder, or put any non-template files in the **Templates** folder. Also, do not move the **Templates** folder out of your local root folder. Doing so causes errors in paths in the templates.

Related topics

“Saving a document as a template” on page 441

“Inserting an editable region” on page 442

“Creating a nested template” on page 456

Saving a document as a template

You can create a template from a new document or from an existing document.

If you insert a template region in a document that hasn't been saved as a template, Dreamweaver advises you that the document will automatically be saved as a template.

To create a template:

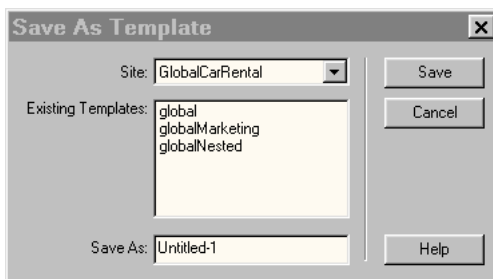
1 Open the document you want to save as a template in the Document window, by doing one of the following:

- To open an existing document, choose File > Open and select the document.
- To open a new blank document, choose File > New. In the dialog box that opens, choose Basic Pages or Dynamic pages. In the document list select the type of page you want to work with, then click Create.

2 When the document opens, choose File > Save as Template.

Note: Unless you selected Don't Show This Dialog Again, you'll receive a warning that says the document you're saving has no editable regions. Click OK to save the document as a template, or click Cancel to exit this dialog box without creating a template.

3 In the dialog box that appears, select a site to save the template in from the Site pop-up menu, and enter a unique name for the template in the Save As text box.



4 Click Save.

The template file is saved in the site's Templates folder, with a .dwt filename extension.

To create a new template using the Assets panel:

1 Open the Assets panel (Window > Assets or F11), if it isn't already open.

2 Select the Templates category.

3 Click the New Template button in the Assets panel.

A new document is added to the Name list. The document name is highlighted.

4 Type a name for the document.

- 5 Click the Edit button in the Assets panel.

A template document opens in the Document window.

For information about creating editable regions in the template see “Inserting an editable region” on page 442.

Creating editable regions

Editable template regions control which areas of a template-based page can be edited.

You can mark an entire table or an individual table cell as editable, but you can't mark multiple table cells as a single editable region. If `<td>` is selected the editable region includes the region around the cell; if not, the editable region affects content inside the cell.

Layers and layer content are separate elements; making a layer editable lets you change the position of the layer as well as its contents, and making a layer's contents editable lets you change only the content of the layer, not its position. To select the layer's content, click in the layer and choose Edit > Select All. To select the layer, make sure invisible elements are showing, and click the icon that represents the location of the layer. For more information about layers, see “Animating Layers” on page 377.

Inserting an editable region

Before you insert an editable region, you should save the document you are working in as a template. If you insert an editable region in a document rather than a template file, Dreamweaver warns you that the document will automatically be saved as a template.

To define an editable template region:

- 1 In the Document window, do one of the following to select the region:

- Select the text or content that you want to set as an editable region.
- Place the insertion point where you want to insert an editable region.

- 2 Do one of the following to insert an editable region:

- Choose Insert > Template Objects > New Editable Region.
- Right-click (Windows) or Control-click (Macintosh) the selected text or object, and choose New Editable Region from the context menu.
- In the Insert bar, select Templates, then click the Editable Region button.

The Editable Region dialog box appears.

- 3 In the Name text box, enter a unique name for the region. (You cannot use the same name for more than one editable region in a particular template.)

Note: Do not use special characters in the Name field.

- 4 Click OK.

The editable region is enclosed in a highlighted rectangular outline in the template, using the highlighting color that is set in preferences. For information about setting template highlighting options, see Setting template highlighting preferences in Dreamweaver Help.

A tab at the upper-left corner of the region shows the name of the region. If you inserted an empty editable region in the document, the name of the region also appears inside the region.

Removing an editable region

If you've marked a region of your template file as editable and you want to lock it (make it noneditable) again, use the Remove Template Markup command.

Note: You can't lock a region in a file based on a template; you can lock a region only in a template file.

To remove an editable region:

- 1 In the document or in the tag selector, select the editable region you want to change.
- 2 Choose Modify > Templates > Remove Template Markup.

The region is now no longer editable.

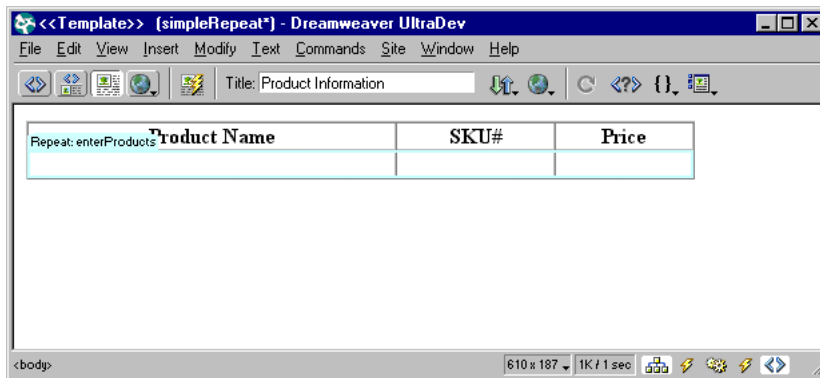
Creating repeating regions

A repeating region adds multiple copies of the selected region in a template-based document. You can use repeating regions to control the layout of regions you want repeated in a page such as a catalogue item and description layout, or to repeat a row for data such as a list of items.

There are two repeating region template objects you can use:

Repeating Region lets you repeat a region but doesn't include an editable region. However, you can insert an editable region in a repeating region to make it editable.

The example below shows a repeating region in a template. In a document based on this template, the user is able to repeat the highlighted table row to expand the table.



Repeating Table lets you define a table and define where editable regions are in each cell in the repeating region. You define options to control which rows are included in the repeating region.

Related topics

“Creating a repeating region in a template” on page 444

“Inserting a repeating table” on page 444

“Adding, deleting, and changing the order of a repeating region entry” on page 453

Creating a repeating region in a template

A repeating region is a section of a template that can be duplicated as often as desired in a template-based page. Repeating regions are typically used with tables; however, you can define a repeating region for other page elements as well.

A repeating region is not an editable region. To make content in a repeating region editable (for example, to allow a user to enter text in a table cell), you must insert an editable region in the repeating region. For information about creating editable regions, see “Inserting an editable region” on page 442. For information about creating an editable repeating table, see “Inserting a repeating table” on page 444.

To insert a repeating region in a template:

- 1 In the Document window, do one of the following:
 - Select the text or content you want to set as a repeating region.
 - Place the insertion point in the document where you want to insert the repeating region.
- 2 Do one of the following, to create a repeating region:
 - Choose Insert > Template Objects > Repeating Region.
 - Right-click (Windows) or Control-click (Macintosh) the selected content, and choose New Repeating Region from the context menu.
 - In the Templates category of the Insert bar, click the Repeating Region button.The Repeating Region dialog box appears.
- 3 In the Name text box, enter a unique name for the template region. (You cannot use the same name for more than one repeating region in a template.)

Note: When you name a region, do not use special characters.

- 4 Click OK.

The repeating region is inserted in the document.

Note: A repeating region isn't editable in the template-based document, unless it contains an editable region. For information about inserting an editable region see “Inserting an editable region” on page 442.

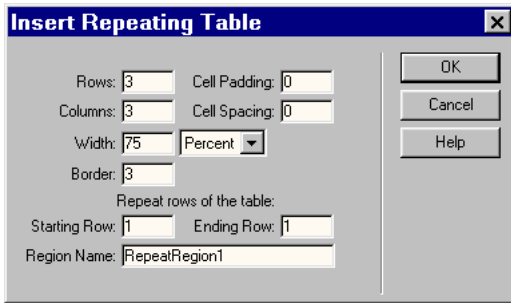
Inserting a repeating table

You use a repeating table to define a repeating region that includes editable regions in a table format. You can define table attributes and set which table cells are editable.

To insert a repeating table:

- 1 In the Document window, place the insertion point in the document where you want to insert the repeating table.
- 2 Do one of the following:
 - Choose Insert > Template Objects > New Repeating Table.
 - Right-click (Windows) or Control-click (Macintosh) the selected content, and choose New Repeating Table from the context menu.

- In the Templates category of the Insert bar, click the Repeating Table button. The Repeating Table dialog box appears.



- 3 Enter new values as desired.
For more information, click the Help button in the dialog box.
- 4 Click OK.

Setting alternating background colors in a repeating region

You can customize a repeating table by alternating the background color of the table rows.

To set alternating table row background colors:

- 1 In the Document window, select the repeat region table row.
- 2 Click the Show Code view or Show Code and Design view button in the Document toolbar so you can access the code for the selected table row.
- 3 In Code view, edit the `<tr>` tag to include the following code:

```
<tr bgcolor="@@(_index & 1) ? '#FFFFFF' : '#CCCCCC' @@">
```

You can replace the `#FFFFFF` and `#CCCCCC` hexadecimal values with other color choices.

This is a code example of a table that includes alternating background table row colors:

```
<table width="75%" border="1" cellspacing="0" cellpadding="0">
<tr><th>Name</th><th>Phone Number</th><th>Email Address</th></tr>
<!-- TemplateBeginRepeat name="contacts" -->
<tr bgcolor="@@(_index & 1) ? '#FFFFFF' : '#CCCCCC' @@">
<td><!-- TemplateBeginEditable name="name" -->name<!-- TemplateEndEditable -->
</td>
<td><!-- TemplateBeginEditable name="phone" -->phone<!-- TemplateEndEditable -->
</td>
<td><!-- TemplateBeginEditable name="email" -->email<!-- TemplateEndEditable -->
</td>
</tr>
<!-- TemplateEndRepeat -->
</table>
```

Defining editable tag attributes

You use the Make Attribute Editable command to allow a template user to modify specified tag attributes in a document created from a template. For example, you can set a background color in the template document yet allow a template user to set a different background color for pages they create. The user can update only the attributes you define as editable.

Creating an editable tag attribute inserts a template parameter in the code. An initial value for the attribute is set in the template document; when a template-based document is created, it inherits the parameter. A template user can then edit the parameter in the document.

The following data types are supported: text, boolean (*true/false*), color, and URL. For information about setting editable tag attributes, see “Defining editable tag attributes in a template” on page 446.

For information about editing the tag in the template-based document, see “Modifying template properties” on page 450.

Defining editable tag attributes in a template

You can set multiple editable attributes for a page element. Only tag attributes that are currently set for the element initially appear in the Attribute pop-up menu. For example, with an image tag the width and height attributes may be the only ones that appear in the menu.

If the attribute that you want to make editable isn’t listed in the Attribute pop-up menu, click Add to add the attribute to the list.

To define an editable tag attribute:

- 1 In the Document window, select an item you want to set an editable tag attribute for.
- 2 Choose Modify > Templates > Make Attribute Editable.
- 3 In the dialog box that appears, do one of the following:
 - If the attribute you want to make editable is listed in the Attribute pop-up menu, select it.
 - If the attribute you want to make editable isn’t listed in the Attribute pop-up menu, click Add, and in the dialog box that opens, enter the name of the attribute you want to add, and click OK.
- 4 Make sure the Make Attribute Editable option is selected.

In the dialog box complete the fields to set the values you want.

- 5 In the Label text box, enter a unique name for the attribute.

Tip: To make it easier to identify a specific editable tag attribute later, use a label that identifies the element and the attribute. For example, you might label an image whose source is editable `logoSrc` or label the editable background color of a `body` tag `bodyBgcolor`.

- 6 In the Type pop-up menu, select the type of value allowed for this attribute by setting one of the following options:
 - To let a user enter a text value for the attribute, choose Text. For example, you can use text with the `align` attribute; the user can then set the attribute’s value to *left*, *right*, or *center*.
 - To set the link to an element, such as the file path to an image, choose URL. Using URL allows Dreamweaver to automatically update the path used in a link. If the user moves the image to a new folder, the Update Links dialog appears.
 - To make the color picker available for selecting a value, choose Color.

- To let a template user type a numerical value to update an attribute (for example, to change the height or width values of an image), choose Number.
- 7 The Default Value field displays the value of the selected tag attribute in the template. Enter a new value in this field to set a different initial value for the parameter in the template-based document.
- 8 If you want to make changes to another attribute of the selected tag, repeat steps 3 through 7.
- 9 Click OK.

Making an editable tag attribute uneditable

A tag previously marked as editable can be marked as uneditable. Make the change in the template and update the pages derived from the template.

To reset an editable tag attribute:

- 1 In the template document, click the element associated with the editable attribute or use the tag selector to select the tag.
- 2 Choose Modify > Templates > Make Attribute Editable.
- 3 In the Attributes pop-up menu, select the attribute you want to affect.
- 4 Deselect the Make Attribute Editable checkbox.
- 5 Click OK.

About optional regions

An optional region is a region in a template that can be set to show or to hide in a template-based document. Use an optional region when you want to set conditions for displaying content in a document. You can set specific values for a template parameter or define conditional statements in a template. Based on the conditions you define a template users can edit the parameters in documents they create and control whether the optional region displays or not.

You can create template parameters and expressions in the Optional Region dialog box or directly in Code view. When you use the Optional Region template object, Dreamweaver inserts template comments in the code.

A template parameter is defined in the head section:

```
<!-- TemplateParam name="departmentImage" type="boolean" value="true" -->
```

At the location where the optional region is inserted, code similar to the code below appears:

```
<!-- TemplateBeginIf cond="departmentImage" -->
<p> </p>
<!-- TemplateEndIf -->
```

Template parameters you create for optional regions can be accessed and edited in the template-based document by using the Template Properties dialog box (Modify > Template Properties).

There are two optional region objects: one lets the template author control which regions show and hide without letting a user edit the content, while the other lets template users edit content and set whether the region shows or hides.

Related topics

“Inserting an optional region” on page 448

“Modifying template properties” on page 450

Inserting an optional region

Use an optional region to control content which may or may not be shown in a template-based document. An optional region is controlled by a conditional statement. The template tab of an optional region is preceded by the word *if*. Based on the condition set in the template, a template user can define whether the region is viewable in pages they create.

An editable optional region allows a template user to edit the content in an optional region. For example, if the optional region includes an image or text, the template user can set whether the content is displayed, as well as make edits to the content if desired. Editable regions are controlled by a conditional statement. You create template parameters and expressions in the Optional Region dialog box, or by typing parameters and conditional statements in Code view.

To insert an optional region:

- 1 In the Document window, select the element you want to set as an optional region.
- 2 Do one of the following:
 - In the Insert bar, select the Templates category, then select the Optional Region button.
 - Choose Insert > Template Objects > Optional Region.
 - Right-click the selected content, and in the context menu select New Optional Region.
- 3 Define options for the optional region.

For information about defining the optional region, click the Help button in the dialog box.
- 4 Click OK.

To insert an editable optional region:

- 1 In the Document window, position the pointer where you want the optional region inserted.

Tip: You cannot wrap a selection to create an editable optional region. Insert the region, then insert the content in the region.
- 2 Do one of the following to open the Optional Region dialog box:
 - In the Insert bar, select the Templates category, then select the Editable Optional Region button.
 - Choose Insert > Template Objects > Editable Optional Region.
 - Right-click the selected content, and in the context menu select Editable Optional Region.

The Optional Region dialog box appears.
- 3 Define options for the optional region.

For information about defining the optional region, click the Help button in the dialog box.
- 4 Click OK.

Modifying an Optional Region

You can edit optional region settings after you've inserted the region in a template. For example, you can change whether the default for the content is set to show or not, link a parameter to an existing optional region, or modify a template expression.

To reopen the Optional Region dialog box:

- 1 Open the Property inspector, if it isn't already open.
- 2 In the Document window, do one of the following:
 - In Design view, click the template tab of the editable region you want to modify.
 - In Design view, click content in the template region, then in the tag selector click the template tag, `<mmtemplate:if>`.
 - In Code view, click the comment tag of the template region you want to modify.
- 3 In the Property inspector, click Edit.
The Optional Region dialog box opens.
- 4 Make the changes you want, then click OK.

Creating a template-based document

Once you set the template design, you apply a template to either a blank document or a document that already contains content.

To create a new document based on a template, you can use the Assets panel or the New Document dialog box. The Assets panel lists templates in your current Dreamweaver site. You can right-click in the Templates category of the Assets panel to create a new template, and to create new documents from a template.

In the New Document dialog box, you can select a template from any of your currently defined Dreamweaver sites to create a new document from. When you create a new document based on a template, you can choose whether the document remains attached to the template. By default, when you change a template, all attached documents update to reflect the template changes. If you deselect the Update Page When Template Changes option when creating a new document, the new document is created as stationery: an independent HTML file, with no template regions. Updates to the template do not change the document.

To create a template-based document in the New Document dialog box:

- 1 Choose File > New to open the New Document dialog box.
- 2 In the New Document dialog box, select the Templates tab.
- 3 In the Templates For list, select the site which contains the template you want to use.
The site templates list updates to display templates in the selected site.
- 4 In the list, select the template you want to use.
- 5 Click Create to create a new template-based page.
The document opens in the Document window.

To create a new document from a template in the Assets panel:

- 1 Open the Assets panel (F11), if it is not already open.
- 2 In the Assets panel, click the Templates icon to view your site templates.
- 3 Right-click the template you want to apply, then select New From Template.

Tip: If you just created the template you want to apply, you may need to click the Refresh button to see it.

The document opens in the Document window.

Editing content in a template-based page

You can easily identify and select template regions in both the template document and in template-based documents. Editable template regions are listed at the bottom of the Modify > Templates submenu.

Editable regions that are inside a repeated region do not appear in the menu. You must locate these regions by looking for tabbed borders in the Document window.

For information about locating and modifying editable tags in a template-based document, see “Modifying template properties” on page 450.

To find an editable region and select it in the document:

Choose Modify > Templates, and choose the name of the region from the list at the bottom of that submenu.

The editable region is selected in the document.

Modifying template properties

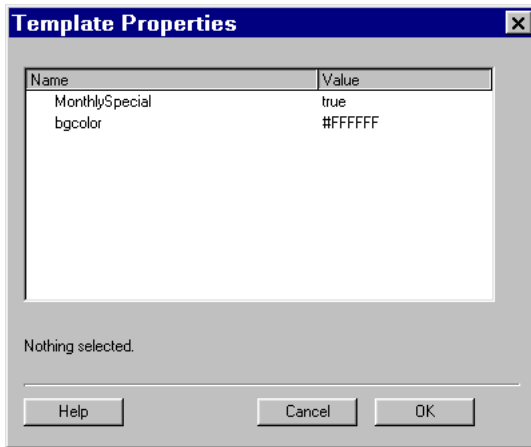
When you create parameters in a template, documents based on the template automatically inherit the parameters and their initial value settings. A template user can update editable tag attributes and other template parameters (such as optional region settings) in the Template Properties dialog box.

For information about creating an editable tag in a template see “Defining editable tag attributes in a template” on page 446. For information about creating optional regions, see “Inserting an optional region” on page 448.

To modify an editable tag attribute:

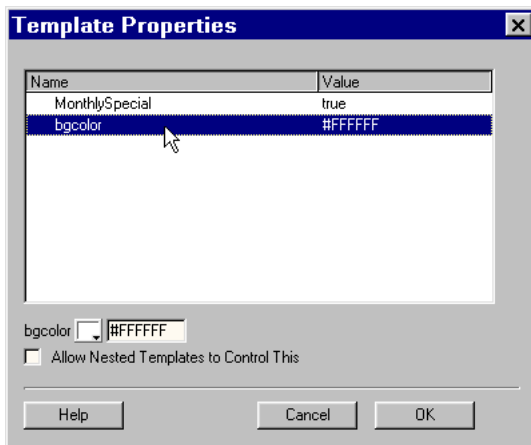
- 1 Open the template-based document, then choose Modify > Template Properties.

The Template Properties dialog box opens. The following example shows two template properties. One is an optional region, the other an editable tag attribute for the template's background color.



- 2 In the Name list, select a the property.

The bottom area of the dialog box updates to show the selected property's label and its assigned value.



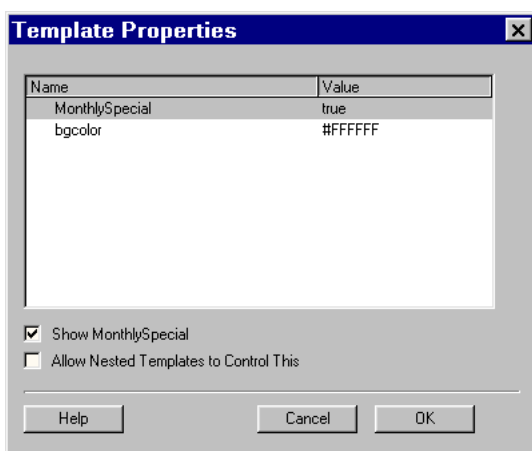
- 3 In the field to the right of the property label, edit the value to modify the property in the document.

Note: The field name and updateable values are defined in the template. Attributes which do not appear in the Name list are not updateable in the document.

- 4 If you are defining an editable tag attribute in a nested template, do one of the following to set whether the parameter is passed through to documents based on the nested template:
 - Check the Allow Nested Templates to Control This checkbox, to pass the editable property to documents based on the nested template; the parameter will appear in the Templates Properties dialog box for documents based on that nested template.
 - To prevent the parameter from passing through do not check this option.

Modifying optional region template parameters

- 1 Open the template-based document, then choose Modify > Template Properties.
The Template Properties dialog box opens.
- 2 In the Name list, select a property.
The dialog box updates to show the selected property's label and its assigned value.



- 3 Select the Show checkbox, to show the optional region in the document, or deselect the checkbox to hide it.
Note: The field name and default setting are defined in the template..
- 4 Check the Allow Nested Templates to Control This checkbox, to pass the editable property along to a documents based on the nested template.

Adding, deleting, and changing the order of a repeating region entry

Use repeating region controls to add, delete, or change the order of entries. When you add a repeating region entry, a copy of the entire repeating region is added. In order to update the content in the repeating regions, the original template must include an editable region in the repeating region.

Template: simpleRepeat		
Product Name	SKU#	Price
Repeat: enterProduct updateProducts Macadamia nuts	Mac3423	12.00 lb.
Brazil nuts updateProducts	Bra9302	9.00 lb.

Options for editing repeating regions are also available in the Modify > Templates menu and the Edit > Repeating Entries menu.

To add, delete, or change the order of a repeating region:

Place the insertion point in the repeating region to select it, then do one of the following:

- Click the plus (+) button to add a repeating region entry below the currently selected entry.
- Click the minus (-) button to delete the selected repeating region entry.
- Click the Down Arrow button to move the selected entry down one position.
- Click the Up Arrow button to move the selected entry up one position.
- Choose Modify > Template > Repeating Entries, and select one of the options in the menu. You can use this menu to insert a new repeating entry or move the selected entry's position.

To cut, copy, and delete entries:

Place the insertion point in the repeating region to select it, then do one of the following:

- Choose Edit > Repeating Entries > Cut Repeating Entries to cut a repeating entry.
- Choose Edit > Repeating Entries > Copy Repeating Entries to copy a repeating entry.
- Choose Edit > Repeating Entries > Delete Repeating Entries to remove a repeating entry.
- Choose Edit > Paste to paste a repeating entry.

Pasting inserts a new entry; it does not replace an existing entry.

About nested templates

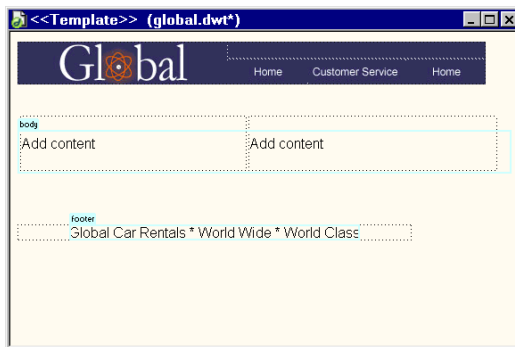
A nested template is a template whose design and editable regions are based on another template. To create a nested template, you must first save the original or base template, then create a new document based on the template, and then save that document as a template. In the new template, you can further define editable regions in areas originally defined as editable from the base template.

Nested templates are useful for controlling content in pages of a site that share many design elements, but have a few variations between pages. For example, a base template might contain broader design areas and be usable by many content contributors for a site, while a nested template might further define the editable regions in pages for a specific section in a site.

Editable regions in a base template are passed through to the nested template, and remain editable in pages created from a nested template unless new template regions are inserted in these regions.

Changes to a base template are automatically updated in templates based on the base template, and in all template-based documents that are based on the main and nested templates.

Example: The Global Car Rental site's base template contains two editable regions, named `body` and `footer`:



To create a nested template, we created a new document based on the template, then saved the document as a template and named it *globalNested*. In the nested template, we made changes in the editable region named *body*. We inserted a graphic in the left column and inserted a new editable region in the right column.



When you add a new editable region in an editable region passed through to the nested template, the highlighting color of the editable region changes to orange. Content you add in an editable region, such as the graphic in the left column, is no longer editable in documents based on the nested template. The blue highlighted editable areas, whether added in the nested template or passed through from the base template, remain editable in documents based on the nested template. Template regions you do not insert an editable region in, pass through to template-based documents as editable regions.



Creating a nested template

You can create a nested template by saving a document based on a template, then creating a new template of that document. Nested templates let you create variations of a base template. You can nest multiple templates to define increasingly specific layout.

To create a nested template:

- 1 Create a document from the template you want to base the nested template on by doing one of the following:
 - In the Assets panel's Templates category, right-click the template you want to create a new document from, then select **New From Template**.
 - Choose **File > New**. In the **New Document** dialog box, click the **Templates** tab and select the site that contains the template you want to use; then in the document list, double-click the template to create a new document.

A new document appears in the Document window.

- 2 Choose **File > Save as Template** or click the **Make Nested Template** button in the Insert bar, to save the new document as a template
- 3 In the new template, add additional content and editable regions.
- 4 Save the template.

In documents based on the nested template, you can add or change content in editable regions passed through from the base template, as well, as editable regions created in the new template.

Applying a template to an existing document

When you apply a template to a document which contains existing content, Dreamweaver attempts to match the existing content to a region in the template. If you are applying a revised version of one of your existing templates, the names are likely to match. Dreamweaver compares the names of the editable regions in the document to the names of the editable regions in the new template; for each region name that matches, Dreamweaver places the existing contents of that region into the new region with the same name.

If you apply a template to a document that hasn't had a template applied to it, there are no editable regions to compare and a mismatch occurs.

Dreamweaver tracks these mismatches and lists them in the **Inconsistent Region Names** dialog box. You can then select which region or regions to move the current page's content to, or you can choose to delete the mismatched content. If you want to keep the content, select the name of an editable region from the pop-up menu in the dialog box; otherwise, choose "nowhere", which discards the unresolved content.

To apply a template to an existing document:

- 1 Choose **File > Open** to open the document you want to apply the template to.
- 2 Perform one of the following actions:
 - Click in the Document window, then choose **Modify > Templates > Apply Template to Page**. Choose a template from the list and click **Select**.
 - Select the template in the Templates category of the Assets panel, and click the **Apply** button.
 - Drag the template from the Templates category of the Assets panel to the Document window's Design view.

- 3 If content exists in the document that can't be automatically assigned to a template region, the Inconsistent Region Names dialog box appears. It lists all the editable regions in the template being applied. Use it to choose a destination for the content.

For information about moving existing content to editable regions in the document, see *Resolving inconsistent region names in Dreamweaver Help*.

Detaching a document from a template

To make changes to the locked regions of a document based on a template, you must detach the document from the template. When the document is detached, the whole document becomes editable.

To detach a document from a template:

- 1 Open the template-based document you want to detach.
- 2 Choose **Modify > Templates > Detach from Template**.

The document is detached from the template and all template code is removed.

Editing and updating templates

When you save a template, Dreamweaver asks if it should update all documents attached to the template. You can use the **Modify > Templates > Update Pages** command. Dreamweaver updates template-based documents in the same site as the template. Dreamweaver will automatically update documents based on the template, including nested templates that are based on a template you're updating.

Updating Dreamweaver 4 templates

You can continue to work with templates that you created in Dreamweaver 4 in the current version of Dreamweaver. Dreamweaver will not automatically change the template syntax in the Dreamweaver 4 template files, so these files can be used with both Dreamweaver 4 and Dreamweaver MX. However, if you insert a new template region in a Dreamweaver 4 template using Dreamweaver MX, it will use the new Dreamweaver MX template syntax, and you will no longer be able to use that template in Dreamweaver 4.

Updating documents that are based on a template

When you make a change to a template, Dreamweaver prompts you to update the documents based on the template. You can also use the update commands to manually update the current document or the entire site. Using the update commands is the same as reapplying the template.

To open and modify an attached template:

- 1 With the template-based document in the Document window, select **Modify > Templates > Open Attached Template**.
- 2 Modify the content of the template as desired. To modify the template's page properties, choose **Modify > Page Properties**. (Documents based on a template inherit the template's page properties.)
- 3 Save your document. Dreamweaver prompts you to update pages based on the template.

To apply template changes to the document:

Select **Modify > Templates > Update Current Page**.

To update the entire site or all documents that use the attached template:

1 Select Modify > Templates > Update Pages.

The Update Pages dialog box appears.

2 In the Look In pop-up menu, do one of the following:

- Select Entire Site, and then choose the site name from the adjacent pop-up menu. This updates all pages in the selected site to their corresponding templates.
- Select Files That Use, and then choose the template name from the adjacent pop-up menu. This updates all pages in the current site that use the selected template.

3 Make sure Templates is selected in the Update option.

4 Click Start.

Dreamweaver updates the files as indicated. If you selected the Show Log option, Dreamweaver provides information about the files it attempts to update, including information on whether they were updated successfully.

Checking template syntax

Dreamweaver automatically checks the template syntax when you save a template. An error message dialog appears if the syntax is badly formed. The error message describes the error and refers to the specific line in the code where the error exists.

You can use the Check Template Syntax command to check the template syntax prior to saving a template. For example, if you manually add a template parameter or expression in Code view, use the command to check that the code follows correct syntax.

To check for valid template syntax, do the following:

Choose Modify > Templates > Check Template Syntax.

About XML

Extensible Markup Language (XML) is a markup language for structured documents. You can think of it as being like a generalization of HTML that lets you define tags. XML is derived from the Standard Generalized Markup Language (SGML). Like SGML, XML is a language for defining sets of tags and the relationships among them.

Tags in XML look similar to HTML tags, consisting of a tag name plus optional attributes, surrounded by angle brackets. As with HTML, content is contained between an opening tag and a closing tag (in which the tag name is preceded by a slash). One syntactical difference is that in XML, an empty tag (one that uses only a single tag, such as ``, instead of an opening tag and a closing tag containing text or other material) must end with a slash just before the closing angle bracket. For example, an `` tag in XML might look like the following example:

```

```

XML is more strict about its syntax than HTML; files that do not strictly conform to XML syntax are rejected by XML parsers.

XML provides a way to mark literal strings, which can include tags that should not be parsed. This method involves an element called a CDATA section, which includes a literal string in square brackets. Anything inside the square brackets is skipped by the XML parser.

Dreamweaver-exported XML uses CDATA sections extensively.

Exporting and importing template XML content

A document based on a template contains a set of data represented by name/value pairs. Each pair consists of the name of an editable region, and the contents of that region. Suppose your template contains three editable regions: `item_number`, `item_price`, and `item_color`. You could fully describe a document based on that template by providing the name of the template and the name/value pairs.

Dreamweaver lets you export the name/value pairs into an XML file so that you can work with the data outside of Dreamweaver (for example, in an XML editor or a text editor, or a database application). Conversely, if you have an XML document that's structured appropriately, you can import the data from it into a document based on a Dreamweaver template. For more information, see "About XML" on page 458.

To export a document's editable regions as XML:

- 1 Choose File > Open, and open a document that's based on a template and contains editable regions.
- 2 Choose File > Export > Template Data as XML.
- 3 In the Export Template Data as XML dialog box, under Notation, select one of the following options:
 - If the document contains repeating regions or template parameters, select Use Standard Dreamweaver XML tag.
 - For templates that do not contain repeating regions or template parameters, select Use Editable Region Names as XML Tags.
- 4 Click OK.
- 5 In the dialog box that appears, enter a name for the XML file and click Save.

Dreamweaver generates an XML file that contains the material from the document's parameters and editable regions, including editable regions inside repeating regions or optional regions. The XML file includes the name of the original template, as well as the name and contents of each template region.

Note: Content in the non-editable regions is not exported to the XML file.

To import XML content:

- 1 Choose File > Import > Import XML into Template.
- 2 In the Import XML dialog box, select the XML file and click Open.

Dreamweaver creates a new document based on the template specified in the XML file. It fills in the contents of each editable region in that document using the data from the XML file. The resulting document appears in a new Document window.

Note that if your XML file isn't set up exactly the way Dreamweaver expects, you may not be able to import your data. One solution to this problem is to export a dummy XML file from Dreamweaver, so that you'll have an XML file with exactly the right structure. Then copy the data from your original XML file into the exported XML file. The result is an XML file with the correct structure that contains the appropriate data, all ready to be imported.

Using XML tag notations

Dreamweaver lets you export XML content using standard Dreamweaver XML tags.

The easiest way to see the tag notations, if you understand XML syntax, is to look at an example. The following XML code was exported from a document based on a template named `newstemplate`. The document contains one defined editable region, named `News_Story` (and a default editable region, `doctitle`, which contains the document's title tag).

Standard Dreamweaver XML tags use a tag named `item`. The `item` tag has a `name` attribute that gives the name of the editable region; the tag contains a `CDATA` section that in turn contains the contents of the editable region. In this example, the `item` tag with `name="doctitle"` identifies the title of the document, and the `item` tag with `name="News_Story"` identifies the editable region.

```
<?xml version="1.0"?>
<templateItems template="/Templates/newstemplate.dwt">
  <item name="doctitle">
    <![CDATA[<title>Today's Headline Story</title>]]>
  </item>
  <item name="News_Story">
    <![CDATA[<p>This is where the story goes.</p>]]>
  </item>
</templateItems>
```

CHAPTER 28

Testing a Site

Before uploading your site to a server and declaring it ready for viewing, it's a good idea to test it locally. (In fact, it's a good idea to test and troubleshoot your site frequently throughout its construction—you can catch problems early and avoid repeating them.)

You should make sure that your pages look and work as expected in the browsers you're targeting, that there are no broken links, and that the pages don't take too long to download. You can also test and troubleshoot your entire site by running a site report.

The following guidelines will help you create a good experience for visitors to your site:

- Make sure your pages function as expected in the browsers you're targeting and that they either work or “fail gracefully” in other browsers.

Your pages should be legible and functional in browsers that do not support styles, layers, plugins, or JavaScript (see “Checking for browser compatibility” on page 462). For pages that fail badly in older browsers, consider using the Check Browser behavior to automatically redirect visitors to another page (see “Check Browser” on page 355).

- Preview your pages in as many different browsers and on as many different platforms as possible. This gives you an opportunity to see differences in layout, color, font sizes, and default browser window size that cannot be predicted in a target browser check (see “Previewing pages in browsers” on page 463).
- Check your site for broken links, and then fix them.

Other sites undergo redesign and reorganization too, and the page you're linking to may have been moved or deleted. You can run a link check report to test your links (see “Checking links in a page or site” on page 465, “Fixing broken links” on page 466, “Opening linked documents in Dreamweaver” on page 467, or “Using Reports to test a site” on page 468).

- Monitor the file size of your pages and the time they take to download (see “Setting download time and size” on page 467).

Keep in mind that for pages that consist of one large table, visitors will see nothing until the entire table finishes loading. Consider breaking up large tables; if this is not possible, consider putting a small amount of content—such as a welcome message or an advertising banner—outside the table at the top of the page so users can view this material while the table downloads.

- Run a few site reports to test and troubleshoot the entire site.

You can check your entire site for problems, such as untitled documents, empty tags, and redundant nested tags (see “Using Reports to test a site” on page 468).

- Once the bulk of the site has been published, continue to update and maintain it.
Publishing your site—that is, pushing it out and making it live—can be accomplished in several ways and is an ongoing process. An important part of the process is defining and implementing a version-control system, either with the tools Macromedia Dreamweaver MX includes or through an external version-control application.
- Use the Macromedia Dreamweaver discussion forums found on the Macromedia website. For information on accessing the discussion groups, see http://www.macromedia.com/go/dreamweaver_newsgroup.
The forums are a great resource for getting information on different browsers, platforms, and so on. You can also discuss technical issues and share helpful hints with other Dreamweaver users.

This chapter contains the following sections:

- “Checking for browser compatibility” on page 462
- “Using Behaviors to detect browsers and plug-ins” on page 463
- “Previewing pages in browsers” on page 463
- “Checking links in a page or site” on page 465
- “Fixing broken links” on page 466
- “Opening linked documents in Dreamweaver” on page 467
- “Setting download time and size” on page 467
- “Using Reports to test a site” on page 468

Checking for browser compatibility

Dreamweaver enables you to construct web pages with elements that are supported by all browsers (for example, images and paragraph text), as well as elements that are supported only by newer browsers (for example, styles and layers).

The Check Target Browsers feature tests the HTML in your documents to see if any tags or attributes are unsupported by your target browsers. The check does not alter the document in any way.

The Check Target Browsers feature uses text files called browser profiles to determine which tags particular browsers support. Dreamweaver includes predefined profiles for Netscape Navigator versions 2.0 and later, for Microsoft Internet Explorer versions 2.0 and later, and Opera versions 2.1 and later. To modify the existing profiles or to create new ones, see the Customizing Dreamweaver section at <http://www.macromedia.com/support/dreamweaver/>.

You can run a target browser check on a document, on a directory, or on an entire site.

Note: Running a target browser check does not check scripts in your site.

To run a target browser check:

- 1 Choose from the following options:
 - To run the check on the current document, first save your file. Dreamweaver performs the check on the last saved version of the file and does not include unsaved changes.
 - To run the check on a directory or site, select a site from the Current Sites pop-up menu in the Site panel. Dreamweaver performs the target browser check on all the HTML files in this folder and any folders in it. You can perform target browser checks on local files only.

- 2 Choose File > Check Page > Check Target Browsers.

If you have not yet selected a primary browser, Dreamweaver prompts you to do so.

- 3 From the list of browsers, select the target browser against which you want to check your site.
- 4 Click Check.

The report appears in the Target Browser Check panel (in the Results panel group).

- 5 To save the report, click the Save Report button in the Target Browser Check panel.

Note: The target browser report is a temporary file—it will be lost if you don't save it.

Using Behaviors to detect browsers and plug-ins

You can use behaviors to determine which browser your visitors are using and whether they have a particular plug-in installed. For more information on behaviors, see “Using JavaScript Behaviors” on page 349.

Check Browser sends visitors to different pages depending on their browser brands and versions (see “Check Browser” on page 355). For example, you might want visitors to go to one page if they have Netscape Navigator 4.0 or later, to go to another page if they have Microsoft Internet Explorer 4.0 or later, and to stay on the current page if they have any other kind of browser.

Check Plugin sends visitors to different pages depending on whether they have the specified plug-in installed (see “Check Plugin” on page 356). For example, you might want visitors to go to one page if they have Macromedia Shockwave and another page if they do not.

Previewing pages in browsers

It's a good idea to test your pages by previewing them in browsers often throughout the design and creation process. By using this strategy, you can catch errors early and not copy or repeat them.

You can preview a document in your target browsers at any time; you don't have to save the document first. All browser-related functions work, including JavaScript behaviors, document-relative and absolute links, ActiveX controls, and Netscape Navigator plug-ins, provided that you have installed the required plug-ins or ActiveX controls in your browsers.

Content linked with a root-relative path does not appear when you preview documents in a local browser (unless you select the Preview Using Temporary File option in Preferences; see “Previewing pages in browsers” on page 463). This is because browsers don't recognize site roots—servers do. To preview content linked with root-relative paths, put the file on a remote server, then choose File > Preview in Browser to view it (see “Site root-relative paths” on page 401).

You can define up to 20 browsers for previewing. All the browsers you define appear on the Preview in Browser menu. It's a good idea to preview in the following browsers: Internet Explorer 4.0, Netscape Navigator 4.0, and at least one text-only browser, like Lynx.

To preview your document in a browser, do one of the following:

- Choose File > Preview in Browser, then choose one of the listed browsers.

If you haven't selected a browser yet, choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X), and then select the Preview in Browser category on the left to select a browser.

- Press F12 to display the current document in the primary browser.

- Press Control+F12 (Windows) or Command+F12 (Macintosh) to display the current document in the secondary browser.

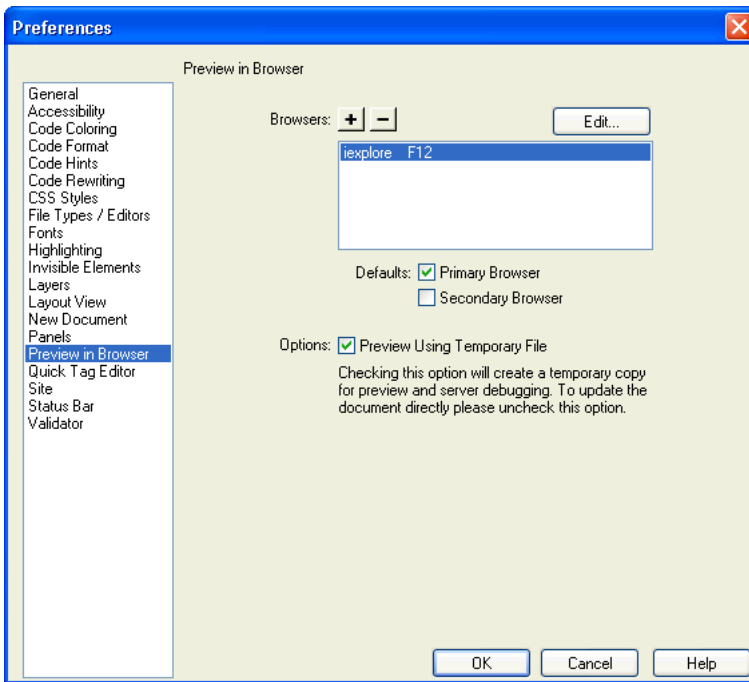
To test links in a browser:

- 1 Choose File > Preview in Browser or press F12.
- 2 Click the active links to verify that each one works properly.

To set or change preferences for your primary and secondary browsers:

- 1 Do one of the following to open the Preview in Browser options:
 - Choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X), and then select Preview in Browser from the category list on the left.
 - Choose File > Preview in Browser > Edit Browser List.

The Preferences dialog box appears with the Preview in Browser options.



- 2 Make changes as necessary.

For more information, click the Help button in the dialog box.

- 3 Click OK.

Checking links in a page or site

Fixing broken links (links that no longer follow a valid path or point to a nonexistent file) on a large site can be a tedious and time-consuming problem. This is because a large site can contain hundreds of links to internal and external documents, and the links might change over time. Orphaned files (files that still exist in the site but no other files in the site link to) can be a problem too, because they take up disk space and may confuse other team members working on the site.

Use the Check Links feature to search for broken links and unreferenced files in an open file, a portion of a local site, or an entire local site. The only links that Dreamweaver verifies are links to documents within the site; Dreamweaver compiles a list of external links that appear in the selected document or documents, but does not verify them.

When Dreamweaver finishes checking links in the specified files, it opens the Link Checker panel (in the Results panel group). This dialog box displays a list of broken links, external links (links that Dreamweaver can't check), and orphaned files. For more information, see “Opening linked documents in Dreamweaver” on page 467.

To check links within the current document:

- 1 Save the file to a location within your local Dreamweaver site.
- 2 Choose File > Check Page > Check Links.

The report appears in the Link Checker panel (in the Results panel group).

- 3 To save the report, click the Save Report button in the Link Checker panel.

Note: The target browser report is a temporary file—it will be lost if you don't save it.

To check links within a portion of a local site:

- 1 In the Site panel, choose a site from the Current Sites pop-up menu.
- 2 In Local view, select the files or folders to check.
- 3 Initiate the check by doing one of the following:
 - Right-click (Windows) or Control-click (Macintosh) one of the selected files, and then choose Check Links > Selected Files/Folders from the context menu.
 - Choose File > Check Page > Check Links.

The report appears in the Link Checker panel (in the Results panel group).

- 4 In the Link Checker panel, select a specific link report from the Show pop-up menu to view another report.

Your report choices are Broken Links and External Links. You can check for Orphaned Files when you check links across an entire site (see the next procedure).

- 5 To save a report, click the Save Report button in the Link Checker panel.

To check links in the entire site:

- 1 In the Site panel, choose a site from the Current Sites pop-up menu.
- 2 Choose Site > Check Links Sitewide.

The report appears in the Link Checker panel (in the Results panel group).

- 3 In the Link Checker panel, select a specific link report from the Show pop-up menu to view another report.

Your report choices are Broken Links, External Links, and Orphaned Files.

- 4 A list of files that fit the report type you selected appears in the Link Checker dialog box.

Note: If you select Orphaned Files as your report type, you can delete orphaned files from the Link Checker panel directly by selecting a file from the list and pressing the Delete key.

A list of files that fit the report type you selected appears in the Link Checker panel.

- 5 To save a report, click the Save Report button in the Link Checker panel.

Fixing broken links

When you check links in Dreamweaver, the Link Checker panel displays a report of broken links, external links, and—if you chose to check the entire site—orphaned files (see “Checking links in a page or site” on page 465).

You can fix broken links and image references directly in the Link Checker panel, or you can open files from the list and fix links in the Property inspector.

To fix links in the Link Checker panel:

- 1 Run a link check report (see “Checking links in a page or site” on page 465).
- 2 In the Broken Links column (not the Files column) of the Link Checker panel (in the Results panel group), select the broken link.
A folder icon appears next to the broken link.
- 3 Click the folder icon to browse to the correct file to link to, or type the correct path and filename.
- 4 Press Tab or Enter (Windows) or Return (Macintosh).

If there are other broken references to this same file, a dialog box appears prompting you to fix the references in the other files as well. Click Yes to have Dreamweaver update all the documents on the list that reference this file. Click No to have Dreamweaver update the current reference only.

Note: If Enable File Check In and Check Out is turned on for the site, Dreamweaver attempts to check out files that require changes. If it cannot check out a file, Dreamweaver displays a warning dialog box and leaves broken references unchanged. See “Using Check In/Check Out” on page 87.

To fix links in the Property inspector:

- 1 Run a link check report (see “Checking links in a page or site” on page 465).
- 2 In the Link Checker panel (in the Results panel group), double-click an entry in the File column.
Dreamweaver opens the document, selects the offending image or link, and highlights the path and filename in the Property inspector. (If the Property inspector is not visible, choose Window > Properties to open it.)
- 3 To set a new path and filename in the Property inspector, click the folder icon to browse to the correct file, or type over the highlighted text.
If you are updating an image reference and the new image appears at the incorrect size, click the W and H labels in the Property inspector or click the Refresh button to reset the height and width values. The W and H labels change from bold to normal type.
- 4 Save the file.

As links are fixed, their entries disappear from the Link Checker list. If an entry still appears in the list after you enter a new path or filename in the Link Checker (or after you save changes in the Property inspector), it means that Dreamweaver cannot find the new file and still considers the link broken.

Opening linked documents in Dreamweaver

Links are not active within Dreamweaver; that is, you cannot open a linked document by clicking the link in the Document window.

To open linked documents in Dreamweaver, do one of the following:

- Select the link and choose Modify > Open Linked Page.
- Press Control (Windows) or Command (Macintosh) and double-click the link.

Note: The linked document must reside on your local disk.

Setting download time and size

Dreamweaver calculates size based on the entire contents of the page, including all linked objects, such as images and plug-ins. Dreamweaver estimates download time based on the connection speed entered in Status Bar preferences. Actual download time varies depending on general Internet conditions.

A good guideline to use when checking download times for a particular web page is the 8-second rule. That is, most users will not wait longer than 8 seconds for a page to load.

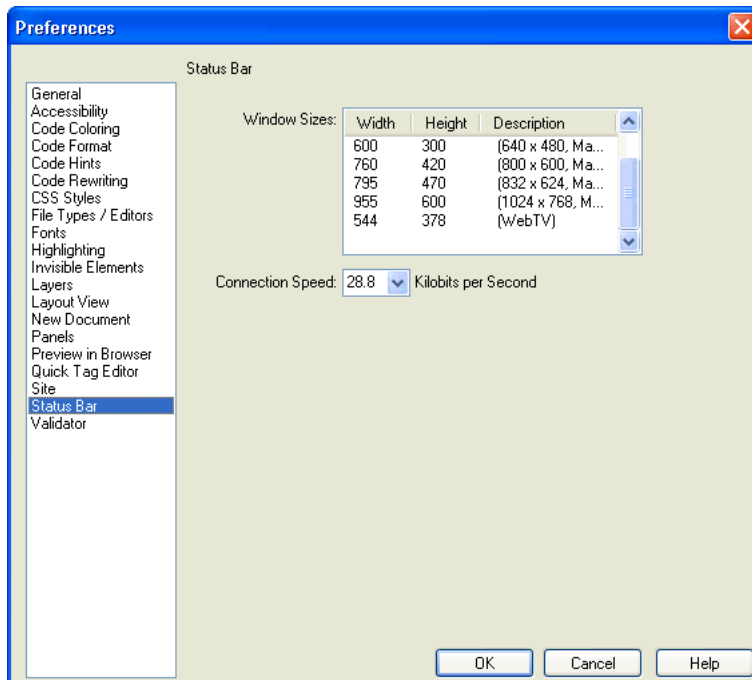
To set download time and size preferences:

- 1 Choose Edit > Preferences or Dreamweaver > Preferences (Mac OS X).

The Preferences dialog box appears.

- 2 Select Status Bar from the Category list on the left.

The Status Bar preference options appear.



- 3 Choose a connection speed with which to calculate download time.

The average connection speed in the United States is 28.8. If you are designing for an intranet, you may want to choose 1500 (T1 speed).

For more information about Status Bar preferences, see “Setting Status Bar preferences” on page 39.

- 4 Click OK.

Using Reports to test a site

When testing your site, you can compile and generate reports for several HTML attributes by using the Reports command. This command lets you check external links, combinable nested font tags, missing Alt text, redundant nested tags, removable empty tags, and untitled documents. You can check selected documents or an entire site for these HTML problems before you publish.

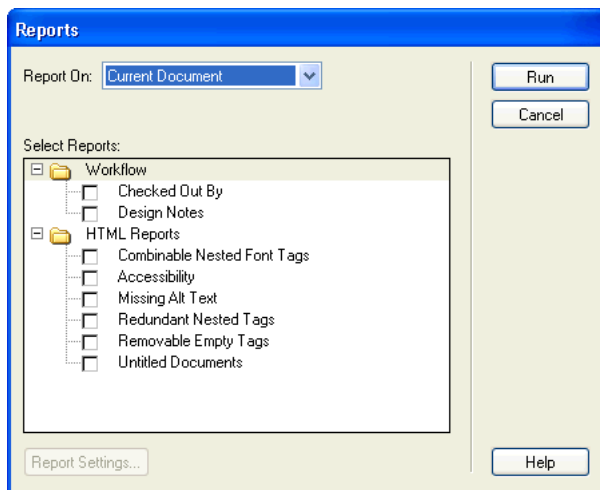
After you run a report, you can save it as an XML file, and then import it into a template instance or a database or spreadsheet and print it, or display it on a website. You can also add different report types to Dreamweaver through the Macromedia Dreamweaver Exchange website (see “Adding extensions to Dreamweaver” on page 53).

To use the Reports command to work with the Check In/Check Out system and Design Notes, see “Using reports to improve workflow” on page 101.

To run reports to test a site:

- 1 Choose Site > Reports.

The Reports dialog box appears.



- 2 Choose a category to report on and a report type to run.

For more information, click the Help button in the dialog box.

- 3 Click Run to create the report.

Depending on what you chose to report on, you may be prompted to save your file, define your site, or select a folder (if you haven't already done so).

A list of results appears in the Site Reports panel (in the Results panel group).

To use and save a report:

- 1 To sort the results, click the column heading you want to sort by.

You can sort by filename, line number, or description.

- 2 Select an item in the report, and then double-click to open the file in the Document window.

- 3 Click Save Report to save the report.

When you save a report, you can import it into a template instance or a database or spreadsheet and print it, or display it on a website.

Tip: After running HTML reports, use the Clean Up HTML command to correct any HTML errors the reports listed.

Part VIII

Making Pages Dynamic

Use the web application development tools in Dreamweaver to add dynamic content to your pages.

This part contains the following chapters:

- Chapter 29, “Optimizing the Workspace for Visual Development”
- Chapter 30, “The Dreamweaver Workflow for Dynamic Page Design”
- Chapter 31, “Storing and Retrieving Data for Your Page”
- Chapter 32, “Defining Sources of Dynamic Content”
- Chapter 33, “Adding Dynamic Content to Web Pages”
- Chapter 34, “Displaying Database Records”
- Chapter 35, “Using ColdFusion Components”
- Chapter 36, “Using Web Services”
- Chapter 37, “Adding Custom Server Behaviors”
- Chapter 38, “Creating Interactive Forms”

CHAPTER 29

Optimizing the Workspace for Visual Development

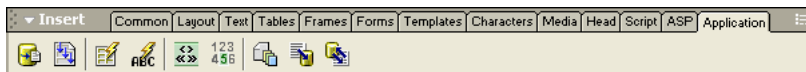
You can optimize the Macromedia Dreamweaver MX workspace to develop web applications visually. For example, you can use panels to rapidly build dynamic pages and you can view data live on your pages while you work.

This chapter contains the following sections:

- “Displaying useful panels” on page 473
- “Viewing your database structure within Dreamweaver” on page 475
- “Viewing live data in Design view” on page 476
- “Working in Design view without live data” on page 480
- “Previewing dynamic pages in a browser” on page 480
- “Restricting database information displayed in Dreamweaver” on page 481

Displaying useful panels

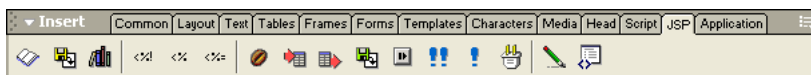
Click the Application tab in the Insert bar to display a set of buttons that let you add dynamic content and server behaviors to your page, as follows.



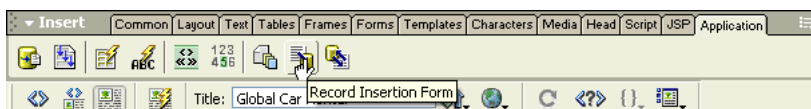
The number and type of icons that appear depend on the document type opened in the Document window. The Insert bar includes buttons to add the following items to the page:

- Recordsets
- Dynamic text or tables
- Forms to insert or update records in a database
- Record navigation bars

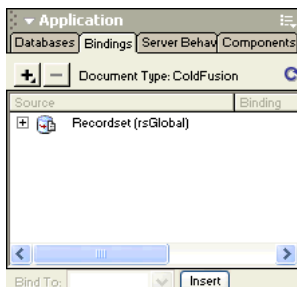
If you switch to Code view (View > Code), additional panels may appear to allow you to insert code in the page. For example, if you view a JSP page in Code view, a JSP panel appears on the Insert bar as follows:



To find out what each icon on the Insert bar does, move your mouse over an icon. A tooltip appears, as follows:

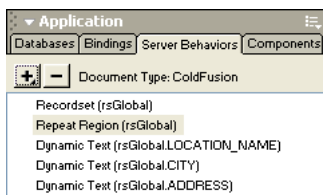


If you want to define sources of dynamic content for your page and add the content to the page, select Window > Data Bindings. The Bindings panel appears as follows:



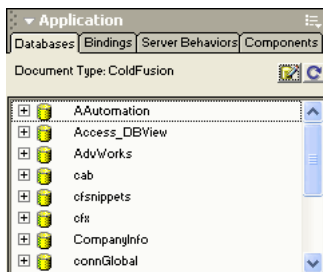
For more information, select Help from the panel group's pop-up menu.

If you want to add server-side logic to your dynamic pages, select Window > Server Behaviors. The Server Behaviors panel appears as follows:



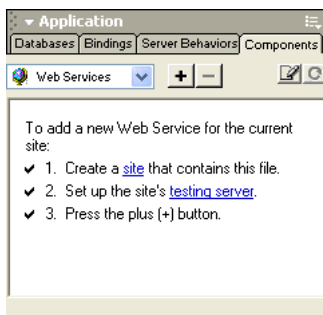
A server behavior is the set of instructions inserted in a dynamic page at design time and executed on the server at runtime. For more information, select Help from the panel group's pop-up menu.

If you want to explore databases or create database connections, select Window > Databases. The Databases panel appears as follows:



For more information, select Help from the panel group's pop-up menu.

If you want to inspect, add, or modify code for JavaBeans, Macromedia ColdFusion components, or web services, select Window > Components. The Components panel appears as follows:



Note: The Components panel is enabled only if you open a ColdFusion, a JSP, or an ASP.NET page. Also, the document might not support certain components. For example, ColdFusion documents do not support JavaBeans.

For more information, select Help from the panel group's pop-up menu.

Viewing your database structure within Dreamweaver

After connecting to your database, you can view its structure within Dreamweaver. For example, you can view the names of the tables, columns, views, and stored procedures in the database. Dreamweaver also identifies the primary key of each table, and the data types of each column. You can even view the data stored in your database tables.

To view the structure of a database:

- 1 Open the Databases panel (Window > Databases).

Dreamweaver populates the Databases panel with all the databases for which connections have been created. If you're developing a ColdFusion site, Dreamweaver populates the panel with all the databases for which data sources have been defined in ColdFusion Administrator.

Note: Dreamweaver looks at the ColdFusion server you defined for the current site. See "Specifying where dynamic pages can be processed" on page 134.

If no database appears in the panel, you must create a database connection. For more information, see "Connecting to a database" on page 135.

- 2 To display the tables, stored procedures, and views in the database, click the plus (+) icon beside a connection in the list.
- 3 To display the columns in the table, click a table name.
The columns icons reflect the data type and also indicates the primary key.
- 4 To view the data in a table, right-click (Windows) or Control-click (Macintosh) the table name in the list and choose View Data from the pop-up menu.

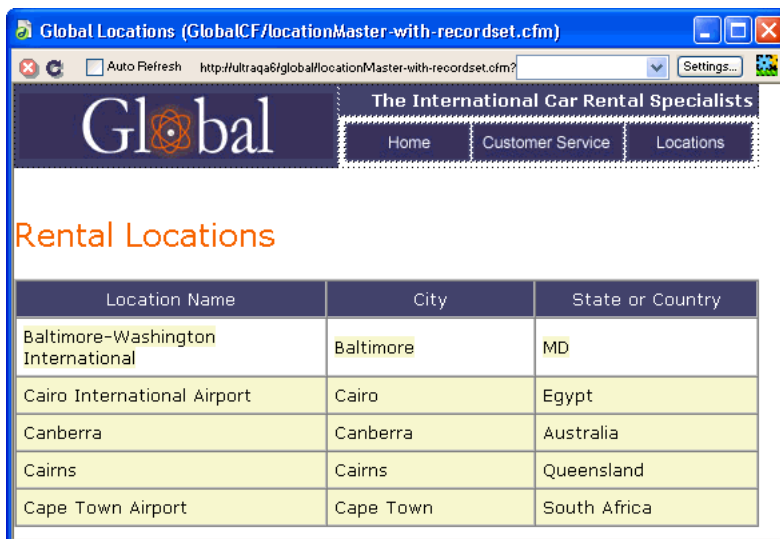
Viewing live data in Design view

By default, Design view (View > Design) gives you a visual representation of the page without dynamic content. The default view is not ideal if you're designing a dynamic page because the dynamic content can fundamentally change the layout of the page.

To see how dynamic content can affect the layout of a page, choose View > Live Data. Dreamweaver displays the page in Design view complete with dynamic content. The following illustration shows a dynamic page with Live Data turned off:



The following illustration shows the same page with Live Data turned on:



When Live Data is turned on in Design view, you can do the following:

- Adjust the page's layout using the page-design tools
- Add, edit, or delete dynamic content
- Add, edit, or delete server behaviors

Note: Links don't work in Design view. To test your links, use the Preview in Browser feature. (See "Previewing dynamic pages in a browser" on page 480.)

When you make a change to the page affecting dynamic content, you can refresh the page by clicking the Refresh button (the circle-arrow icon). Dreamweaver can also refresh the page for you automatically. For more information, see "Refreshing the page" on page 479.

Requirements for displaying live data

To view live data in Design view, you must do the following:

- Define a folder to process dynamic pages. For instructions, see "Specifying where dynamic pages can be processed" on page 134.

When you turn on Live Data, a temporary copy of the open document is sent to the folder for processing. The resulting page is returned and displayed in Design view, and the temporary copy on the server is deleted.

If the page displays an error message when you turn on Live Data, make sure the URL prefix in the Site Definition dialog box is correct. For more information, see "About the URL prefix" on page 134.

- Copy related files (if any) to the folder. For more information, see “Copying dependent files” on page 478.
- Provide the page with any parameters a user would normally provide. For instructions, see “Providing the page with expected parameters” on page 479.

If you have difficulty getting Live Data view to work, see “Troubleshooting Live Data view” on page 480.

Copying dependent files

Some dynamic pages rely on other files to work properly. You must upload all related files, including server-side includes and dependent files such as image files and JSP class files, to the folder you defined for processing dynamic pages (see “Specifying where dynamic pages can be processed” on page 134). Dreamweaver does not automatically copy dependent files to the folder when you turn on Live Data in Design view.

Note: Live Data supports code in server-side includes and application files such as `global.asa` (ASP) and `application.cfm` (ColdFusion). Make sure to upload these files to the server before turning on Live Data.

To copy dependent files to the application server:

- 1 Open the Site panel (Window > Site Files), then click the Expand button (the last icon on the panel toolbar).

The Site panel expands to full size.

- 2 Click the Application Server icon on the expanded Site panel toolbar (the second icon from the left).

The application server’s root folder appears under Remote Site.

- 3 Under Local Folder, select the dependent files.
- 4 Click the blue up arrow on the toolbar to copy the files to the application server, or drag the files to the appropriate folder under Remote Site.

You need to do this only once for your site unless you add more dependent files, in which case you must copy them to the folder too.

Providing the page with expected parameters

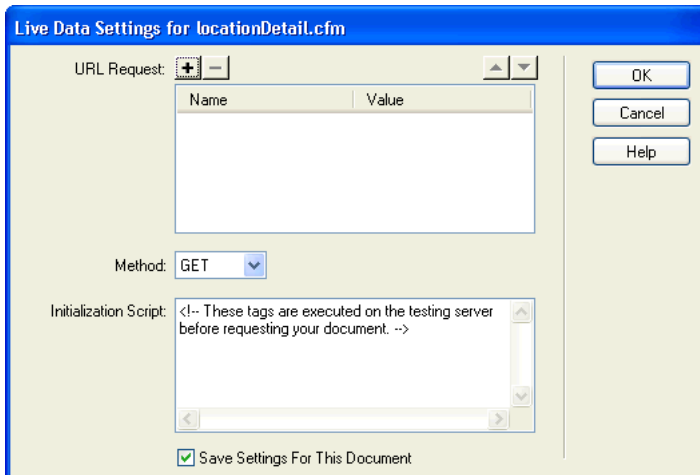
To generate dynamic content, some pages require parameters from the user—for example, a page needs the ID number of a record to find and display that record. Without that data, Dreamweaver cannot generate dynamic content to display in Design view.

If a page expects parameters from the user, you must provide the parameters as follows.

To provide the page with data expected from users:

- 1 In the Document window, choose Live Data Settings from the View menu.

The Live Data Settings dialog box appears.



- 2 Complete the dialog box and click OK.

For more information, click the Help button in the dialog box.

If you specified the GET method in the Live Data Settings dialog box, a text box appears on the Design view toolbar. Use this text box to enter different URL parameters, then click the Refresh button (the circle-arrow icon) to see how the parameters affect the page.

Enter each URL parameter in the following format:

```
name=value;
```

In this format, *name* is the URL parameter name expected by your page and *value* is the value held by that parameter. For more information, see “Understanding URL parameters” on page 494.

Refreshing the page

With Live Data turned on, click the Refresh button (the circle-arrow icon) on the document toolbar if you want to refresh the page after making a change affecting dynamic content.

You can also select the Auto Refresh option on the toolbar. With this option turned on, the page refreshes whenever you make a change affecting dynamic content. If you have a slow database connection, you might want to leave this option off when working in the Live Data window.

Troubleshooting Live Data view

Many problems with Live Data view can be traced back to missing or incorrect values in the Site Definition dialog box (Site > Edit Sites).

Check the Testing Server category of the Site Definition dialog box. The Remote Folder box should specify a folder capable of processing dynamic pages (see “Specifying where dynamic pages can be processed” on page 134). Here’s an example of a suitable remote folder if you’re running IIS or PWS on your hard disk:

```
C:\inetpub\wwwroot\myapp\
```

Verify that the URL Prefix box specifies a URL that corresponds (or “maps”) to the remote folder. For example, if PWS or IIS is running on your local computer, then the following remote folders have the following URL prefixes:

Remote folderx	URL prefix
C:\inetpub\wwwroot\	http://localhost/
C:\inetpub\wwwroot\myapp\	http://localhost/myapp/
C:\inetpub\wwwroot\fs\planes	http://localhost/fs/planes

For more information, see “About the URL prefix” on page 134.

Working in Design view without live data

If Live Data is turned off or if you’re temporarily disconnected from your application server, you can still work on your dynamic pages in Design view. Dreamweaver uses placeholders to visually represent dynamic content on the page. For example, the placeholder for dynamic text extracted from a database uses the syntax `{RecordsetName.ColumnName}`, where `Recordset` is the name of the recordset and `ColumnName` is the name of the column you chose from the recordset.

Sometimes, the length of the placeholders for dynamic text distorts the page’s layout in Design view. You can solve the problem by using empty curly braces as placeholders.

To use empty curly braces as placeholders for dynamic text:

- 1 Choose Edit > Preferences > Invisible Elements or Dreamweaver > Preferences > Invisible Elements (Mac OS X).
- 2 In the Show Dynamic Text As pop-up menu, choose {}.
- 3 Click OK.

Previewing dynamic pages in a browser

Web application developers often debug their pages by checking them quickly and often in a web browser. Dreamweaver accommodates this work style with the Preview in Browser command (F12). The command lets you quickly view dynamic pages in a browser without uploading them to a server first.

Tip: You can also use Design view to quickly check your pages while working on them. Design view displays a fully editable, visual representation of your page, including live data. For more information, see “Viewing live data in Design view” on page 476.

When you use this command, Dreamweaver runs a temporary copy of the page on a web server before displaying it in your browser. (Dreamweaver then deletes the temporary file from the server.)

To preview dynamic pages, you must complete the Testing Server category of the Site Definition dialog box. For more information, see “Specifying where dynamic pages can be processed” on page 134.

Preview in Browser does not upload related pages such as a results or a detail page, dependent files such as image files, or server-side includes. To upload a missing file, choose Window > Site to open the Site panel, select the file under Local Folder, and click the blue up arrow on the toolbar to copy the file to the web server folder.

Restricting database information displayed in Dreamweaver

Advanced users of large database systems like Oracle should restrict the number of database items retrieved and displayed by Dreamweaver at design time. An Oracle database may contain items that Dreamweaver cannot process at design time. You can create a schema in Oracle, then use it in Dreamweaver to filter out unnecessary items at design time.

Note: You cannot create a schema or catalog in Microsoft Access.

Other users may benefit from restricting the amount of information Dreamweaver retrieves at design time. Some databases contain dozens or even hundreds of tables, and you might not want Dreamweaver to list them all while you work. In the Recordset dialog box in Dreamweaver, for example, clicking the Tables pop-up menu lists all the tables in the specified database. If that database contains dozens of tables, the list will be long and hard to use. If the database contains hundreds of tables, Dreamweaver might take a long time to retrieve them on certain systems. A schema or catalog can restrict the number of database items Dreamweaver gets at design time.

You must create a schema or catalog in your database system before you can apply it in Dreamweaver. Consult your database system documentation or your system administrator.

You cannot apply a schema or catalog in Dreamweaver if you’re developing a ColdFusion application.

To apply a schema or catalog in Dreamweaver if you’re developing an application other than a ColdFusion one:

- 1 Open a dynamic page in Dreamweaver, then open the Databases panel (Window > Databases).
If the database connection already exists, a list of connections appears in the panel. Right-click (Windows) or Control-click (Macintosh) the connection, and choose Edit Connection from the pop-up menu. The dialog box appropriate for the connection appears.
If the connection does not exist, click the plus (+) button and create it. For more information, see “Connecting to a database” on page 135.
- 2 In the dialog box for the connection, click Advanced.
The Restrict dialog box appears.
- 3 Specify your schema or catalog, and click OK.

CHAPTER 30

The Dreamweaver Workflow for Dynamic Page Design

A key benefit provided by Macromedia Dreamweaver MX is the ability to create dynamic websites without having to be experienced in programming languages. Dreamweaver's visual tools let you develop dynamic websites without having to hand code the complex programmatic logic required to create a site supporting the display of dynamic content stored in a database. Dreamweaver lets you create dynamic websites using any of several popular web programming languages and server technologies. These include Macromedia ColdFusion, ASP.NET, Microsoft Active Server Pages (ASP), JavaServer Pages (JSP), and PHP.

This chapter outlines the key steps you must follow to successfully design and create a dynamic website. At the end of each section, you will find references to the specific procedures necessary to develop a dynamic page. The five key steps to developing a dynamic web page are:

- Designing the page
- Creating a source of dynamic content
- Adding dynamic content to a web page
- Enhancing the functionality of a dynamic page
- Testing and debugging the page

Designing the page

A key step in designing any website—whether static or dynamic—is the visual design of the page. When adding dynamic elements to a web page, the design of the page becomes crucial to its usability. Carefully consider how users will interact with both individual pages, and the website as a whole.

A common method of incorporating dynamic content into a web page is to create a table to present content, and import dynamic content into one or more of the table's cells. Using this method you can present information of various types in a structured format.

For information on designing pages, see the following chapters:

- Chapter 17, “Laying Out Pages in Layout View,” on page 241
- Chapter 16, “Presenting Content with Tables,” on page 227
- Chapter 19, “Inserting and Formatting Text,” on page 271
- Chapter 20, “Inserting Images,” on page 297

Creating a source of dynamic content

Dynamic websites require a content source from which to extract data before they can display it on a web page. In Dreamweaver, these data sources can be databases, request variables, server variables, form variables, or stored procedures.

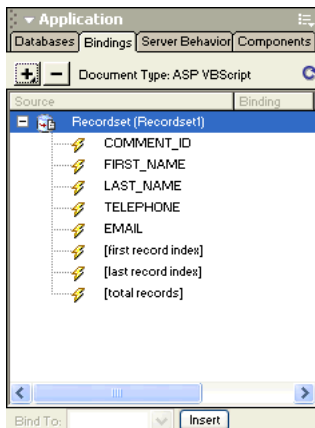
Before you can use these content sources in a web page, you must:

- Create a connection to the dynamic content source (such as a database) and the application server processing the page
- Specify what information in the database you want to display, or what variables you want to include in the page
- Use Dreamweaver's point-and-click interface to select and insert dynamic content elements into the selected page

Dreamweaver allows you to easily connect to a database and create a recordset from which to extract dynamic content. A **recordset** is the result of a database query. It extracts the specific information you request and allows you to display that information within a specified page. You define the recordset based on the information contained in the database and the content you want to display.

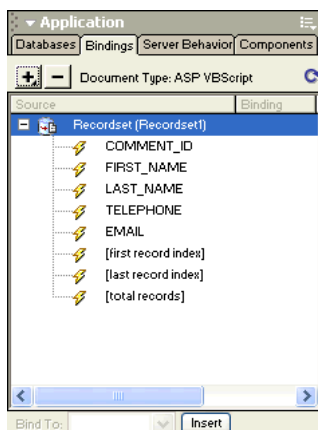
Different technology vendors may use different terminology for a recordset. In ASP and ColdFusion, a recordset is defined as a **query**. In JSP, a recordset is called a **resultset**. ASP.NET refers to a recordset as a **DataSet**. If you are using other sources of data, such as user input or server variables, the name of the data source that is defined in Dreamweaver is the same as the data source name itself.

To use a content source in Dreamweaver, you use the Bindings panel to create the data source. The Bindings panel, shown below, lets you create data sources for databases and different variable types. When you create a data source it is stored in the Bindings panel, where it can be selected and inserted into the current page.



To create a recordset in Dreamweaver, you use the Recordset dialog box. You can launch the Recordset dialog from either the Server pane of the Insert bar, or from the Bindings panel. The Simple Recordset dialog box lets you select an existing database connection and create a database query by selecting the table or tables whose data you want to include in the recordset. You can even use the Filter section of the dialog box to create simple search and return criteria for the query. You can test the query from within the Recordset dialog box, and make any needed adjustments before adding it to the Bindings panel.

Once a database connection is established, and a recordset defined, the recordset appears in the Bindings panel. From here you can import it into any web page within the defined site. The illustration below shows the Bindings panel with the recordset for an employee database open. You can insert any of the values shown into a web page by selecting the item, and clicking the Insert button at the bottom of the panel. The selected item is inserted into the specified placeholder within the page.



For information on databases, and the procedures needed to create a database connection, see the following chapters:

- Chapter 32, “Defining Sources of Dynamic Content,” on page 501
- Appendix A, “Beginner’s Guide to Databases,” on page 651
- Appendix C, “Setting Up a DSN in Windows,” on page 671

Adding dynamic content to a web page

After you define a recordset or other data source, and add them to the Bindings panel, you can insert the dynamic content the recordset represents into the page. The Dreamweaver menu-driven interface makes adding dynamic content elements as easy selecting a dynamic content source from the Bindings panel, and inserting it into an appropriate text, image, or form object within the current page.

When you insert a dynamic content element or other server behavior into a page, Dreamweaver inserts a server-side script into the page's source code. This script instructs the server to retrieve data from the defined data source and render it within the web page.

To place dynamic content within a web page, you can:

- Place it at the insertion point specified by the cursor in either Code or Design view.
- Replace a text string or other placeholder.
- Insert it into an HTML attribute. For example, dynamic content can define the `src` attribute of an image or the `value` attribute of a form field.

For detailed procedures on adding dynamic content to a page, see:

- Chapter 33, “Adding Dynamic Content to Web Pages,” on page 519
- Chapter 38, “Creating Interactive Forms,” on page 573

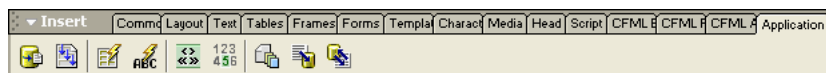
Enhancing the functionality of a dynamic page

In addition to adding dynamic content, Dreamweaver lets you easily incorporate complex application logic into web pages. You do this using Server behaviors. **Server behaviors** are predefined pieces of server-side code that add application logic to web pages, providing greater interaction and functionality. The Dreamweaver server behaviors allow you to add application logic to a web site without having to write the code yourself. The server behaviors supplied with Dreamweaver MX support ColdFusion, ASP, ASP.NET, JSP and PHP document types.

The server behaviors are written and tested to be fast, secure, and robust. Dreamweaver's built-in server behaviors have been designed to support cross-platform web pages for all browsers.

Adding server behaviors to a page

To add server behaviors to a page, select them from either the Insert bar, or the Server Behaviors panel. To use the Insert bar, click the Application tab and click a server behavior button. To use the Server Behaviors panel, choose Window > Server Behaviors, click the plus (+) button on the panel, and choose a server behavior from the pop-up menu. The illustration below shows the Server Behavior buttons available from the Insert bar.



Dreamweaver provides a point-and-click interface that makes applying dynamic content and complex behaviors to a page as easy as inserting textual and design elements. The following server behaviors are available:

- Define a recordset from an existing database. The recordset you define is then stored in the Bindings panel.
- Display multiple records on a single page. You select either an entire table or individual cells or rows that contain dynamic content, and specify the number of records to display on each page view.
- Create and insert a dynamic table into a page, and associate the table with a recordset. You can later modify both the table's appearance and the repeated region using the Property inspector and Repeated Region Server Behavior respectively.
- Insert a dynamic text object into a page. The text object you insert is an item from a predefined recordset, to which you can apply any of the Dreamweaver MX data formats.
- Create record navigation and status controls, master/detail pages, and forms for updating information in a database.
- Displaying more than one record from a database record.
- Creating recordset navigation links that allow users to view the previous or next records from a database record.
- Adding a record counter to help users keep track of how many records were returned, and where they are in the returned result.

You can also extend Dreamweaver server behaviors by writing your own, or installing server behaviors written by third-parties.

For more information on enhancing the functionality of web pages using server behaviors, see the following chapters:

- Chapter 34, “Displaying Database Records,” on page 527
- Chapter 40, “Building Pages that Search Databases,” on page 607
- Chapter 41, “Building Pages that Modify Databases,” on page 619
- Chapter 42, “Building Pages that Restrict Access to Your Site,” on page 639
- Chapter 37, “Adding Custom Server Behaviors,” on page 557

Testing and debugging the page

Before making a dynamic page—or an entire website—available on the web, you will want to test its functionality. Chapter 28, “Testing a Site,” on page 461 provides guidelines to help you test a website’s usability and cross-platform compatibility. For more information about designing web sites for people with auditory, visual, and other disabilities, see Chapter 23, “Dreamweaver and Accessibility,” on page 333. You should become familiar with the issues described in this chapter and consider how your application’s functionality might affect people with disabilities.

Testing Dynamic Content

Dreamweaver allows you to preview and edit dynamic content using the Live Data Window.

Note: Links don’t work in the Live Data window. To test your links, use the Dreamweaver Preview in Browser feature. (See “Previewing pages in browsers” on page 463.)

While dynamic content is displayed, you can perform the following tasks:

- Adjust the page’s layout using the Dreamweaver page-design tools
- Add, edit, or delete dynamic content
- Add, edit, or delete server behaviors

To achieve this effect, Dreamweaver runs the dynamic page on your server before displaying it in the Live Data window. Whenever you switch to the Live Data window, a temporary copy of the open document is sent to your application server for processing. The resulting page is returned and displayed in the Live Data window, and the temporary copy on the server is deleted.

You can toggle between the Document window and the Live Data window by choosing Live Data from the View menu. If a page expects data from the user—for example, the ID number of a record selected in a master page—you can provide the page with that data yourself in the Live Data Settings dialog box.

To enter live data parameters:

- 1 Make the necessary changes to the page.
- 2 If your page expects URL parameters from an HTML form using the GET method, enter the name/value pairs in the text box on the toolbar and click the Refresh button (the circle-arrow icon).

Enter the test data in the following format:

```
name=value;
```

In this format, *name* is the URL parameters name expected by your page and *value* is the value held by that parameter.

You can also define name/value pairs in the Live Data Settings dialog box (View > Live Data Settings) and save them with the page.

- 3 Click the Refresh button if your page needs refreshing.

Editing dynamic content on a page

You can modify or remove dynamic content from a page by changing the server behavior that provides the content, or deleting the behavior altogether. For example, you can edit a recordset server behavior to return more records to the page.

Dynamic content on a page is listed in the Server Behaviors panel. For example, if you add a recordset to your page, the Server Behaviors panel lists it as follows:

```
Recordset(myRecordset)
```

If you add another recordset to your page, the Server Behaviors panel lists both recordsets as follows:

```
Recordset(mySecondRecordset)  
Recordset(myRecordset)
```

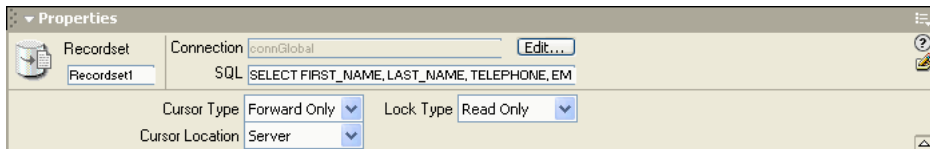
To edit a server behavior providing dynamic content:

- 1 Open the Server Behaviors panel (Window > Server Behaviors).
- 2 Click the plus (+) button to display the server behaviors, and double-click the server behavior in the Server Behaviors panel.

The dialog box you used to define the original data source appears.

- 3 Make your changes in the dialog box and click OK.

You can also use the Property inspector to edit the recordsets on the page. Open the Property inspector (Window > Properties), then select the recordset in the Server Behaviors panel (Window > Server Behavior). Here's the Property inspector for a recordset:



Deleting dynamic content

After adding dynamic content to a page, you can delete it by selecting the dynamic content on the page and pressing Delete. You can also delete it by selecting the dynamic content in the Server Behaviors panel and clicking the minus (-) button.

Note: This operation removes the server-side script in the page that retrieves the dynamic content from the database. It does not delete the data in the database.

CHAPTER 31

Storing and Retrieving Data for Your Page

Web-based applications and dynamic websites require a content source from which to retrieve data. Both data and data sources can take many forms. Typically, data consists of text or numerical information returned to a web page, and displayed in some form to the user.

This chapter contains the following sections:

- “Using a database to store content” on page 491
- “Collecting data submitted by users” on page 492
- “Accessing data stored in session variables” on page 495

Using a database to store content

Using a database to store content allows you to separate your website’s design from the content you want to display to users of the site. Instead of writing individual HTML files for every page, you only need to write a page—or template—for the different kinds of information you want to present. Using a database, you can provide new content to a website by simply uploading content into a database and then retrieving that content dynamically in response to a user’s request. A key advantage of storing content information in a database is the ability to update information in a single source, and then populate that change throughout the website without having to search through the pages that might contain the information, and manually edit each page.

Databases come in many forms depending upon the amount and the complexity of the data they must store. A database commonly installed on Windows computers is Microsoft Access. If you are new to databases, Access provides an easy-to-use interface that lets you work with database tables. While you can use Access as a data source for most website applications, be aware that Access has a file size limitation of 2 gigabytes (GB), and is limited to 255 concurrent users. For this reason, Access is a reasonable choice for website development and corporate workgroups. However, if you anticipate a large user community accessing the site, plan on using a database designed to support your sites intended user base.

For websites requiring a greater degree of flexibility in their modeling of data, and the ability to support large, concurrent user communities, server-based relational databases (typically referred to as RDBMS) are commonly used. Common relational databases used to store content for web-based applications and dynamic sites include MySQL, Microsoft SQL Server, and Oracle.

Whatever database you choose to support your website, you can use Dreamweaver to design web forms to insert, update, or delete data from the database.

To learn more about databases, see Appendix A, “Beginner’s Guide to Databases,” on page 651.

Accessing data stored in a database

Web pages can't directly access the data stored in a database. Instead, they interact with a **recordset**. A recordset is a subset of the information, or records, extracted from the database. This subset of information is extracted using a database **query**. A query is a search statement designed to find and extract specific information from a database. Macromedia Dreamweaver MX uses the Structured Query Language (SQL) to build queries. Although you do not need to learn SQL (pronounced "sequel") to be able to create simple queries using Dreamweaver, a basic knowledge of this easy-to-understand language lets you create more advanced queries, and thus provides you with greater flexibility in designing dynamic pages. To learn the basics of SQL, see Appendix B, "SQL Primer," on page 663.

A SQL query can produce a recordset that includes only certain columns, only certain records, or a combination of both. A recordset can also include all the records and columns of a database table. However, because applications rarely need to use every piece of data in a database, you should strive to make your recordsets as small as possible. Because the web server temporarily holds the recordset in memory, using a smaller recordset uses less memory, and can potentially improve server performance.

Collecting data submitted by users

You can use web pages to gather information from users, store that information in the server's memory, then use the information to create a dynamic response based on the user's input. The most common tools for gathering information are HTML forms and hypertext link selections.

HTML forms let you gather information from users and store it in the server's memory. An HTML form can send the information either as form parameters or as URL parameters. If you set the form's method attribute to POST, the browser includes the form's values in the body of the message sent to the server. If you set the form's method attribute to GET, the browser appends the form values to the URL specified in the action attribute and sends the information to the server.

Hypertext links also let you gather information from users and store it in the server's memory. You specify a value (or values) to be submitted when a user clicks a link—a preference, for example—by appending the value to the URL specified in the anchor tag. When a user clicks the link, the browser sends the URL and the appended value to the server.

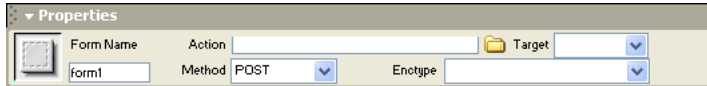
This section describes how to create form and URL parameters for use in web applications. The following topics are discussed in this section:

- "Understanding form parameters" on page 493
- "Understanding URL parameters" on page 494
- "Creating URL parameters using HTML links" on page 495
- "Creating URL parameters using HTML links" on page 495
- "Retrieving form and URL parameters" on page 495

Understanding form parameters

Form parameters are sent to the server using an HTML form using either the `POST` or `GET` method. When using the `POST` method, parameters are sent in the body of the message. In contrast, the `GET` method appends parameters to the requested URL.

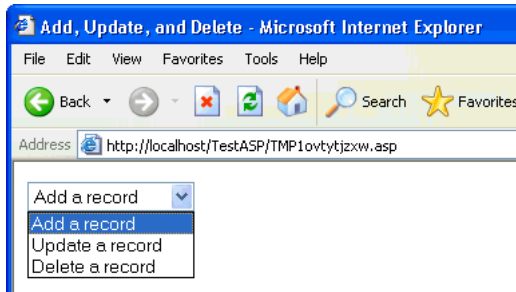
You can use Dreamweaver to quickly design HTML forms that send form parameters to the server. For instructions, see Chapter 38, “Creating Interactive Forms,” on page 573. Be aware of the method you choose to transmit information from the browser to the server. The example below illustrates a form’s property inspector with the form method set to `POST`:



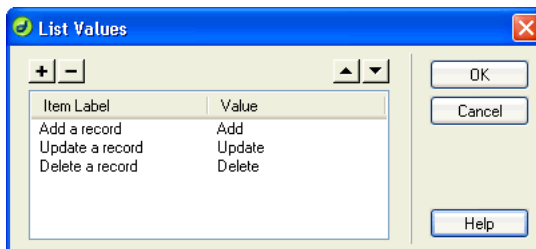
Form parameters take the names of their corresponding form objects. For example, if your form contains a text field named `txtLastName`, then the following form parameter is sent to the server when the user clicks the Submit button:

```
txtLastName=enteredvalue
```

In cases where a web application expects a precise parameter value (for example, when it performs an action based on one of several options), use a radio button, check box, or list/menu form object to control the possible values the user can submit. This prevents users from mis-typing information, and causing an application error. The example below depicts a pop-up menu form offering three choices:



Each menu choice corresponds to a hard-coded value that is submitted as a form parameter to the server. The List Values dialog box—shown below—matches each list item to a value (Add, Update, or Delete):



Understanding URL parameters

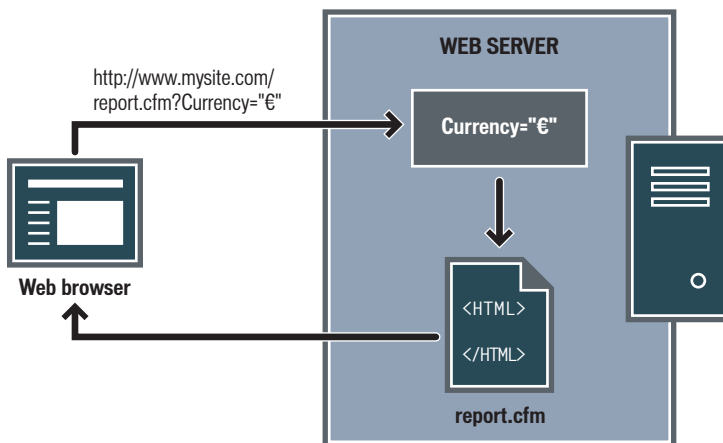
A URL parameter is a name/value pair appended to a URL. The parameter begins with a question mark (?) and takes the form `name=value`. If more than one URL parameter exists, each parameter is separated by an ampersand (&). The example below shows a URL parameter with two name/value pairs:

```
http://server/path/document?name1=value1&name2=value2
```

Use URL parameters to pass user-supplied information from the browser to the server. When a server receives a request, and parameters are appended to the URL of the request, the server puts the parameters at the disposal of the requested page before serving that page to the browser.

In this example, imagine that the application is a web-based storefront. Because they want to reach the widest possible audience, the developers of the site have designed the site to support foreign currencies. When users log in to the site, they can select what currency in which to view the prices of the available items.

- 1 The browser requests the page `report.cfm` from the server. The request includes the URL parameter `Currency="euro"`. The `Currency="euro"` variable specifies that all monetary amounts retrieved be displayed as the European Union's Euro.
- 2 The server temporarily stores the URL parameter in memory.
- 3 The `report.cfm` page retrieves and uses the parameter to retrieve the cost of items in Euros. These monetary amounts can either be stored in a database table of different currencies, or converted from a single currency associated with each item any currency supported by the application.
- 4 The server sends the `report.cfm` page to the browser, and displays the value of items in the requested currency. When this user terminates the session, the server destroys the value of the URL parameter, freeing up server memory to hold the requests of new users logging in to the site.



URL parameters are created when the HTTP's `GET` method is used in conjunction with an HTML form. The `GET` method specifies that the parameter value be appended to the URL request when the form is submitted.

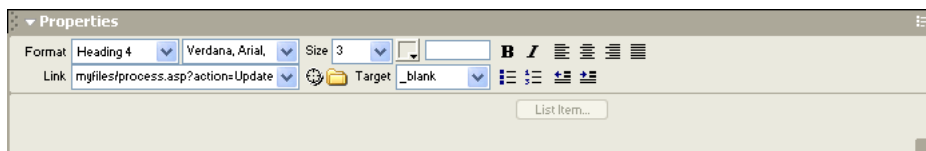
Typical uses of URL parameters include personalizing websites based on a user's preferences. For example, a URL parameter consisting of a username and password can be used to authenticate a user, displaying only information that user has subscribed to. Common examples of this include financial websites that display individual stock prices based on stock market symbols the user has previously chosen. Web application developers commonly use URL parameters to pass values to variables within applications. For example, you could pass search terms to SQL variables in a web application to generate search results.

Creating URL parameters using HTML links

Creating a URL parameters within an HTML link consists of using the `href` attribute of the HTML anchor tag. You can enter the URL parameters directly in the attribute by switching to Code view (View > Code). In the example below, three links create a single URL parameter (`action`) with three possible values (`Add`, `Update`, and `Delete`). Depending on the link the user clicks, a different parameter value is sent to the server, and the requested action performed:

```
<a href="http://www.mysite.com/myfiles/  
index.asp?action=Add">Add a record</a>  
<a href="http://www.mysite.com/myfiles/  
index.asp?action=Update">Update a record</a>  
<a href="http://www.mysite.com/myfiles/  
index.asp?action=Delete">Delete a record</a>
```

You can use Dreamweaver's Property inspector to create the same URL parameters by selecting the link and choosing Window > Properties. In the following example, the URL parameter was entered in the Property inspector:



Retrieving form and URL parameters

After a form or URL parameter is created, Dreamweaver can retrieve the value and use it in a web application. For more information, see “Defining URL parameters” on page 507.

After defining the form or URL parameter in Dreamweaver, you can insert its value within a page. For more information, see Chapter 33, “Adding Dynamic Content to Web Pages,” on page 519.

Accessing data stored in session variables

Session variables provide a mechanism through which user information can be stored and accessed for use by web applications. Typically, session variables store information (usually form or URL parameters submitted by users) and make that information available to all of the applications's pages for the duration of the user's visit. For example, when users log on a web portal that provides access to e-mail, stock quotes, weather reports, and daily news, the web application stores the login information in a session variable that identifies the user throughout the site's pages. This allows the user to see only the types of content they have selected as they navigate through the site. Session variables can also provide a safety mechanism in the form of a time-out that terminates the user's session if the account remains inactive for too long a period of time. This also frees server memory and processing resources if the users forgets to log off of a website.

Session variables are commonly used to store user display preferences, answers to multipart questionnaires, items chosen for purchase in so-called “shopping cart” applications, and running score tallies for online games.

This section describes the following session variable topics:

- “Understanding session variables” on page 496
- “Collecting information to store in session variables” on page 497
- “Storing information in session variables” on page 498
- “Example of information stored in session variables” on page 499
- “Retrieving data from session variables” on page 500

Understanding session variables

Web servers (or more correctly, the HTTP protocol) are stateless, meaning that they do not keep track of the browsers connecting to them, or of the individual page requests by users. Every time a web server receives a request for a web page and responds to it by delivering the relevant page to the user’s browser, the web server “forgets” about both the browser making the request and the web page it sent. When the same user requests a related page at a later time, the web server sends the page without knowing the last page that it sent to that user.

While the stateless nature of HTTP makes for a simple, easy-to-implement protocol, it makes more advanced web applications, such as personalized content generation, more difficult. For example, in order to customize a site’s content for an individual user, the user must first be identified. Most websites use some form of username/password login to accomplish this. If multiple customized pages will be displayed, a mechanism for keeping track of which users are logged in is necessary, as most users would find it unacceptable to provide their username/password for each of the site’s pages.

To allow for the creation of complex web applications, and the storage of user-supplied data across all of a site’s pages, most application server technologies include support for **session management**. Session management allows web applications to maintain state across multiple HTTP requests, allowing a user’s requests for web pages during a given time period to be viewed as part of the same interactive session.

Session variables store information for the life of the user’s session. The user’s session begins when he or she first opens a page within the application. The session ends when the user does not open another page in the application for a certain period of time, or when the user explicitly terminates the session (typically by clicking a “log-off” link). While it exists, the session is specific to an individual user, and every user has a separate session.

Use session variables to store information that every page in a web application can access. The information can be as diverse as the user’s name, preferred font size, or a flag indicating whether or not the user has successfully logged in. Another common use of session variables is to keep a running tally, such as the number of questions the user answered correctly so far on an online quiz, or the products the user selected so far from an online catalog.

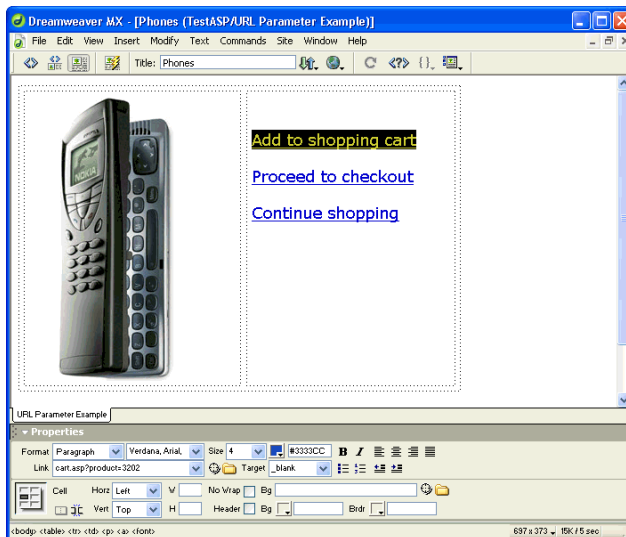
Note that session variables can only function if the user’s browser is configured to accept cookies. The server creates a session ID number that uniquely identifies the user when the session is first initiated, then sends a cookie containing the ID number to the user’s browser. When the user requests another page on the server, the server reads the cookie in the browser to identify the user and to retrieve the user’s session variables stored in the server’s memory.

Collecting information to store in session variables

Before creating a session variable, you must first obtain the information you want to store, then send it to the server for storage. This section describes how to gather and send information to the server using HTML forms or hypertext links containing URL parameters. You can also obtain information from cookies stored on the user's computer, from the HTTP headers sent by the user's browser with a page request, or from a database.

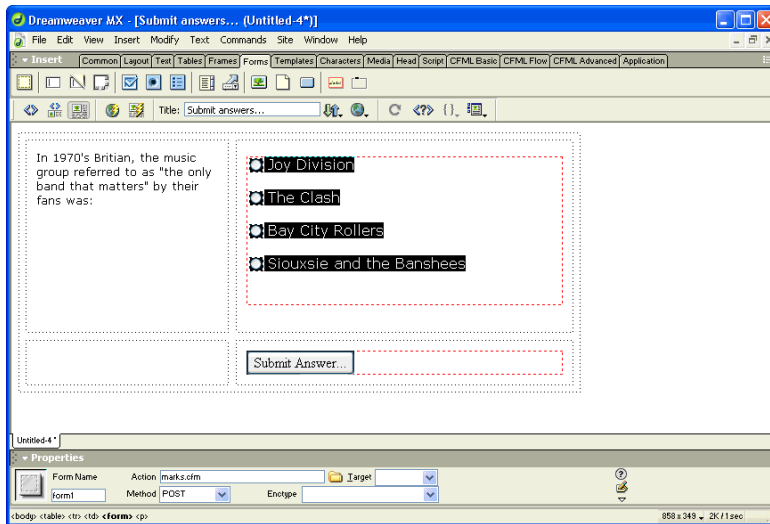
Storing URL parameters in session variables

In the following example, a product catalog uses hard-coded URL parameters created using a hypertext link to send product information back to the server to be stored in a session variable. When a user clicks the “Add to shopping cart” link, the product ID (3202 in this example), is stored in a session variable while the user continues to shop. When the user proceeds to the check-out page, the product number stored in the session variable is retrieved.



Storing form parameters in session variables

In this example, a form-based quiz page sends the selected information back to the server, where the `marks.cfm` page scores the quiz, and stores the number of correct answers in a session variable.



Storing information in session variables

Once information is sent to the server, you store the information in session variables by adding the appropriate code for your server model to the page specified by the URL or form parameter. Referred to as the “destination” page, this page is specified in either the `action` attribute of the HTML form or the `href` attribute of the hypertext link on the starting page.

The HTML syntax for each appears as follows:

```
<form action="destination.html" method="get" name="myform"> </form>  
<param name="href" value="destination.html">
```

Both the server technology used and the method you use to obtain the information determines the code used to store the information in a session variable. The basic syntax for each server technology is as follows:

ColdFusion

```
<CFSET session.variable_name = value>
```

ASP and ASP.NET

```
<% Session("variable_name") = value %>
```

The value expression is usually a server expression such as `Request.Form("lastname")`. For example, if you use a URL parameter called `product` (or an HTML form with the `get` method and a text field called `product`) to gather information, the following statements store the information in a session variable called `prodID`:

ColdFusion

```
<CFSET session.prodID = url.product>
```

ASP and ASP.NET

```
<% Session("prodID") = Request.QueryString("product") %>
```

If you use an HTML form with the `post` method and a text field called `txtProduct` to gather the information, then the following statements store the information in the session variable:

ColdFusion

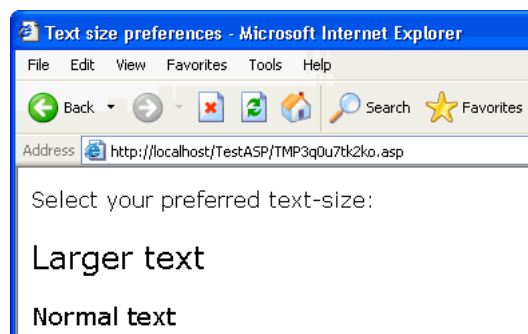
```
<CFSET session.prodID = form.txtProduct>
```

ASP and ASP.NET

```
<% Session("prodID") = Request.Form("txtProduct") %>
```

Example of information stored in session variables

You're working on a site with a large audience of senior citizens. In Dreamweaver, you add two links to the start page that let users customize the size of the site's text. For larger, easy-to-read text, the user clicks one link, and for regular-size text, the user clicks another link:



Each link has a URL parameter called `fontsize` that submits the user's text preference to the server, as the following Macromedia ColdFusion example shows:

```
<a href="resort.cfm?fontsize=large">Larger Text</a><br>
<a href="resort.cfm?fontsize=small">Normal Text</a>
```

You decide to store the user's text preference in a session variable and use it to set the font size on each page the user requests.

Near the top of the destination page, you enter the following code to create a session called `font_pref` that stores the user's font size preference.

ColdFusion

```
<CFSET session.font_pref = url.fontsize>
```

ASP and ASP.NET

```
<% Session("font_pref") = Request.QueryString("fontsize") %>
```

When the user clicks the hypertext link, the page sends the user's text preference in a URL parameter to the destination page. The code on the destination page stores the URL parameter in the `font_pref` session variable. For the duration of the user's session, all the pages of the application retrieve this value, and display the selected font size.

Retrieving data from session variables

Once you store a value in a session variable, you can use Dreamweaver to retrieve the value from session variables and use it in a web application. For more information, see “Defining session variables” on page 509.

After you define the session variable in Dreamweaver, you can insert its value in a page. For more information, see Chapter 33, “Adding Dynamic Content to Web Pages,” on page 519.

CHAPTER 32

Defining Sources of Dynamic Content

Dynamic websites require a data source from which to retrieve and display dynamic content. Macromedia Dreamweaver MX lets you use databases, request variables, URL variables, server variables, form variables, stored procedures and other sources of dynamic content. Depending on the data source, you can either retrieve new content to satisfy a request, or modify the page to meet the needs of users.

This chapter describes the procedures for defining both databases and variables as sources of dynamic content. To learn more about the technologies discussed in this chapter, refer to “HTML and web technologies resources” on page 29.

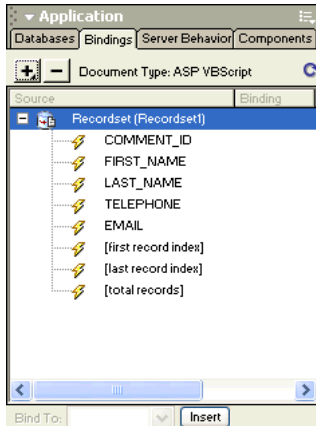
This chapter contains the following sections:

- “Understanding dynamic content sources” on page 502
- “Defining a recordset” on page 502
- “Defining URL parameters” on page 507
- “Defining session variables” on page 509
- “Defining application variables” on page 510
- “Defining server variables” on page 511
- “Caching content sources” on page 516
- “Changing or deleting content sources” on page 516
- “Copying a recordset from one page to another page” on page 517

Understanding dynamic content sources

A dynamic content source is a store of information from which you can retrieve and display dynamic content for use in a web page. Sources of dynamic content include not only information stored in a database, but values submitted by HTML forms, values contained in server objects, values of JavaBeans properties, and other content sources.

Any content source you define in Dreamweaver is added to the list of content sources in the Bindings panel. Once you create a content source and include it in the Bindings panel, you can easily insert it into the currently selected page.



Defining a recordset

When using a database as a content source for a dynamic web page, you must first create a recordset in which to store the retrieved data. **Recordsets** serve as an intermediary between the database storing the content and the application server generating the page. Recordsets consist of the data returned by a database query, and are temporarily stored in the application server's memory for faster data retrieval. The server discards the recordset when it is no longer needed.

The recordset itself is a collection of data retrieved from a specified database. It can include an entire database table, or a subset of the table's rows and columns. These rows and columns are retrieved by means of a database query that is defined in the recordset. Database queries are written in Structured Query Language (SQL), a simple language that allows you to retrieve, add, and delete data to and from a database. The SQL builder included with Dreamweaver lets you create simple queries without having to understand SQL. However, if you want to create complex SQL queries, you will need to learn SQL, and manually write SQL statements that you enter into Dreamweaver.

Note: Microsoft ASP.NET refers to a recordset as a DataSet. If you are working with ASP.NET document types, the dialog boxes and menu choices specific to ASP.NET use the label DataSet. The Dreamweaver documentation generically refers to both types as recordsets, but uses DataSet when specifically describing ASP.NET features.

For more information on using databases and SQL, see the following chapters, Appendix A, “Beginner’s Guide to Databases,” on page 651, and Appendix B, “SQL Primer,” on page 663.

Before you define a recordset for use with Dreamweaver, you must create a connection to a database and—if no data exists yet—enter data into the database. If you have not yet defined a database connection for your site, refer to the database connection chapter for the server technology you are developing for, and follow the instructions on creating a database connection.

Database connections for all Dreamweaver supported server technologies are described in the following chapters:

- Chapter 7, “Database Connections for ColdFusion Developers,” on page 139
- Chapter 9, “Database Connections for ASP Developers,” on page 151
- Chapter 8, “Database Connections for ASP.NET Developers,” on page 145
- Chapter 10, “Database Connections for JSP Developers,” on page 165
- Chapter 11, “Database Connections for PHP Developers,” on page 171

This section contains the following topics:

- “Creating a recordset without writing SQL” on page 503
- “Creating an advanced recordset by writing SQL” on page 504
- “Creating SQL queries using the Database Items tree” on page 506

Creating a recordset without writing SQL

This section describes how to define a recordset using Dreamweaver’s Recordset dialog box, which allows you to create a recordset without having to manually enter SQL statements. Defining a recordset using this method can be as easy as selecting a database connection and table from the Recordset dialog box’s pop-up menus.

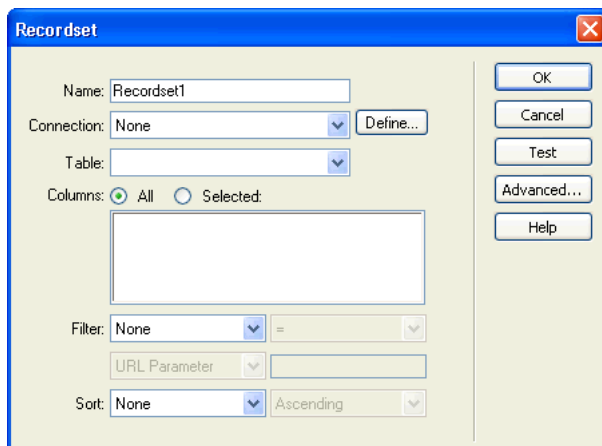
If you want to write your own SQL statements, use the Advanced Recordset dialog box. (See “Creating an advanced recordset by writing SQL” on page 504.)

To define a recordset without writing SQL:

- 1 In the Document window, open the page that will use the recordset.
- 2 Choose Windows > Bindings to display the Bindings panel.

- 3 In the Bindings panel, click the plus (+) button and choose Recordset (Query) from the pop-up menu.

The Simple Recordset dialog box appears. If you are developing a Macromedia ColdFusion MX or ASP.NET site, the Recordset dialog box is slightly different from the example shown below. If the Advanced Recordset dialog appears instead, switch to the Simple Recordset dialog box by clicking the Simple button.



- 4 Complete the dialog box.
For more information, click the Recordset dialog box's Help button.
- 5 Click the Test button to execute the query and ensure that it retrieves the information you intended.

If you defined a filter that uses parameters input by users, the Test button displays the Test Value dialog box. Enter a value in the Test Value text box and click OK. If an instance of the recordset is successfully created, a table displaying data extracted from the recordset appears.

- 6 Click OK to add the recordset to the list of available content sources in the Bindings panel.

Creating an advanced recordset by writing SQL

The Advanced Recordset dialog box allows you to write your own SQL statements, or use the graphical Database Items tree to create a SQL statement

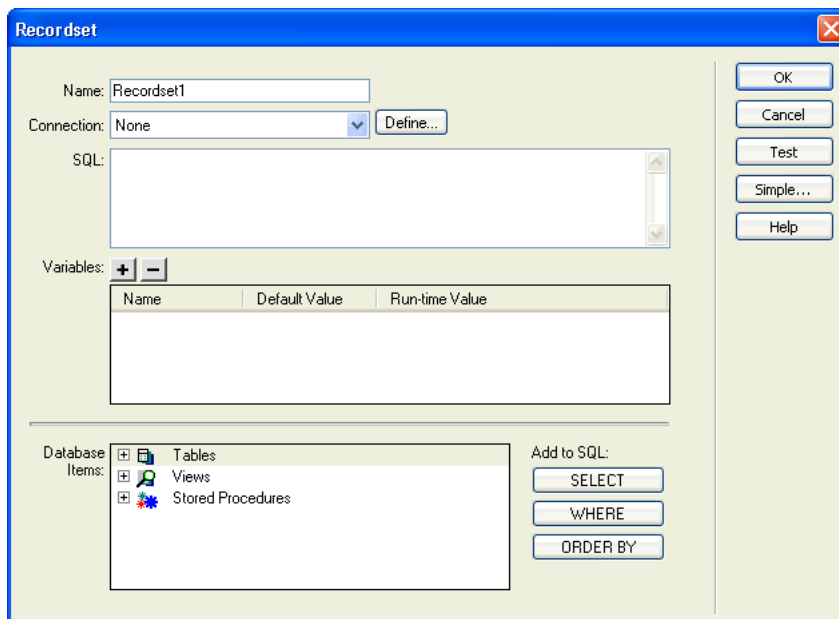
Note: If you are writing SQL statements for ASP.NET document types, refer to "Writing SQL for ASP.NET" on page 506 for rules specific to ASP.NET.

To define a recordset writing SQL:

- 1 In the Document window, open the page that will use the recordset.
- 2 Choose Windows > Bindings to display the Bindings panel.

- 3 In the Bindings panel, click the plus (+) button and select Recordset (Query) from the pop-up menu.

The Advanced Recordset dialog box appears. If you are developing a ColdFusion MX or ASP.NET site, the Recordset dialog box is slightly different from the example shown below. If the Simple Recordset dialog appears instead, switch to the Advanced Recordset dialog box by clicking the Advanced button.



- 4 Complete the dialog box.

For instructions on completing the Recordset dialog box, click the Recordset dialog box's Help button.

- 5 Click the Test button to execute the query and ensure that it retrieves the information you intended.

If you defined a filter that uses parameters input by users, the Test button displays the Test Value dialog box. Enter a value in the Test Value text field and click OK. If an instance of the recordset is successfully created, a table displaying the data from the recordset appears.

- 6 Click OK to add the recordset to the list of available content sources in the Bindings panel.

Writing SQL for ASP.NET

When writing SQL statements in the Advanced DataSet dialog box, there are conditions specific to ASP.NET that you must be aware of:

Parameters

The syntax you use to reference parameters varies depending on the database connection in use. The connection types are:

OLE DB

When connecting to a database using OLE DB, parameters must be referenced using a question mark (?). For example:

```
SELECT * FROM Employees WHERE HireDate > ?
```

Microsoft SQL Server

When connecting to Microsoft SQL Server using the Managed Data Provider for SQL Server supplied with the .NET Framework, all parameters must be named. For example:

```
SELECT * FROM Employees WHERE HireDate > @hireDate
```

Inserting code within SQL statements

When inserting code within SQL statements written for ASP.NET, you must enclose all strings in quotes (“ ”), and encase the code in parentheses ():

```
SELECT * FROM Employees WHERE HireDate > “+ (Request.queryString(“hireDate”))
```

Database Connections in ASP.NET

To learn more about database connections in ASP.NET, see Chapter 8, “Database Connections for ASP.NET Developers,” on page 145.

Creating SQL queries using the Database Items tree

Instead of manually typing SQL statements into the SQL text box, you can use the Database Item’s point-and-click interface to create complex SQL queries. The Database Items tree lets you select database objects, and link them using the SQL SELECT, WHERE, and ORDER BY clauses. Once you create a SQL query you can define any variables using the Variables portion of the dialog box.

The following examples describe two SQL statements and the steps for creating them using the Advanced Recordset dialog box’s Database Items tree.

Selecting a table

This example selects the entire contents of the Employees table. The SQL statement defining the query appears as:

```
SELECT * FROM Employees
```

To create this query:

- 1 Expand the Tables branch to display all of the tables in the selected database.
- 2 Select the Employees table.
- 3 Click the Select button.
- 4 Click OK to add the recordset to the Bindings panel.

Selecting specific rows from a table and ordering the results

The following example selects two rows from the Employees table, and selects the job type using a variable that you must define. The results are then ordered by employee name.

```
SELECT emp1No, emp1Name  
FROM Employees  
WHERE emp1Job = 'varJob'  
ORDER BY emp1Name
```

To create this query:

- 1 Expand the Tables branch to display all of the tables in the selected database, then expand the Employees table to display the individual table rows.
- 2 Build the SQL statement as follows:
 - Select emp1No and click the Select button.
 - Select emp1Name and click the Select button.
 - Select emp1Job and click the Where button.
 - Select emp1Name and click the Order By button.
- 3 Place the insertion point after WHERE emp1Job in the SQL text area and type = 'varJob' (include the equal sign).
- 4 Define the variable 'varJob' by clicking the plus (+) button in the Variables area and entering the following values in the Name, Default Value, and Run-Time Value columns: varJob, CLERK, Request("job").
- 5 Click OK to add the recordset to the Bindings panel.

Defining URL parameters

URL parameters store retrieved information input by users. To define a URL parameter you create a form or hypertext link that uses the GET method to submit data. The information is appended to the URL of the requested page and communicated to the server. When using URL variables, the query string contains one or more name/value pairs that are associated with the form fields. These name/value pairs are appended to the URL.

Before you begin, make sure you pass a form or URL parameter to the server. For instructions, see “Understanding form parameters” on page 493.

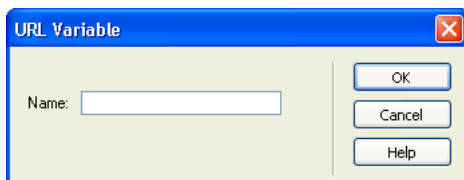
To define a URL parameter:

- 1 In the Document window, open the page that will use the variable.
- 2 Choose Windows > Bindings to display the Bindings panel.

- 3 In the Bindings panel click the plus (+) button and choose one of the following from the pop-up menu:

Document Types	Menu item in Bindings panel for URL variable
ASP	Request Variable > Request.QueryString
ColdFusion	URL Variable
JSP	Request Variable
PHP	URL Variable

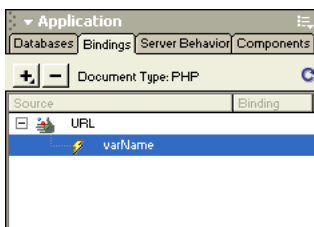
The URL Variable dialog box is displayed.



- 4 Enter the name of the URL variable in the text field and click OK.

The URL variable name is normally the name of the HTML form field or object used to obtain its value.

- 5 The URL variable appears in the Bindings panel.



Once you define the URL variable, you can use its value in the currently selected page. For more information, see Chapter 33, “Adding Dynamic Content to Web Pages,” on page 519.

Defining form parameters

Form parameters store retrieved information that is included in the HTTP request for a web page. If you create a form that uses the `POST` method, the data submitted by the form is passed to the server. Before you begin, make sure you pass a form parameter to the server. For instructions, see Chapter 31, “Storing and Retrieving Data for Your Page,” on page 491.

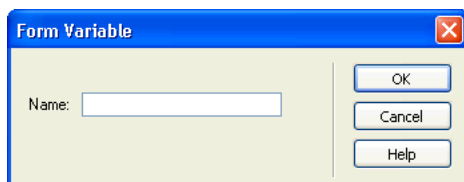
To define a form parameter:

- 1 In the Document window, open the page that will use the variable.
- 2 Choose **Windows > Bindings** to display the Bindings panel.

- 3 In the Bindings panel click the plus (+) button and choose one of the following from the pop-up menu:

Document Types	Menu item in Bindings panel for form variable
ASP	Request Variable > Request.Form
ColdFusion	Form Variable
JSP	Request Variable
PHP	Form Variable

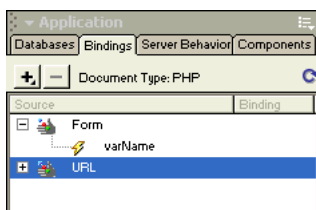
The Form Variable dialog box is displayed:



- 4 Enter the name of the form variable in the dialog box and click OK.

The form parameter name is normally the name of the HTML form field or object used to obtain its value.

- 5 The form parameter appears in the Bindings panel.



After you define the form parameter as a content source, you can use its value in your page. For more information, see Chapter 31, “Storing and Retrieving Data for Your Page,” on page 491.

Defining session variables

You can use session variables to store and display information maintained for the duration of a user’s visit (or session). The server creates a different session object for each user and maintains it for a set period of time or until the object is explicitly terminated. For more information, see “Accessing data stored in session variables” on page 495.

Because session variables last throughout the user’s session and persist when the user moves from page to page within the website, they’re ideal for storing user preferences. Session variables can also be used for inserting a value in the page’s HTML code, assigning a value to a local variable, or providing a value to evaluate a conditional expression.

Before defining session variables for a page, you must create them in the source code. For instructions, see the following sections:

- “Collecting data submitted by users” on page 492
- “Accessing data stored in session variables” on page 495

After you create a session variable in the web application’s source code, you can use Dreamweaver to retrieve its value and use it in a web page.

To define a predefined session variable:

- 1 Create a session variable in the source code and assign a value to it.

For example, this ASP example instantiates a session called `username`, and assigns it the value `Cornelius`:

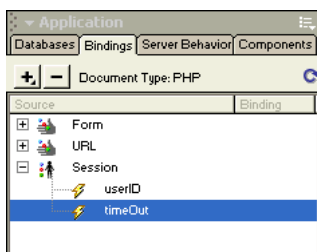
```
<% Session(username) = "Cornelius" %>
```

- 2 Choose **Window > Bindings** to display the Bindings panel.
- 3 Click the plus (+) button and choose **Session Variable** from the pop-up menu.
- 4 Enter the name of the variable you defined in the source code.

In this example, it is `username`.

- 5 Click **OK**.

The session variable appears in the Bindings panel.



Once you define the session variable, you can use its value in your page. For more information, see Chapter 33, “Adding Dynamic Content to Web Pages,” on page 519.

Defining application variables

In ASP and ColdFusion, you can use application variables to store and display information that is maintained for the lifetime of the application and persists from user to user. The application’s lifetime lasts from the time the first user requests a page in the application to the time the web server is stopped. (An application is defined as all the files in a virtual directory and its subdirectories.)

Because application variables last for the lifetime of the application, and persist from user to user, they’re ideal for storing information that must exist for all users, such as the current time and date. The value of the application variable is defined in the application’s code.

Note: There are no application objects in JSP or PHP.

To define an application variable for a page:

- 1 Open a dynamic document type in the Document window.
- 2 Choose Window > Bindings to display the Bindings panel.
- 3 Click the plus (+) button and select Application Variable from the pop-up menu.
- 4 Enter the name of the variable as defined in the application's source code.
- 5 Click OK.

The application variable appears in the Bindings panel.

Once you define the application variable, you can use its value in a page. For more information, see Chapter 33, "Adding Dynamic Content to Web Pages," on page 519.

Defining server variables

You can define server variables as sources of dynamic content for use within a web application.

Server variables vary from document type to document type and include form variables, URL variables, session variables, and application variables. For more information on these variable types, see the following sections:

- "Defining URL parameters" on page 507
- "Defining form parameters" on page 508
- "Defining session variables" on page 509
- "Defining application variables" on page 510

This section discusses server variables for different document types that you can define as content sources. The document types and their related sections are:

- "Defining ASP server variables" on page 511
- "Defining ColdFusion server variables" on page 512
- "Defining JSP server variables" on page 513
- "Defining JavaBeans (JSP only)" on page 514

Defining ASP server variables

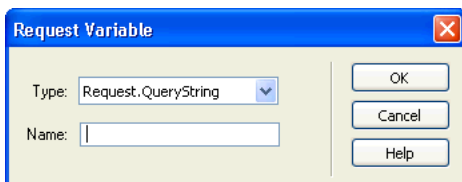
You can define the following ASP server variables as sources of dynamic content:

`Request.Cookie`, `Request.QueryString`, `Request.Form`, `Request.ServerVariables`, and `Request.ClientCertificates`.

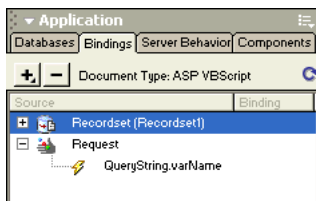
To define a server variable for an ASP page:

- 1 Open the Bindings (Window > Bindings).

- 2 Click the plus (+) button and choose Request Variable from the pop-up menu.
The Request Variable dialog box displays.



- 3 Choose one of the request collections from the Type pop-up menu.
For example, if you want to access the information in the `Request.ServerVariables` collection, choose Server Variables. If you want to access the information in the `Request.Form` collection, choose Form.
For more information on the ASP server variables, click the Help button.
- 4 Specify the variable in the collection that you want to access.
For example, if you want to access the information in the `Request.ServerVariables("HTTP_USER_AGENT")` variable, enter the argument `HTTP_USER_AGENT`. If you want to access the information in the `Request.Form("lastname")` variable, enter the argument `lastname`.
- 5 Click OK.
The server variable appears in the Bindings panel.



Defining ColdFusion server variables

You can define the following ColdFusion server variables as sources of dynamic content: client, cookie, CGI, server, and local variables.

Client variables associate data with a specific client. Client variables maintain the application's state as the user moves from page to page in the application, as well as from session to session. Maintaining state means to preserve information from one page (or session) to the next so that the application "remembers" the user and the user's previous choices and preferences.

Cookie variables access cookies passed to the server by the browser.

CGI variables provide information about the server running ColdFusion, the browser requesting a page, and other information about the processing environment.

Server variables can be accessed by all clients and applications on the server. They persist until the server is stopped.

Local variables are created with the `CFSET` tag or `CFPARAM` tag within a ColdFusion page.

To define a server variable for a ColdFusion page:

- 1 Open the Bindings panel (Window > Bindings).
- 2 Click the plus (+) button and choose the server variable from the pop-up menu.
- 3 Enter the name of the variable.
For more information on possible name entries, click the Help button.
- 4 Click OK.

The ColdFusion server variable appears in the Bindings panel.

For more information, see:

- “Defining URL parameters” on page 507
- “Defining session variables” on page 509
- “Defining application variables” on page 510
- Chapter 33, “Adding Dynamic Content to Web Pages,” on page 519

Defining PHP server variables

To define a server variable for a PHP page:

- 1 Open the Bindings panel (Window > Bindings).
- 2 Click the plus (+) button and choose the variable from the pop-up menu.
- 3 Enter the name of the variable.
- 4 Click OK.

The PHP server variable appears in the Bindings panel.

For more information, see Chapter 32, “Defining URL parameters,” on page 507, Chapter 33, “Adding Dynamic Content to Web Pages,” on page 519.

Defining JSP server variables

You can define a request variable as a source of dynamic content for JSP pages.

To define a request variable for a JSP page:

- 1 Open the Bindings panel (Window > Bindings).
- 2 Click the plus (+) button and choose Request Variable from the pop-up menu.
- 3 Enter the name of the variable.
- 4 Click OK.

The JSP server variable appears in the Bindings panel.

For more information see “Defining URL parameters” on page 507, and Chapter 33, “Adding Dynamic Content to Web Pages,” on page 519.

Defining JavaBeans (JSP only)

JavaBeans are architectural elements of multi-tier JSP applications. JavaBeans are typically used as part of a middle “business-logic” layer meant to separate the presentation logic from data-access logic. In these applications, the JavaBeans, (also referred to as “beans”) not the JSP pages, contain the logic that directly accesses the database.

In Dreamweaver, JavaBeans are treated as sources of dynamic content for JSP pages, and appear in the Bindings panel. You can double-click JavaBeans in the Bindings panel to view their properties, and then drag the properties to the page to create dynamic data references.

You can also define a JavaBeans collection (a set of JavaBeans) as a source of dynamic content. However, Dreamweaver supports only repeated regions and dynamic bindings when using JavaBeans collections.

Copies of the bean class (or of the .zip or .jar file containing the bean class) must reside in the following locations:

- On the system running Dreamweaver, a copy of the bean class must reside in the Dreamweaver Configuration/classes folder or in the system’s classpath. (Dreamweaver uses this copy of the class at design time.)
- On the system running the JSP application server, the bean class must reside in the application server’s classpath. (Your application server uses this copy of the class at runtime.) The application server’s classpath varies from application server to application server, but generally the classpath is to a WEB-INF folder with a classes/bean sub-folder.

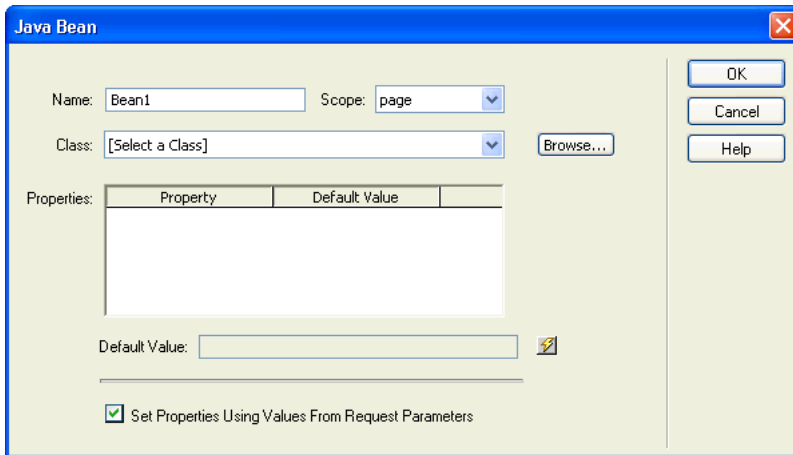
If Dreamweaver and the application server are both running on the same system, and the application server uses the system classpath (not an internal classpath), a single copy of the JavaBeans’ class can reside on the computer in the system classpath. Both the application server and Dreamweaver will use this copy of the class. Otherwise, copies of the JavaBeans class must reside in two paths on the computer (as described above).

The folder structure must match the JavaBeans’ package. For example, if the JavaBeans’ package is called `com.ardvark.myBean`, then the package must be stored in `/com/ardvark/` within the classpath or in Dreamweaver’s Configuration/classes folder.

To define a JavaBean for a JSP page:

- 1 Choose Window > Bindings to display the Bindings panel.
- 2 Click the plus (+) button and choose JavaBean from the pop-up menu.

The JavaBean dialog box appears.



- 3 Complete the dialog box and click OK.

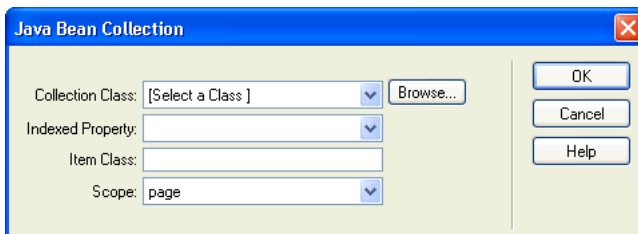
For instructions on completing the JavaBean dialog box click the dialog box's Help button.

- 4 The newly defined bean appears in the Bindings panel.

To define a JavaBean collection for a JSP page:

- 1 Choose Window > Bindings to display the Bindings panel.
- 2 Click the plus (+) button and choose JavaBean Collection from the pop-up menu.

The JavaBean Collection dialog box appears.



- 3 Complete the dialog box and click OK.

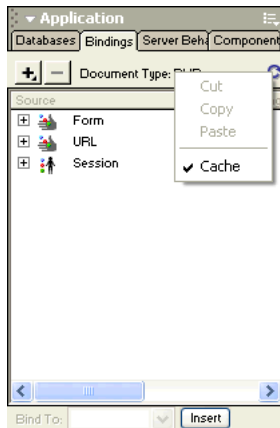
For instructions on completing the JavaBean dialog box click the dialog box's Help button.

- 4 The newly defined JavaBean collection appears in the Bindings panel.

Caching content sources

You can cache—or store—sources of dynamic content in a Design Note. This lets you work on a site even if you don't have access to the database or application server storing the sources of dynamic content. Caching may also speed up development by eliminating repeated access across a network to the database and application server.

To cache content sources, click the arrow button in the top right corner of the Bindings panel and toggle Cache in the pop-up menu.



If you make changes to one of the content sources, you can refresh the cache by clicking the Refresh button (the circle-arrow icon) in the top right corner of the Bindings panel. (Expand the panel if you don't see the button.)

Changing or deleting content sources

You can change or delete any existing source of dynamic content—that is, any content source listed in the Bindings panel.

Changing or deleting a content source in the Bindings panel does *not* change or delete any instance of that content on the page. It merely changes or deletes it as a possible source of content for the page. To edit or delete an instance of the content on the page, see “Changing dynamic content” on page 526 and “Deleting dynamic content” on page 526.

To change a source of content in the Bindings panel:

- 1 In the Bindings panel, double-click the name of the content source you want to edit.
- 2 Make your changes in dialog box that appears.
- 3 If satisfied with your work, click OK.

To delete a source of content from the Bindings panel:

- 1 In the Bindings panel, select the content source from the list.
- 2 Click the minus (-) button.

Copying a recordset from one page to another page

You can copy a recordset from one page to another within a defined site.

To copy a recordset to another page:

- 1 Select the recordset in either the Bindings panel or the Server Behaviors panel.
- 2 Right-click the recordset and choose Copy from the pop-up menu.
- 3 Open the page you want to copy the recordset to.
- 4 Right-click the Bindings panel or the Server Behaviors toolbar, and choose Paste from the pop-up menu.

CHAPTER 33

Adding Dynamic Content to Web Pages

After you define one or more sources of dynamic content, you can use the sources to add dynamic content to the page. Content sources can include a column in a recordset, a value submitted by an HTML form, the value contained in a server object, or other data. For more information, see Chapter 32, “Defining Sources of Dynamic Content,” on page 501.

In Macromedia Dreamweaver MX, you can place dynamic content almost anywhere in a web page or its HTML source code:

- You can place dynamic content at the insertion point.
- You can replace a text string with dynamic content.
- You can insert it in an HTML attribute.

For example, dynamic content can define the `src` attribute of an image, or the `value` attribute of a form field.

This chapter contains the following sections:

- “Making text dynamic” on page 520
- “Making images dynamic” on page 521
- “Making HTML attributes dynamic” on page 523
- “Making ActiveX, Flash, and other object parameters dynamic” on page 525
- “Changing dynamic content” on page 526

About Adding Dynamic Content

You add dynamic content to a page by choosing one of your content sources in the Bindings panel. Dreamweaver inserts a server-side script in the page's source code instructing the server to transfer the data from the content source to the page's HTML source code when the page is requested by a browser.

There is often more than one way to make a given page element dynamic. For example, to make an image dynamic you can use the Bindings panel, the Property inspector, or the Image command in the Insert menu. This chapter describes the most efficient ways of making various page elements dynamic.

By default, an HTML page can display only one record at a time. To display the other records in the recordset, you can add a link to move through the records one at a time (see “Creating recordset navigation links” on page 529), or you can create a repeated region to display more than one record on a single page (see “Displaying multiple behaviors” on page 533).

Making text dynamic

You can replace existing text with dynamic text, or you can place dynamic text at a given insertion point on the page.

Dynamic text adopts any text formatting applied to the existing text or to the insertion point. For example, if a Cascading Style Sheet (CSS) style affects the selected text, the dynamic content replacing it is also affected by the style. You can add or change the text format of dynamic content by using any of the Dreamweaver text formatting tools. “Applying typographic and page layout elements to dynamic data” on page 528.

You can also apply a data format to dynamic text. For example, if your data consists of dates, you can specify a particular date format such as 04/17/00 for U.S. visitors, or 17/04/00 for Canadian visitors. “Applying formats to data” on page 528.

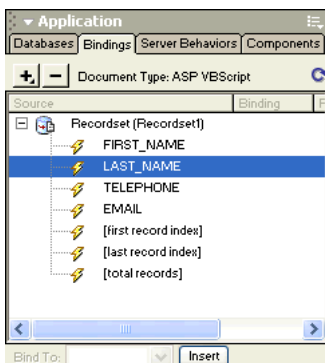
You can replace regular text on your page with dynamic text, or you can add dynamic text at the insertion point on the page.

To add dynamic text:

- 1 Open the Bindings panel by choosing Window > Bindings.
- 2 Make sure the Bindings panel lists the content source you want to use.

The content source should contain plain text (ASCII text). Plain text includes HTML. If no content sources appear in the list, or if the available content sources don't meet your needs, click the plus (+) button to define a new content source. See Chapter 32, “Defining Sources of Dynamic Content,” on page 501.
- 3 In Design view, select text on the page, or click where you want to add dynamic text.

- 4 In the Bindings panel, select a source of content from the list. If you select a recordset, specify the column you want in the recordset.



- 5 Click Insert, or drag the content source onto the page.

The dynamic content appears on the page if you're working in Design view with Live Data turned on (View > Live Data).

If Live Data is turned off, a placeholder appears instead. (If you selected text on the page, the placeholder replaces the text selection.) The placeholder for recordset content uses the syntax `{RecordsetName.ColumnName}`, where `RecordsetName` is the name of the recordset and `ColumnName` is the name of the column you chose from the recordset.

Sometimes, the length of the placeholders for dynamic text distorts the page's layout in the Document window. You can solve the problem by using empty curly braces as placeholders.

To create dynamic text placeholders:

- 1 Choose Edit > Preferences > Invisible Elements or Dreamweaver > Preferences > Invisible Elements (Mac OS X).
- 2 In the Show Dynamic Text As pop-up menu, choose `{ }`.
- 3 Click OK.

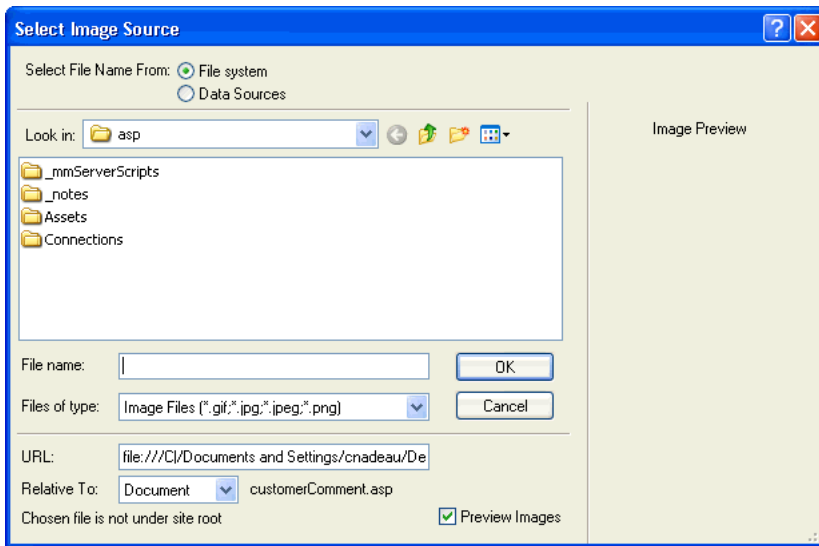
Making images dynamic

You can make images on your page dynamic. For example, suppose you design a page to display items for sale at a charity auction. Each page would include descriptive text and a photo of one item. The page's general layout would remain the same for each item, but the photo (and descriptive text) could change.

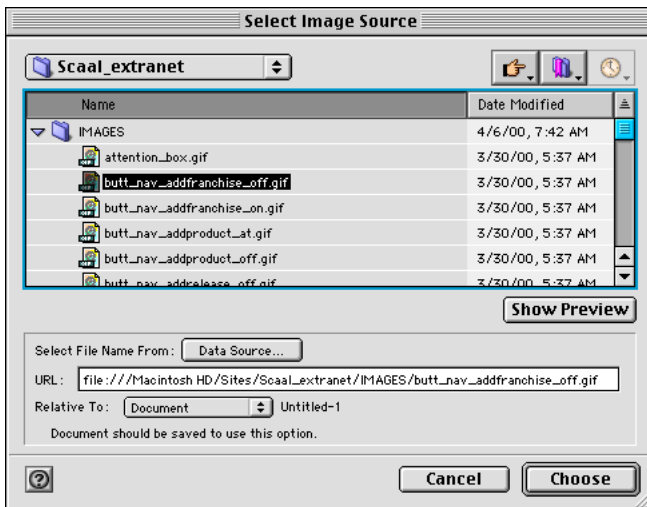
To make an image dynamic:

- 1 With the page open in Design view (View > Design), place the insertion point where you want the image to appear on the page, then select Insert > Image.

The Select Image Source dialog box appears.



On the Macintosh, the dialog box is different:



- 2 Click the Data Sources option (Windows) or the Data Source button (Macintosh).

A list of content sources appears.

- 3 Select a content source from the list.

The content source should be a recordset containing the paths to your image files. Depending on the file structure of your site, the paths can be absolute, document relative, or root relative.

Note: Dreamweaver MX does not currently support binary images stored in a database.

If no recordsets appear in the list, or if the available recordsets don't meet your needs, define a new recordset. For instructions, see "Defining a recordset" on page 502.

- 4 Click OK.

Making HTML attributes dynamic

You can dynamically change the appearance of a page by binding HTML attributes to data. For example, you can change the background image of a table by binding the table's background attribute to a field in a recordset.

You can bind HTML attributes with the Bindings panel or with the Property inspector. The two methods are described below.

To make HTML attributes dynamic with the Bindings panel:

- 1 Open the Bindings panel by choosing Window > Bindings.

- 2 Make sure the Bindings panel lists the data source you want to use.

The content source should contain data that's appropriate for the HTML attribute you want to bind. If no sources of content appear in the list, or if the available content sources don't meet your needs, click the plus (+) button to define a new data source. For instructions, see Chapter 32, "Defining Sources of Dynamic Content," on page 501.

- 3 In Design view, select an HTML object.

For example, to select an HTML table, click inside the table and click the `<table>` tag in the tag selector on the bottom-left of the Document window.

- 4 In the Bindings panel, select a content source from your list.
- 5 In the Bind To box, select an HTML attribute from the pop-up menu.
- 6 Click Bind.

The next time the page runs on the application server, the value of the data source will be assigned to the HTML attribute.

To make HTML attributes dynamic with the Property inspector:

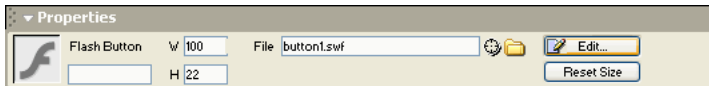
- 1 In Design view, select an HTML object and open the Property inspector (Window > Properties).

For example, to select an HTML table, click inside the table and click the `<table>` tag in the tag selector on the bottom-left of the Document window.

- 2 If the attribute you want to bind has a folder icon next to it in the Property inspector's Standard view, click the folder icon to open a file selection dialog box, then click the Data Sources option to display a list of data sources. Skip to step 6.

- 3 If the attribute you want to bind does not have a folder icon next to it in the Standard view, click the List tab (the lower of the two tabs) on the left side of the inspector.

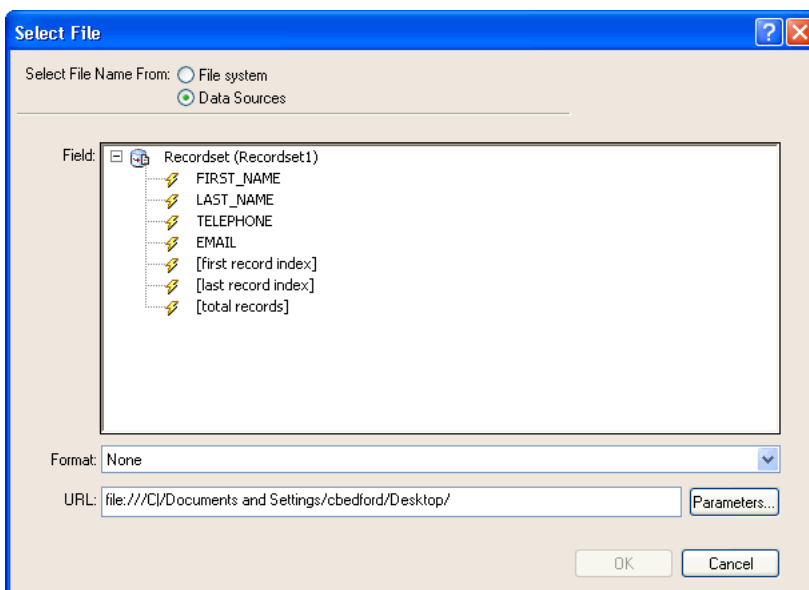
The Property inspector's List view appears.



- 4 If the attribute you want to bind is not listed in the List view, click the plus (+) button, then enter the attribute's name or click the small arrow button and select the attribute from the pop-up menu.
- 5 To make the attribute's value dynamic, click the attribute, then click the lightning-bolt icon or folder icon at the end of the attribute's row.

If you clicked the lightning-bolt icon, a list of data sources appears.

Here's an example:



If you clicked the folder icon, a file selection dialog box appears. Click the Data Sources option to display a list of content sources.

- 6 Select a source of content from the list of content sources.

The content source should hold data that's appropriate for the HTML attribute you want to bind. If no content sources appear in the list, or if the available content sources don't meet your needs, define a new content source. For instructions, see Chapter 31, "Storing and Retrieving Data for Your Page," on page 491.

- 7 Click OK.

The next time the page runs on the application server, the value of the data source will be assigned to the HTML attribute.

Making ActiveX, Flash, and other object parameters dynamic

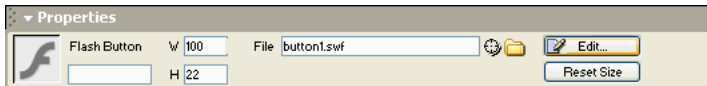
You can make the parameters of Java applets and plug-ins dynamic, as well as the parameters of ActiveX, Flash, Shockwave, Director, and Generator objects.

Before starting, make sure the fields in your recordset hold data that's appropriate for the object parameters you want to bind.

To make object parameters dynamic:

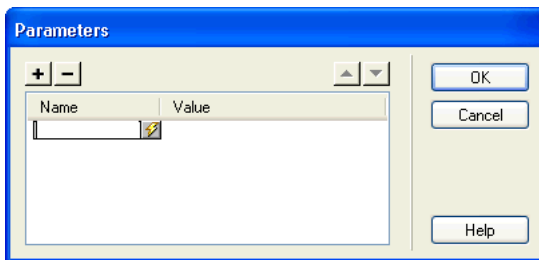
- 1 In Design view, select an object on the page and open the Property inspector (Window > Properties).

Here's the Property inspector of a Flash object:



- 2 Click the Parameters button.

The Parameters dialog box appears.



- 3 If your parameter does not appear in the list, click the plus (+) button and enter a parameter name in the Parameter column.
- 4 Click the parameter's Value column, then click the lightning-bolt icon to specify a dynamic value. A list of data sources appears.
- 5 Select a data source from the list.
The data source should hold data that's appropriate for the object parameter you want to bind. If no data sources appear in the list, or if the available data sources don't meet your needs, define a new data source. For instructions, see Chapter 32, "Defining Sources of Dynamic Content," on page 501.
- 6 Click OK.

Changing dynamic content

You can change the dynamic content on your page by editing the server behavior that provides the content. For example, you can edit a recordset server behavior to provide more records to your page.

Dynamic content on a page is listed in the Server Behaviors panel. For example, if you add a recordset to your page, the Server Behaviors panel lists it as follows:

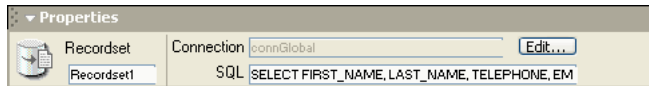
```
Recordset(myRecordset)
```

If you add another recordset to your page, the Server Behaviors panel lists both recordsets as follows:

```
Recordset(mySecondRecordset)  
Recordset(myRecordset)
```

To edit a server behavior providing dynamic content, double-click the server behavior in the Server Behaviors panel. The same dialog box you used to define the original data source appears. Make your changes in the dialog box and click OK.

You can also use the Property inspector to edit the recordsets on your page. Open the Property inspector (Window > Properties), then select the recordset in the Server Behaviors panel (Window > Server Behavior). Here's the Property inspector for a recordset:



If you edit a recordset in the Live Data window with the Auto Refresh option not selected, you must refresh the page to see your changes. To refresh the page, click the Refresh button or choose View > Refresh Live Data.

Deleting dynamic content

After adding dynamic content to a page, you can delete it by selecting the dynamic content on the page and pressing Delete. You can also delete it by selecting the dynamic content in the Server Behaviors panel and clicking the minus (-) button.

Note: This operation removes the server-side script in the page that retrieves the dynamic content from the database. It does not delete the data in the database.

CHAPTER 34

Displaying Database Records

Displaying database records involves retrieving information stored in a database or other source of content, and rendering that information to a web page. Macromedia Dreamweaver MX provides many methods of displaying dynamic content, and provides several built-in server behaviors that let you enhance both the presentation of dynamic content, and allow users to more easily search through and navigate information returned from a database.

Dreamweaver provides the following server behaviors and formatting elements to let you enhance the display of dynamic data:

Formats let you apply different types of numerical, monetary, date/time, and percentage values to dynamic text.

Repeated Region server behaviors let you display multiple items returned from a database query, and let you specify the number of records to display per page.

Recordset Navigation server behaviors let you insert navigation elements that allow users to move to the next or previous set of records returned by the recordset. For example, if you choose to display 10 records per page using the Repeated Region server object, and the recordset returns 40 records, you can navigate through the records 10 at a time.

Recordset Status Bar server behaviors let you include a counter that shows users where they are within a set of records relative to the total number of records returned.

Show Region server behaviors lets you choose to show or hide items on the page based on the relevance of the currently displayed records. For example, if a user has navigated to the last record in a recordset, you can hide the “next” link, and display only the “previous” records link.

This chapter contains the following sections:

- “Applying typographic and page layout elements to dynamic data” on page 528
- “Applying formats to data” on page 528
- “Creating recordset navigation links” on page 529
- “Showing and hiding regions based on recordset results” on page 532
- “Displaying multiple behaviors” on page 533
- “Creating an ASP.NET DataGrid or DataList web control” on page 538
- “Creating a record counter” on page 535

Applying typographic and page layout elements to dynamic data

A powerful feature of Dreamweaver is the ability to present dynamic data within a structured page, and to apply typographic formatting using HTML and CSS. To apply formats to dynamic data in Dreamweaver, format the tables and placeholders for the dynamic data using Dreamweaver's formatting tools. When the data is inserted from its data source, it will automatically adopt the font, paragraph, and table formatting you specified.

To learn about Dreamweaver's formatting features, and how to apply them to dynamic data elements, see Chapter 16, "Presenting Content with Tables," on page 227 and Chapter 19, "Inserting and Formatting Text," on page 271.

Applying formats to data

Dreamweaver comes with several predefined data formats that you can apply to dynamic data elements. The data format styles include date and time, currency, numerical, and percentage formats.

For example, if the price of an item in a recordset reads 10.989, you can display the price on the page as \$10.99 by selecting Dreamweaver's "Currency - 2 Decimal Places" format. This format takes and displays a number using two decimal places. If the number has more than two decimal places, the data format rounds the number to the closest decimal. If the number has no decimal places, the data format adds a decimal point and two zeros.

To apply data formats to dynamic content:

- 1 Select the dynamic content in either the Live Data window or its placeholder in the Document window.
- 2 Choose Window > Bindings to display the Bindings panel.
- 3 Click the down arrow button in the Format column.
If the down arrow is not visible, expand the panel.
- 4 From the Format pop-up menu, select the data format category you want.
Ensure that the data format is appropriate for the type of data you are formatting. For example, the Currency formats work only if the dynamic data consists of numerical data. Note that you cannot apply more than one format to the same data.
- 5 Verify that the format was applied correctly by previewing the page in either the Live Data window or a browser.

Customizing existing data formats

You can customize Dreamweaver's existing data formats or create your own.

To customize a data format:

- 1 Open a page that contains dynamic data in Design view.
- 2 Select the dynamic data whose format you want to customize.
- 3 Choose Window > Bindings to display the Bindings panel.
The bound data item whose dynamic text you selected will be highlighted.
- 4 Click the down arrow in the Format column to expand the pop-up menu of available data formats.
If the down arrow is not visible, expand the panel.

- 5 Select Edit Format List from the pop-up menu.
The Edit Format List dialog box appears.
- 6 Complete the dialog box and click OK.
For more information, click the Help button in the dialog box.

Creating new data formats

You can create new data formats to suit any type of dynamic data you might want to display.

To create a new data format:

- 1 Open a page containing dynamic data in Design view.
- 2 Select the dynamic data you want to create a custom format for.
- 3 Choose Window > Bindings to display the Bindings panel, and click the down arrow in the Format column.
If the down arrow is not visible, expand the panel.
- 4 Select Edit Format List from the pop-up menu.
The Edit Format List dialog box appears.
- 5 Click the plus (+) button and select a format type.
- 6 Define the format and click OK.
- 7 Enter a name for the new format in the Name column.
- 8 Click OK to close the Edit Format List dialog box.

Creating recordset navigation links

Recordset navigation links let users move from one record to the next, or from one set of records to the next. For example, after designing a page to display five records at a time, you might want to add links such as “Next” or “Previous” that let users display the five next or previous records.

Dreamweaver lets you create four types of navigation links to move through a recordset: First, Previous, Next, and Last. A single page can contain any number of these links, provided they all work on a single recordset. You can't add links to move through a second recordset on the same page.

Note: Microsoft ASP.NET refers to a recordset as a DataSet. If you are working with ASP.NET document types, the dialog boxes and menu choices specific to ASP.NET use the label DataSet. The Dreamweaver documentation generically refers to both types as recordsets, but uses DataSet when specifically describing ASP.NET features.

Recordset navigation links require the following dynamic elements:

- A recordset to navigate
- Dynamic content on the page to display the record or records
- Text or images on the page to serve as a clickable navigation bar
- A “Move To Record” set of server behaviors to navigate the recordset

You can add the last two elements using the Record Navigation Bar server object, or you can add them separately using the Dreamweaver design tools and the Server Behaviors panel.

Creating a navigation bar using the Recordset Navigation Bar server behavior

You can create a recordset navigation bar in a single operation using the Recordset Navigation Bar server behavior. The server object adds the following building blocks to the page:

- An HTML table with either text or image links
- A set of “Move to” server behaviors
- A set of “Show Region” server behaviors

The text version of the Recordset Navigation Bar looks like this:



The image version of the Recordset Navigation Bar looks like this:



Before placing the navigation bar on the page, make sure the page contains a recordset to navigate and a page layout in which to display the records.

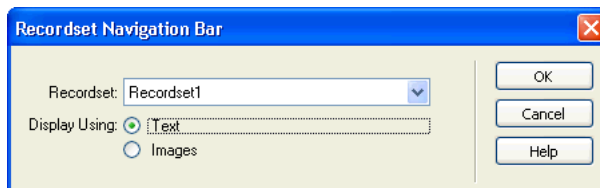
After placing the navigation bar on the page, you can use Dreamweaver’s design tools to customize the bar to your liking. You can also edit the “Move to” and “Show Region” server behaviors by double-clicking them in the Server Behaviors panel.

If you want to build the navigation bar block by block using the Dreamweaver design tools and the Server Behaviors panel, see “Creating a custom recordset navigation bar” on page 531.

To create the recordset navigation bar with the server object:

- 1 In Design view, place the insertion point at the location on the page where you want the navigation bar to appear.
- 2 Display the Recordset Navigation Bar dialog box (Insert > Application Objects > Recordset Navigation Bar).

The Insert Recordset Navigation Bar dialog box appears.



- 3 Select the recordset you want to navigate from the Recordset pop-up menu.
- 4 From the Display Using section, select the format to display the navigation links on the page. The Text option places text links on the page, while the Images option lets you use graphical images as links.

In the image version of the navigation bar, Dreamweaver uses its own image files. You can replace these images with image files of your own after placing the bar on the page.

5 Click OK.

Dreamweaver creates a table that contains text or image links that allow the user to navigate through the selected recordset when clicked. When the first record in the recordset is displayed, the “First” and “Previous” links or images are hidden. When the last record in the recordset is displayed, the “Next” and “Last” links or images are hidden.

You can customize the layout of the navigation bar using the Dreamweaver design tools.

Creating a custom recordset navigation bar

If you want to create a recordset navigation bar that uses more complex layout and formatting styles than that offered by the simple table created by the Recordset Navigation Bar server object, you may prefer to create your own navigation bar. To do this, you must first create the necessary navigation links in either text or images, place them within the page in design view, and assign individual server behaviors to each navigation link.

The individual server behaviors that you can assign to navigation links are:

- Move to first page
- Move to last page
- Move to next page
- Move to previous page

If you would prefer to use the Dreamweaver’s built-in Recordset Navigation Bar server object to create a navigation bar, see “Creating a navigation bar using the Recordset Navigation Bar server behavior” on page 530.

Navigation bar design tasks

When creating a custom navigation bar, begin by creating its visual representation using Dreamweaver’s page-design tools. You don’t have to create a link for the text string or image, Dreamweaver will create one for you.

The page you create the navigation bar for must contain a recordset to navigate. See “Defining a recordset” on page 502 for more information.

A simple recordset navigation bar might look like this, with link buttons created out of images, or other content elements:



After you have added a recordset to a page, and have created a navigation bar, you must apply individual server behaviors to each navigation element. For example, a typical recordset navigation bar contains representations of the following links matched to the appropriate behavior:

Navigation link	Server behavior
Go to first page	Move to first page
Go to the previous page	Move to previous page
Go to the next page	Move to next page
Go to the last page	Move to last page

To assign server behaviors to record navigation links:

- 1 In Design view, select the text string or image on the page you want to use as a record navigation link.
- 2 Open the Server Behaviors panel (Window > Server Behaviors) and click the plus (+) button.
- 3 Choose Recordset Paging from the pop-up menu, then select a server behavior appropriate to that link from the listed server behaviors.

Note that if the recordset contains a large number of records, the Move to Last Record server behavior can take a long time to run when the user clicks the link.

- 4 In the Recordset pop-up menu, select the recordset containing the records.
- 5 Click OK.

Showing and hiding regions based on recordset results

Dreamweaver includes a set of server behaviors that let you show or hide a region based on the results returned by a recordset. For example, in a page using “Previous” and “Next” record links to navigate a results page, you can specify that the “Previous” records link be shown on all results pages except the first (which has no previous results), and that the “Next” records link be shown on all pages except the last (which has no next results).

You can also specify that a region be displayed or hidden based on whether the recordset is empty or not. If a recordset is empty (for example, no records were found matching the query), you can display a message informing the user that no records were returned. This is especially useful when creating search pages that rely on user input search terms to run queries against. Similarly, you can display an error message if there is a problem connecting to a database, or if a user’s user name and password do not match those recognized by the server.

The Show Region server behaviors are:

- Show If Recordset Is Empty
- Show If Recordset Is Not Empty
- Show If First Page
- Show If Not First Page
- Show If Last Page
- Show If not Last Page

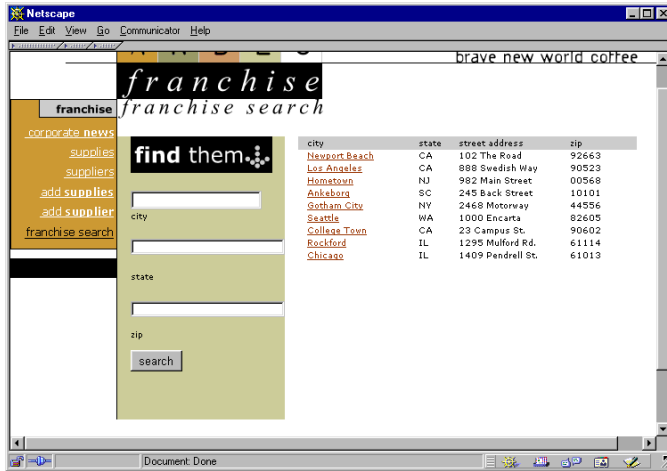
To show a region only when it’s needed:

- 1 In Design view, select the region on the page to show or hide.
- 2 In the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button.
- 3 Choose Show Region from the pop-up menu, then choose one of the listed server behaviors.
- 4 Click OK.

Displaying multiple behaviors

The Repeat Region server behavior lets you display multiple records from a recordset within a page. Any dynamic data selection can be turned into a repeated region. However, the most common regions are a table, a table row, or a series of table rows.

The example below illustrates how the Repeat Region server behavior is applied to a table row, and specifies that nine records be displayed per page. The row itself displays four different records: city, state, street address, and zip code.



To create a table such as the one shown above, you must create a table containing dynamic content, and apply the Repeat Region server behavior to the table row containing the dynamic content. When the page is processed by the application server, the row is repeated the number of times specified in the Repeat Region server object, with a different record inserted in each new row.

The page you create the navigation bar for must contain a recordset to navigate. For more information, see “Defining a recordset” on page 502.

To create a repeated region:

- 1 In Design view, select a region that contains dynamic content.

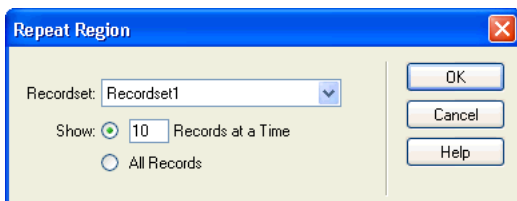
The selection can be anything, including a table, a table row, or even a paragraph of text.

To select a region on the page precisely, you can use the tag selector on the left corner of the document window. For example, if the region is a table row, click inside the row on the page, then click the rightmost `<tr>` tag in the tag selector to select the table row.

- 2 Choose Window > Server Behaviors to display the Server Behaviors panel.

- 3 Click the plus (+) button, and select Repeat Region.

The Repeat Region dialog box appears.



- 4 Select the name of the recordset to use from the pop-up menu.
- 5 Select the number of records to display per page.
- 6 Click OK.

In the Document window, a thin, tabbed gray outline appears around the repeated region. In the Live Data window (View > Live Data), the gray outline disappears and the selection expands to display the number of records you specified.

Creating a table with a Repeat Region server behavior

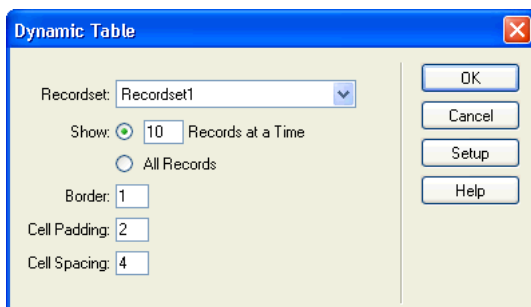
The Dynamic Table server object lets you create a table containing dynamic content and apply the Repeat Region behavior from a single dialog box. This server object is especially useful as it simultaneously populates a table with dynamic content from a recordset and applies the Repeat Region server behavior.

Note: The Dynamic Table server object is not available when using ASP.NET document types. To create a table containing dynamic content and repeating regions, you must manually insert the table, insert dynamic content from the Data Binding panel, and apply a repeating region server behavior if needed. For more information, see “Displaying multiple behaviors” on page 533.

To create a dynamic table:

- 1 Choose Insert > Application Objects > Dynamic table to display the Dynamic Table dialog box.

The Dynamic Table dialog box appears.



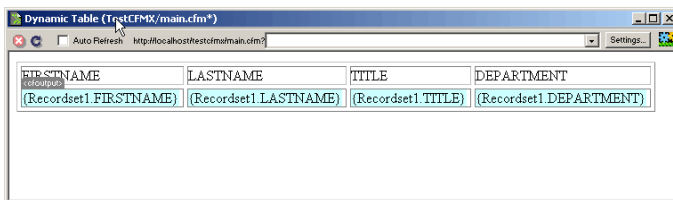
- 2 Select the recordset you want to use from the Recordset pop-up menu.
- 3 Select the number of records to display per page.

- 4 You may at this time input values for the table border, cell padding, and cell spacing.

The Dynamic Table dialog retains the values you enter for table borders, cell padding, and cell spacing. If you are working on a project that will need several dynamic tables requiring the same look, you may want to enter the table layout values, as this will further simplify page development. Note that you can adjust these values after inserting the table using the table Property inspector.

- 5 Click OK.

A table and placeholders for the dynamic content defined in its associated recordset are inserted into the page.



In this example the recordset contains four records: FIRSTNAME, LASTNAME, TITLE, and DEPARTMENT. The table's Heading row is populated with the names of each record item. You can edit the headings using any descriptive text you want, or replace them with representative images.

Creating a record counter

Record counters give users a reference point when they are navigating through a set of records. Typically, record counters display the total number of records returned, and the current records being viewed. For example, if a recordset returns 40 individual records, and 8 records are displayed per page, the record counter on the first page would indicate "Displaying records 1-8 of 40."

To create a record counter for a page, you must first create a recordset for the page, an appropriate page layout to contain the dynamic content, and a recordset navigation bar. To learn more about creating these elements, and adding them to a page, see the following sections:

- "Defining a recordset" on page 502
- "Creating recordset navigation links" on page 529
- "Displaying multiple behaviors" on page 533
- "Creating a table with a Repeat Region server behavior" on page 534

Once you have the above elements in the page, you can create a record counter.

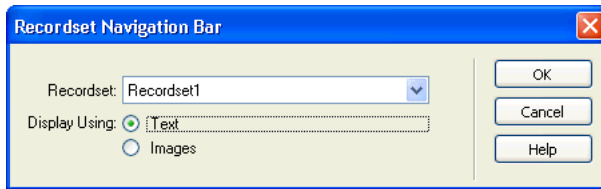
Creating a record counter using the Recordset Navigation Status object

The Recordset Navigation Status object creates a text entry on the page to display the current record status.

To use the Recordset Navigation Status server object:

- 1 Place the cursor at the point where you want to insert the record counter.
- 2 Choose Insert > Application Objects > Recordset Navigation Status.

The Insert Recordset Navigation Status dialog box is displayed.



Select the recordset you want to use from the Recordset pop-up menu.

- 3 Click OK.

The Recordset Navigation Status server object inserts a text record counter that appears similar to the one shown below:

Records {Employees_first} to {Employees_last} of {Employees_total}

You can use Dreamweaver's page-design tools to customize the record counter.

When viewed in the Live Data window or a browser the counter will appear similar to the one shown below:

Records 1 to 1 of 22

Creating custom record counters

You can use individual record count behaviors to create custom record counters. Creating a custom record counter allows you to create a record counter beyond the simple, single row table inserted by the Recordset Navigation Status server object. You can arrange design elements in a number of creative ways, and apply an appropriate server behavior to each element.

You can create a simple record counter using the Recordset Navigation Status server object. This server object inserts a complete record counter that you can apply text formatting to using Dreamweaver's page-design tools.

The Record Count server behaviors are:

- Display Starting Record Number
- Display Ending Record Number
- Display Total Records

To create a custom record counter for a page, you must first create a recordset for the page, an appropriate page layout to contain the dynamic content, and a recordset navigation bar. To learn more about creating these elements, and adding them to a page, see the following sections:

- “Defining a recordset” on page 502
- “Creating recordset navigation links” on page 529
- “Displaying multiple behaviors” on page 533
- “Creating a table with a Repeat Region server behavior” on page 534

Once you have the above elements in the page, you can create a custom record counter.

This example creates a record counter that will appear similar to that created in the previous section “Creating a record counter using the Recordset Navigation Status object” on page 536. The record counter in this example will appear as:

Displaying records *StartRow* thru *EndRow* of *RecordSet.RecordCount*.

In this example the text in san-serif font represents the record count placeholders that will be inserted in the page.

To create a custom record counter:

- 1 In Design view, type the counter’s text on the page. The text can be anything you choose.

Displaying records thru of .

- 2 Place the insertion point at the end of the text string.

- 3 Open the Server Behaviors panel (Window > Server Behaviors).

- 4 Click the plus (+) button in the upper left corner, and click on Display Record Count. Within this submenu select Display Total Records. The Display Total Records behavior is inserted into the page, and a placeholder is inserted where the insertion point was. The text string should now appear as:

Displaying records thru of *{Recordset1.RecordCount}*.

- 5 Place the insertion point after the word *records*, and select the Display Starting Record Count Number from the Server Behaviors > plus (+) button > Record Count panel. The text string should now appear as:

Displaying records *{StartRow_Recordset1}* thru of *{Recordset1.RecordCount}*.

- 6 Now place the insertion point between the words *thru* and *of*, and select the Display Starting Record Count Number from the Server Behaviors > plus (+) button > Record Count panel. The text string should now appear as:

Displaying records *{StartRow_Recordset1}* thru *{EndRow_Recordset1}* of *{Recordset1.RecordCount}*.

- 7 Confirm that the counter functions correctly by viewing the page in the Live Data window (View > Live Data), the counter should look similar to the example below:

Displaying records 1 thru 8 of 40.

If the results page has a navigation link to move to the next set of records, clicking the link would update the record counter to read as follows:

Showing records 9 thru 16 of 40.

Links don't work in the Live Data window. To test them, you can use Dreamweaver's Preview in Browser feature. Make sure the Preview Using Live Data Server option is selected in Preferences (Edit > Preferences > Preview in Browser or Dreamweaver > Preferences > Preview in Browser (Mac OS X)), then select File > Preview in Browser.

Creating an ASP.NET DataGrid or DataList web control

The ASP.NET DataGrid and DataList controls provide numerous options for displaying different data types (especially dynamic content from a database), and simplify the process of binding data sources to the controls. Dreamweaver supports both the DataGrid and DataList controls as server behaviors. The controls provide the following features:

DataGrid creates a multi-column, data-bound grid. This control allows you to define various types of columns, both to layout the contents of the grid and to add specific functionality (edit button columns, hyperlink columns, and so on).

DataList displays items from a data source using templates. You can customize the appearance of the control by manipulating the templates that make up its different components.

Adding a DataGrid to a page

The Dreamweaver DataGrid allows you to insert an ASP.NET DataGrid web control. The DataGrid control renders tables as multi-column grids, and can include different column types (heterogeneous columns) for defining the layout of cell contents. These include bound, button, and template columns, among others. In addition, the DataGrid supports interactive functionality such as column sorting, editing, and commands. The column types available in the DataGrid are:

DataGrid Column Type	Description
Simple Data Field	Referred to as a "bound column" in ASP.NET, the Simple Data Field column lets you specify which data source field to display, and the data format the field will use with a .NET formatting expression.
Free Form	Referred to as a "template column" in ASP.NET, the Free Form column lets you create combinations of HTML text and server controls to design a custom layout for a column. The controls within a free form column can be data-bound. Free form columns give you added flexibility in defining the layout and functionality of the grid contents, because you have complete control over how the data is displayed and what happens when users interact with rows in the grid.
Hyperlink	The Hyperlink Column displays information as hypertext links. A typical use is to display data (such as a customer number or product name) as a hyperlink that users can click to navigate to a separate page that provides details about that item.

DataGrid Column Type	Description
Edit, Update, Cancel Buttons	Referred to as the Edit Command Column in ASP.NET the Edit, Update, Cancel Button column lets users perform in-place editing of information in DataGrid rows. To do so, create an Edit, Update, Cancel column. At runtime, this column displays a button labeled "Edit". When the user clicks the Edit button, the row data is displayed in editable controls such as text boxes, and the Edit button is replaced with Update and Cancel buttons.
Delete Buttons	The Delete Button lets a user delete a particular row by clicking a button.

To learn more about the DataGrid control, and how it can be used to format dynamic data, visit the following websites:

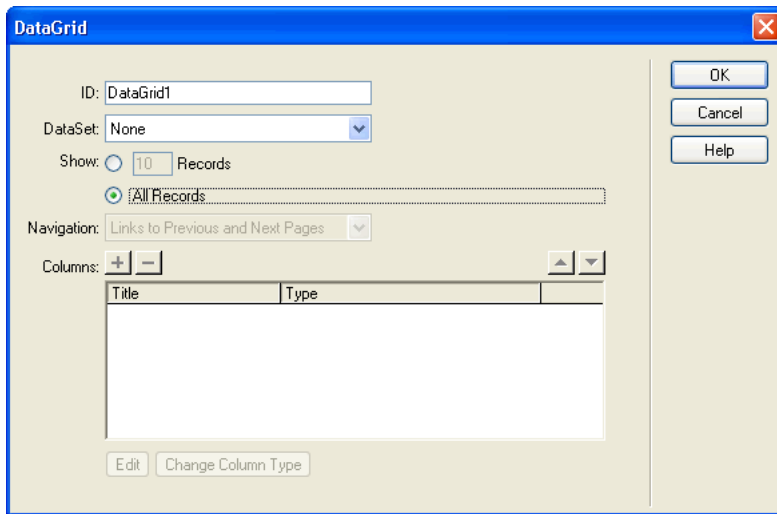
- Microsoft MSDN at: <http://msdn.microsoft.com>
- Microsoft ASP.NET at: <http://asp.net>

Before inserting the DataGrid server behavior, you must define a DataSet (referred to as a recordset by other document types) for the DataGrid. For more information, see “Defining a recordset” on page 502.

To add a DataGrid object to a page:

- 1 Open the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button and select DataGrid.

The DataGrid dialog box appears.



- 2 Complete the DataGrid dialog box and click OK.

For more information, click the Help button in the dialog box.

In the Document window, the DataGrid will display with a tabbed, gray outline surrounding it. In the Live Data window (View > Live Data), the gray outline disappears and the object’s placeholder is replaced with the specified DataGrid.

Adding a DataList to your page

The Dreamweaver DataList server behavior allows you to insert the ASP.NET DataList control into a page. The DataList control is used to implement a form-based front-end for data sources, and allows you to present data in a grid, where each record is rendered on a different row, and many rows can be displayed at once.

To learn more about the DataList control, and how it can be used to format dynamic data, visit the following websites:

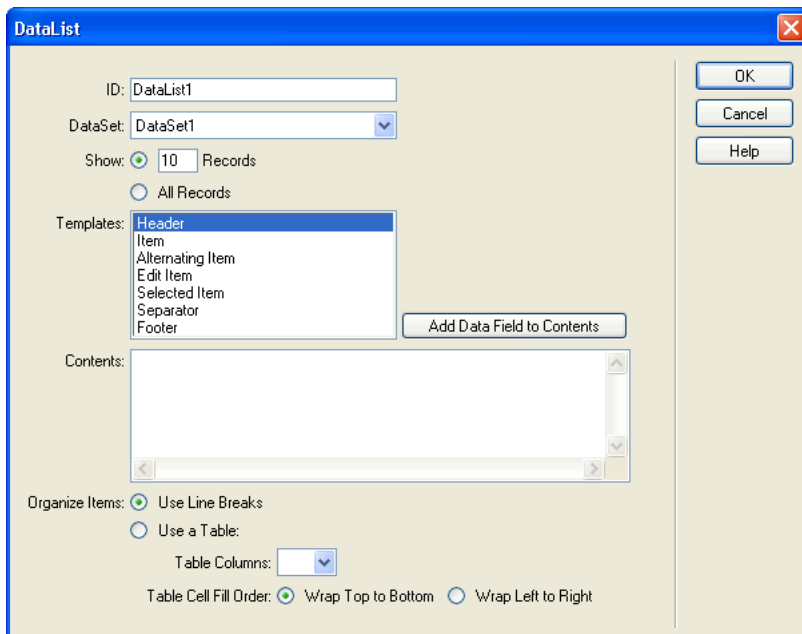
- Microsoft MSDN at: <http://msdn.microsoft.com>
- Microsoft ASP.NET at: <http://asp.net>

Before inserting the DataList server behavior, you must define a DataSet (referred to as a recordset by other document types) for the DataList. For more information, see “Defining a recordset” on page 502.

To add a DataList object to your page:

- 1 Open the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button, and select DataList.

The DataList dialog box appears.



- 2 Complete the DataList dialog box and click OK.

For more information, click the Help button in the dialog box.

In the Document window, the DataList object will display with a tabbed, gray outline surrounding it. In the Live Data window (View > Live Data), the gray outline disappears and the object's placeholder is replaced with the specified DataList.

Modifying a DataGrid or DataList object

Four ways exist to modify DataGrid and DataList objects:

- In Design view, click the icon on the upper left corner of the DataGrid or DataList to switch to edit mode, then add or modify the content of any of the tabbed regions that appear
Note: You can also drag sources of dynamic content from the Bindings panel into a tabbed region
- In Design view, select the object and use the Property inspector to change its attributes
- Double-click the object in the Server Behaviors panel and change its properties in the dialog box that opens
- In Code view, select the DataGrid or DataList on the page, and use its tag dialog to change its attributes

CHAPTER 35

Using ColdFusion Components

ColdFusion components (CFCs) let you encapsulate application and business logic into self-contained, reusable units. CFCs also provide a fast, easy way to create web services.

You can use Macromedia Dreamweaver MX to create and modify CFCs. You can also use Dreamweaver to build web pages that use CFCs.

Note: You can use CFCs only with Macromedia ColdFusion MX. CFCs are not supported in ColdFusion 5.

This chapter contains the following sections:

- “Understanding ColdFusion components” on page 543
- “Visually creating a component in Dreamweaver” on page 544
- “Viewing ColdFusion components in Dreamweaver” on page 544
- “Editing ColdFusion components in Dreamweaver” on page 545
- “Building web pages that use ColdFusion components” on page 546

Understanding ColdFusion components

A ColdFusion component is a reusable software unit written in ColdFusion markup language (CFML). CFCs help make your code reusable and easy to maintain.

This section describes the advantages of CFCs. Later sections describe how Dreamweaver can help you work with CFCs. For information on CFC tags and syntax, see ColdFusion help in Dreamweaver (Help > Using ColdFusion).

ColdFusion components are meant to provide a simple yet powerful way for developers to encapsulate elements of their websites. Generally, you should use components for application or business logic. Use custom tags for presentation elements such as customized greetings, dynamic menus, and so on.

As with many other types of construction, dynamic sites can often benefit from interchangeable parts. For example, a dynamic site may run the same query repeatedly, or calculate the total price of shopping cart pages and recalculate it every time an item is added. These tasks can be handled by components. You can fix, improve, extend, and even replace a component with minimal impact to the rest of your application.

Suppose an online store calculates shipping charges based on the price of orders. For orders under \$20, the shipping charge is \$4; for orders between \$20 and \$40, the shipping charge is \$6, and so on. You could insert the logic for calculating the shipping charge in both the shopping cart page and the checkout page, but that would mix HTML presentation code and CFML logic code and generally make the code difficult to maintain and reuse.

You decide to create a ColdFusion component called Pricing that has, among other things, a function called ShippingCharge. The function takes a price as an argument and returns a shipping charge. For example, if the value of the argument is 32.80, then the function returns 6.

In both the shopping cart page and the checkout page, you insert a special tag to invoke the ShippingCharge function. When the page is requested, the function is invoked and a shipping charge is returned to the page.

Later, the store announces a special promotion: free shipping for all orders above \$100. You make the change to the shipping rates in one place—the ShippingCharge function of the Pricing component—and all the pages using the function automatically get accurate shipping charges.

Visually creating a component in Dreamweaver

You can use Dreamweaver to visually define a ColdFusion component and its functions. Dreamweaver creates a .cfc file and inserts the necessary CFML tags for you.

Note: Depending on the component, you may have to complete some code by hand.

To create a ColdFusion component visually:

- 1 Open a ColdFusion page in Dreamweaver.
- 2 In the Components panel (Window > Components), select CF Components from the pop-up menu.
- 3 On the Components panel, click the plus (+) button.
The Create Component dialog box opens.
- 4 Complete the dialog box and click OK.

For more information, click the Help button in the dialog box.

Dreamweaver writes a .cfc file and saves it in the folder you specified. The new component also appears in the Components panel (after clicking Refresh).

To remove a component, you must delete the .cfc file by hand from the server.

Viewing ColdFusion components in Dreamweaver

Dreamweaver provides a way to visually examine the ColdFusion components defined for your site. Dreamweaver reads the .cfc files located on the server and displays information about them in an easy-to-navigate tree view in the Components panel.

Note: Dreamweaver looks for the components on your testing server (see “Specifying where dynamic pages can be processed” on page 134). If you want to view components located on another server, change the testing server settings.

Specifically, you can perform any of the following actions in Dreamweaver:

- List all the ColdFusion components defined for the site
- Explore the functions and arguments of each component
- Inspect the properties of functions that serve as web services

To view the ColdFusion components in Dreamweaver:

- 1 Open any ColdFusion page in Dreamweaver.
- 2 In the Components panel (Window > Components), select CF Components from the pop-up menu.
- 3 Click the Refresh button on the panel to retrieve the components.

Dreamweaver displays the component packages on the server. A component package is a folder that contains CFC files.

If existing component packages do not appear, click the Refresh button on the panel toolbar.
- 4 Click the plus (+) icon beside the package name to view the components stored in the package.
- 5 To list the functions of a component, click the plus (+) icon beside the component name.
- 6 To see the arguments a function takes, as well as the arguments' type and whether they are required or optional, open the function's branch in the tree view.

Functions that take no arguments have no plus (+) icon beside them.
- 7 To quickly view the details of an argument, a function, a component, or a package, select the item in the tree view, then click the Get Details icon on the panel toolbar.

You can also right-click (Windows) or Control-click (Macintosh) the item and select Get Details from the pop-up menu.

Dreamweaver displays details about the item in a message box.

Editing ColdFusion components in Dreamweaver

Dreamweaver provides a streamlined way of editing the code of the ColdFusion components defined for your site. For example, you can add, change, or delete any component function without leaving Dreamweaver.

To use this feature, your development environment must be set up as follows:

- ColdFusion MX must be running locally.
- In the advanced Site Definition dialog box in Dreamweaver, the Access type specified in the Testing Server category must be Local/Network.
- In the advanced Site Definition dialog box, the path of your local root folder must be the same as the path of the testing server folder (for example, `c:\inetpub\wwwroot\cf_projects\myNewApp`). You can examine and change these paths by choose Site > Edit Sites.
- The component must be stored in the local site folder or any of its subfolders on your hard disk.

Before you can edit a CFC, open any ColdFusion page in Dreamweaver and display the components in the Components panel. To display the components, open the Components panel (Window > Components), select CF Components from the panel's pop-up menu, and click the Refresh button on the panel.

Because ColdFusion MX is running locally, Dreamweaver displays component packages on your hard disk. For more information, see "Viewing ColdFusion components in Dreamweaver" on page 544.

To edit a component:

- 1 To edit a component file generally, open the package and double-click the component name in the tree view.
Dreamweaver opens the component's file in Code view.
- 2 To edit a specific function, argument, or property, double-click the item in the tree view.
- 3 Make your changes by hand in Code view.
- 4 Save the file (File > Save).

To see any new function in the Components panel, refresh the view by clicking the Refresh button on the panel toolbar.

Building web pages that use ColdFusion components

One way to use a component function in your web pages is to write code in the page that invokes the function when the page is requested. You can use Dreamweaver to help you write this code.

Note: For other ways to use components, see ColdFusion help in Dreamweaver (Help > Using ColdFusion).

To use a ColdFusion component in a web page:

- 1 In Dreamweaver, open the ColdFusion page that will use the component function.
- 2 Switch to Code view (View > Code).
- 3 Open the Components panel (Window > Components), then select CF Components from the panel's pop-up menu.
- 4 Find the component you want and insert it using one of the following techniques:
 - Drag a function from the tree view to the page. Dreamweaver inserts code in the page to invoke the function.
 - Select the function in the panel and click the Insert icon on the panel toolbar (the second icon on the right). Dreamweaver inserts the code in the page at the insertion point.
- 5 If you insert a function that has arguments, complete the argument code by hand.
For more information, see ColdFusion help in Dreamweaver (Help > Using ColdFusion).
- 6 Save the page (File > Save).

CHAPTER 36

Using Web Services

Web services are an emerging technology that allow web pages to access distributed applications. By offering both access to information and application functionality as a service, web services can be delivered and paid for as streams of services that allow ubiquitous access from any platform. The web page that connects to the web service is commonly known as a **consumer**, and the service itself is known as a **publisher**. Macromedia Dreamweaver MX lets you create pages and sites that are consumers of web services. Dreamweaver currently supports the creation of web service consumers using ColdFusion, ASP.NET, and Java Server Pages (JSP) document types. Specifically, Dreamweaver allows you to perform the following web service development tasks:

- Select web services available on the Internet
- Generate a web service proxy that allows the web page to communicate with the web service publisher

The proxy (also known as an abstraction class) contains the fields, methods, and properties of the web service, and makes them available to the locally hosted page. When you generate a proxy for your page, Dreamweaver lets you view them in the Components panel.

- Drag methods and data types into the page's code

Before you create a web page that uses a web service, you must be familiar with the underlying server technology of the application you want to use and the programming constructs that the application requires.

Dreamweaver allows you to author web pages that can access web services and make use of the functionality the services provide. In addition, you can create and publish web services for deployment using Macromedia ColdFusion MX.

This chapter discusses the following topics:

- “Understanding web services” on page 548
- “Installing and configuring proxy generators” on page 550
- “Adding a web service proxy using the WSDL description” on page 552
- “Adding a web service to a page” on page 554
- “Editing the UDDI web service site list” on page 556

Understanding web services

Web services allow applications to communicate and share information across the Internet, regardless of operating system or programming language. Examples of web services, and the information and functionality they provide, include the following:

- User authentication and authorization
- Credit card validation
- Financial markets services that return stock prices associated with specified ticker symbols
- Purchasing services that allow users to order and buy products online
- Information services that provide news or other information types based on a selected interest, location, or other personal information

By providing functionality as a service that a web page connects to and uses as needed, web services give developers and service providers greater flexibility in designing and deploying powerful, distributed applications.

The parts of a web service

Web services consist of the following basic components:

- **Service publishers** provide hosted applications, and make them available for use. Web services can be provided either for free, or as a fee-based service.
- **Service brokers** maintain a registry of service providers with descriptions of service offerings and links to their applications.
- **Service consumers** are the web pages that access and use the remote web service.

Finding web service publishers

Web services themselves are made available by service publishers. Typically, the service publisher makes its web service available through a web-based registry that maintains a directory of available services that you can access. A number of websites provide such a directory, including:

- X Methods at <http://www.xmethods.net>
- IBM Business Registry at <http://www-3.ibm.com/services/uddi/protect/registry.html>
- Microsoft UDDI registry at <http://uddi.microsoft.com/default.aspx>

These registries use the Universal Description, Discovery and Integration (UDDI) service, an open, e-commerce service registry that provides a forum for businesses to describe themselves and the goods or services they can provide to other businesses. A group of companies, called **operators**, maintain the registry. The operators have pledged to share all public information about registrants among themselves and with users of the service, and to maintain inter-operability among the multiple peer nodes of the UDDI service network. In addition to public web services, there are also private UDDI registries available on a subscription basis.

The UDDI specification is based on existing Internet standards, ensuring that it is platform and implementation neutral.

Web service software components

In order for a web page to access and use a web service, it must communicate with the service and have a description of the functionality the service provides, the available methods that it can invoke, and the parameters the service returns. The Web Service Description Language (WSDL) is an XML-based description of the service. Each web service provides a WSDL that describes how to bind to the service, the available methods that the web page can invoke, and the data inputs and outputs. The WSDL can reside in either a file, or it can be generated by the web service at runtime.

Communication between the web page requesting the service, and the web service itself uses the Simple Object Access Protocol (SOAP). SOAP is an XML-based protocol that lets a web client access and invoke the web service's methods and parameters.

Web services information and specifications

To learn more about web services, and the underlying technologies that make them possible, visit the following websites:

- WSDL specification at <http://www.w3.org/TR/wsdl>
- UDDI specification at <http://www.uddi.org/specification.html>
- XML specification at <http://www.w3.org/TR/REC-xml>
- SOAP specification at <http://www.w3.org/TR/SOAP/>

The Dreamweaver web service workflow

To create a page or site that consumes a web service using Dreamweaver, you must perform the following tasks:

- 1 Install and configure a proxy generator.

Proxy generators generate a web service proxy, a software component that a web page uses to communicate with the web service publisher. The web service proxy is generated from the WSDL that describes the web service. Depending on the server technology you want to develop web service consumers for, you may need to install and configure a proxy generator that supports that technology.

Dreamweaver comes pre-configured with AXIS, the Apache SOAP proxy generator that supports JSP web service development. If you are developing ColdFusion 6 pages, the web service proxy generator is included in the ColdFusion server. Establishing a connection to the ColdFusion server gives you access to the proxy generator.

If you are developing web service pages for use with ASP.NET, you must install the ASP.NET SDK, which is available from Microsoft.

For information on installing and configuring a proxy generator not supplied with Dreamweaver, see “Installing and configuring proxy generators” on page 550.

- 2 Using a browser, view a web-based registry of web services.

There are several sources of web services, ranging from web service registry sites to simple lists. The registries use UDDI, a standard that lets service providers and requestors find and transact with one another. UDDI allows businesses to locate services on the web that meet their needs. For example, using UDDI you can specify certain criteria such as the lowest price for a certain service, or that specific information be returned.

- 3 After locating and selecting a web service that provides the functionality you need, enter the URL of the WSDL in the Adding a Web Service dialog box.
- 4 Generate a proxy for the web service from the WSDL description of the service publisher.

To embed a web service into a web page, you must create a proxy. The proxy provides the web page with the necessary information to communicate with the web service, and access the methods the web service provides.

To create a proxy from the WSDL file, use a proxy generator. After you create the proxy, you can install it either:

 - On the local computer where you are developing the web service consumer.
 - On the server computer running the application server. To deploy the web page and have it communicate with the web service publisher you must install the proxy on the server.
- 5 Using Dreamweaver, add the web service to a page and edit the necessary parameters and methods to make use of the service's functionality.

For more information, see "Adding a web service to a page" on page 554.

Installing and configuring proxy generators

Dreamweaver installs the AXIS proxy generator, which supports JSP web services. AXIS is an open source proxy generator distributed through the Apache SOAP project. In addition, you can add proxy generators that support other vendor's web service implementations, or new web service technologies. This section describes how to obtain proxy generators and configure them to work with Dreamweaver.

To learn more about AXIS, see the Apache AXIS website at: <http://xml.apache.org/axis/index.html>.

Obtaining additional proxy generators

If you want to install a proxy generator that is not supplied with Dreamweaver, you must obtain the proxy generator and any related software components from the vendor. You should be able to download all the necessary files from the vendor's website.

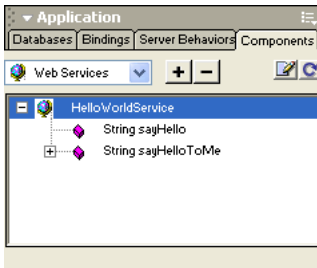
Some proxy generators create proxies that depend on other software libraries, which must be appropriately installed so the proxy generator can access them. For example, the AXIS proxy generator creates proxies that depend on the Apache SOAP library, which, in turn, depends on other software libraries (note that all the necessary software needed to use AXIS is installed by default with Dreamweaver). When selecting a proxy generator, consult the provided documentation, and ensure that you have all the required software components and libraries, so that you can properly install and configure them.

After you have properly installed and configured the proxy generator, you must configure it to work with Dreamweaver.

Note: Currently, web services development with Dreamweaver is limited to the Windows environment. To develop pages that access web services on the Macintosh, you must use a separate application server running either Windows NT/2000/XP or UNIX on which to run the web service proxy and its application environment.

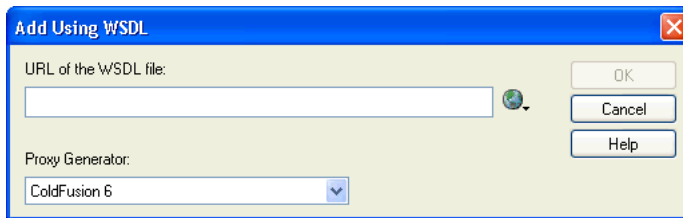
To configure a proxy generator to work with Dreamweaver:

- 1 Choose Window > Server Components to open the Server Components panel.

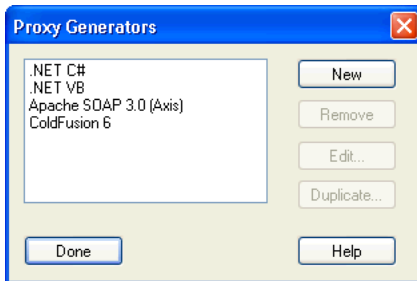


- 2 In the Server Components panel, choose Web Services from the pop-up menu in the upper left of the panel, then click the plus (+) button and select Add Using WSDL.

The Add Using WSDL dialog box appears.



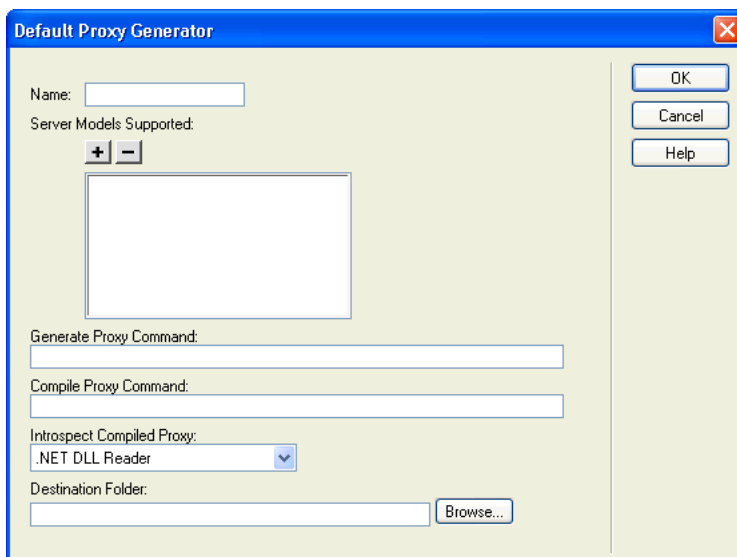
- 3 In the Add Using WSDL dialog box, choose Edit Proxy Generator List from the Proxy Generator pop-up menu. The Proxy Generator dialog box appears.



- 4 Click New, select the proxy generator from the pop-up menu, and click Done.

If the proxy generator you want to use not appear in the list, choose Default Proxy Generator to display the Default Proxy Generator dialog box.

- 5 The Default Proxy Generator dialog box lets you configure the selected proxy generator, or you can configure a new proxy generator. The dialog box's text boxes vary depending on the proxy generator you chose. Complete the dialog box and click OK.



For information on completing the dialog text box, click the Help button in the dialog box.

- 6 When you have completed setting the Default Proxy Generator settings, click OK.
When Dreamweaver reads a WSDL description of a web service, Dreamweaver carries out the following actions relating to the fields of the Default Proxy Generator:
 - Reads the WSDL as input to find the web service.
 - Generates the web service web service proxy with the specified runtime environment.
 - Compiles the proxy with the specified compiler.
 - Outputs the proxy source code and the compiled proxy in the specified destination folder.

Adding a web service proxy using the WSDL description

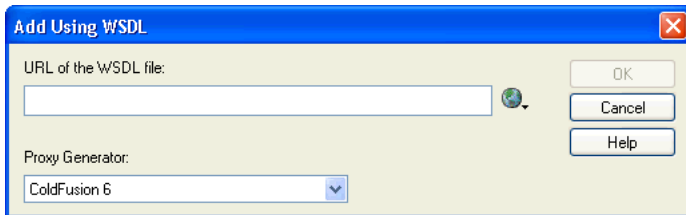
After you have specified a proxy generator and configured the web service server models you want to support, you need to find a web service that provides the desired functionality and generate a proxy for that service.

To select a web service and generate a proxy from its WSDL file:

- 1 Open the page that you want to add the web service to.
- 2 Choose Window > Server Components to open the Server Components panel.

- 3 In the Server Components panel, choose Web Services from the pop-up menu in the upper left of the panel, click the plus (+) button, and select Add Using WSDL.

The Add Using WSDL dialog box appears.



- 4 Specify the URL of the WSDL file you want to use.

If you know the URL of the WSDL file, enter it in the URL of the WSDL text box.

If you don't know the URL of the WSDL file, you can browse a directory of web services. When you find the web service you want, copy and paste the URL of the web service into the WSDL edit box. To launch a web browser, click the UDDI browse button and select one of the listed web service registries. Dreamweaver will launch the browser and open the selected registry. Locate the web service you want to use, and copy the URL of its WSDL file to the Clipboard (Control+C in Windows or Command+C on the Macintosh). Return to the Web Services Chooser and paste the URL into the dialog box.

You can edit the list of web service registries to include additional web service directories or specific web service providers. For more information, see “Editing the UDDI web service site list” on page 556.

- 5 Select a proxy generator that supports your desired web services server model from the Proxy Generator pop-up menu.

Make sure the proxy generator is installed and configured on your system. For more information, see “Installing and configuring proxy generators” on page 550.

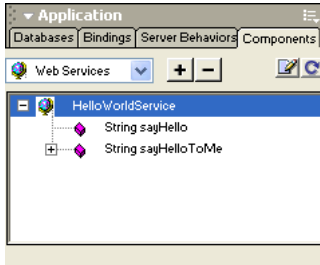
- 6 Click OK.

The proxy generator creates a proxy for the web service and introspects it. Introspection is the process where the proxy generator queries the internal structure of the web service proxy, and makes its interfaces, methods, and properties available through Dreamweaver.

The web service is now available for use in the site, and appears in the Server Components panel.

Adding a web service to a page

After selecting a web service, generating its proxy, and adding it to the Server Components panel, you can insert it into a page. The illustration below shows the Server Components panel with the web service proxy `HelloWorld` added. The `HelloWorld` proxy provides one method, `sayHello`, which prints “Hello World!”.



The following examples instantiate the `HelloWorld` web service using ColdFusion. To learn more about creating a web services, and to see additional examples using .NET and JSP, visit the Macromedia Support Center at: http://www.macromedia.com/go/creating_web_services.

To add a web service to a page:

- 1 In the Document window, in Code view, drag the `sayHello` method into the page’s HTML. Dreamweaver adds the method and dummy parameters to the page.
- 2 Edit the inserted code with appropriate service instance names, data types, and parameter values, as required by the web service. The web service should provide descriptions of the data types and parameter values.

In the ColdFusion example shown below, the web service is enclosed by the `<cfinvoke>` tags. When developing a web service in ColdFusion, use `<cfinvoke>` to instantiate the web service and invoke its methods.

```
<html>
<head>
<title>Web Service</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>

<cfinvoke
webservice="http://velcro-qa-7:8100/helloworld/HelloWorld.cfc?wsdl"
method="sayHello"
returnvariable="aString">
</cfinvoke>

</body>
</html>
```

- 3 If you want to bind a return value to a visual element, switch to Design view and place a visual element on the page that can accept data binding. Then switch back to Code view and enter the appropriate code to bind the returned value to the visual element. When creating web services, refer to the technology provider's documentation for the proper syntax with which to both instantiate the service and display the returned values to the page.

In this example, the value returned for the variable `aString` is output using the ColdFusion `<cfoutput>` tag. This will display the sentence "The web service says: HelloWorld!" to the page.

```
<html>
<head>
<title>Web Service</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<cfinvoke
webservice="http://velcro-qa-7:8100/helloworld/HelloWorld.cfc?wsdl"
method="sayHello"
returnvariable="aString">
</cfinvoke>
```

```
The web service says: <cfoutput>#aString#</cfoutput>
</body>
</html>
```

- 4 When you deploy web pages to a production server, Dreamweaver automatically copies the pages, the proxy, and any necessary libraries to the web server.

Note: If you develop the application with a proxy that is installed on a separate computer from the one where you developed the pages, or if you use a site management tool that does not copy all of the related files to the server, you must make certain to deploy both the proxy and any dependant library files. Otherwise, your pages cannot communicate with the web service application.

Editing the UDDI web service site list

The Web Service Chooser provides a list of UDDI-based web service directories from which you can select web services. You can edit this list to add or delete web service directories.

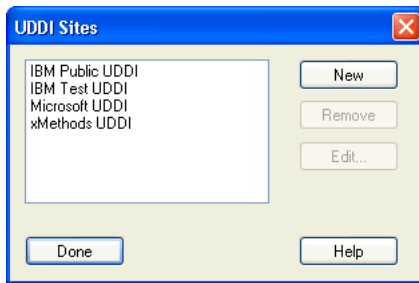
To edit the web service site list:

- 1 In Dreamweaver, select Window > Components to open the Components panel.
- 2 In the Components panel, choose Web Services from the pop-up menu in the upper left of the panel, then click the plus (+) button to add a web service.

The Add Using WSDL dialog box appears.

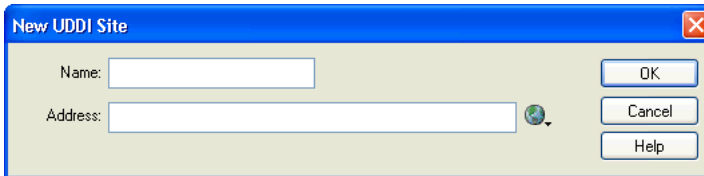
- 3 In the Web Services Chooser, click the globe icon and choose Edit UDDI Site List from the pop-up menu.

The UDDI Sites dialog box is displayed.



- 4 From the UDDI Sites dialog box you can add new web service sites, edit the name and URL of existing sites, and remove unwanted sites.

To remove an existing site, select it in the list and click the Remove button. To add a new site, or modify an existing one, click the New or Edit button and complete the dialog box shown below.



CHAPTER 37

Adding Custom Server Behaviors

Macromedia Dreamweaver MX comes with a set of built-in server behaviors that let you easily add dynamic capabilities to a site. If you want to extend Dreamweaver's functionality, you can create new server behaviors to suit your development needs, or obtain server behaviors from the Macromedia Exchange website.

This chapter contains the following sections:

- “Installing additional server behaviors” on page 557
- “Creating server behaviors” on page 558
- “Editing and modifying server behavior code” on page 571

Installing additional server behaviors

Many Macromedia partners and independent developers create custom server behaviors to address specific web development needs. You can access and download these custom server behaviors from the Macromedia Exchange for Dreamweaver website.

The server behaviors and other extensions available through the Macromedia Exchange website allow you to easily add new features to Dreamweaver. Each server behavior includes a short description, user reviews, and a discussion group where you can post questions and get support for the server behaviors you download.

To access Macromedia Exchange:

- 1 In Dreamweaver MX select Help > Dreamweaver Exchange.

Your browser will open the Macromedia Exchange for Dreamweaver web page.

- 2 Log on to the Exchange using your Macromedia ID, or, if you have not yet created a Macromedia Exchange ID for yourself, follow the instructions to open a Macromedia account.

Note: You can also access the Macromedia Exchange from the Server Behaviors panel (Window > Behaviors) by clicking the plus (+) button and choosing Get More Server Behaviors.

To install a server behavior or other extension in Dreamweaver:

- 1 Launch the Extension Manager by selecting Help > Manage Extensions.
- 2 Select File > Install Package in the Extension Manager.

For more information, see the Extension Manager online help.

Creating server behaviors

If you are a developer proficient in ColdFusion, ASP.NET, JavaScript, VBScript, PHP, or Java, you can write your own server behaviors. The steps to create a server behavior include the following tasks:

- Writing one or more code blocks that perform the required action.

For information on creating server behaviors with the Dreamweaver Server Behavior Builder see “Using the Server Behavior Builder” on page 558. For information on the syntax supported by Dreamweaver server behaviors, see “Writing code blocks” on page 561.

- Specifying where the code block should be inserted within the page’s HTML source.

For information on positioning code blocks within a page see “Positioning code blocks” on page 567.

- If the server behavior requires that a value be specified for a parameter, creating a dialog box that prompts the web developer applying the behavior to supply an appropriate value.

For information on supplying parameter values to a server behavior using a dialog box see “Creating a dialog box for a custom server behavior” on page 568.

- Testing the server behavior before making it available to others.

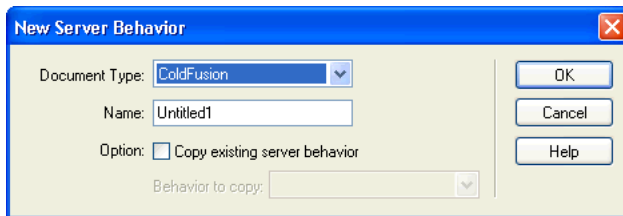
For guidelines on testing server behaviors see “Testing server behaviors” on page 570.

Using the Server Behavior Builder

Use the Server Behavior Builder to add the code block or blocks that the behavior inserts into the page.

To write server behavior code blocks:

- 1 In the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button and choose New Server Behavior from the pop-up menu.
- 2 The New Server Behavior dialog box appears.

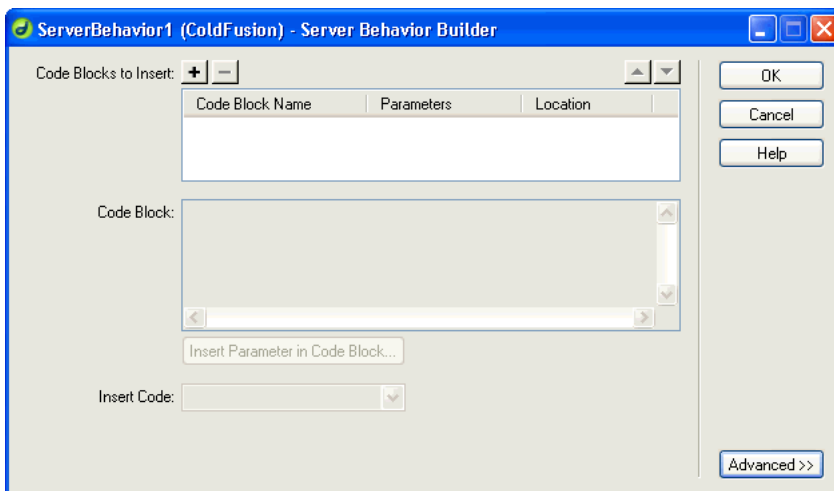


- 3 From the Document Type pop-up menu, select the document type that you are developing the server behavior for.
- 4 In the Name text box, enter a name for the server behavior.
- 5 If you want to copy an existing server behavior to add to the behavior you are creating, select the Copy Existing Server Behavior check box.

When this check box is selected, a list of available server behaviors is displayed in the Behavior to Copy pop-up menu.

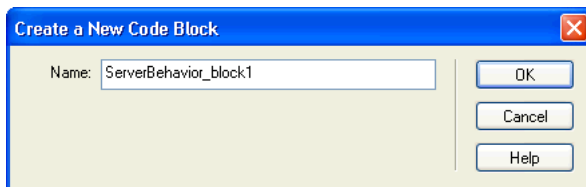
6 Click OK.

The Server Behavior Builder dialog box is displayed.



7 To add a new code block, click the plus (+) button.

The Create a New Code Block dialog box is displayed.



8 Enter a name for the code block you want to create.

9 In the Code Block text box, enter the code necessary to implement the server behavior. When you enter code in this text box, you can insert only a single tag or code block. If you need to enter multiple tags or code blocks, you must create that many individual code blocks in the Server Behavior Builder dialog box.

10 If the server behavior requires runtime parameters, you can include them in the code by clicking the Insert Parameters in Code Block button. The Insert Parameters in Code Block dialog box is displayed.

11 In the parameter Name pop-up menu, enter a name for the parameters, and click OK.

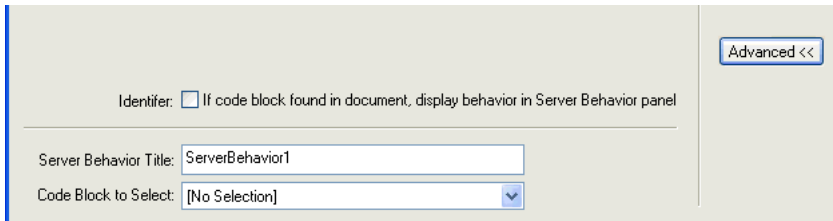
The parameter is inserted into the code block at the insertion point created by the cursor prior to defining the parameter.

12 Select an option from the Insert Code pop-up menu specifying the location in which to embed the code blocks.

For more information see “Positioning code blocks” on page 567.

- 13 You can specify additional information about the server you are creating using the Advanced options panel.

Click the Advanced button to display the advanced options.



Identifier specifies whether the code block should be treated as an identifier.

By default, every code block is an identifier. If Dreamweaver finds an identifier code block anywhere in a document, it lists the behavior in the Server Behaviors panel. Use the Identifier check box to specify whether the code block should be treated as an identifier.

At least one of the server behavior's code blocks must be an identifier. A code block should *not* be an identifier if the same code block is used in another server behavior, or if the code block provides a commonly used piece of functionality that occurs naturally within a page.

Server Behavior Title specifies the title of the behavior as it appears in the Server Behaviors panel.

When a page designer clicks the plus (+) button of the Server Behaviors panel, the new server behavior's title will appear in the pop-up menu. When a designer applies an instance of a server behavior to a document, the behavior appears in the list of applied behaviors in the Server Behaviors panel. Use the Server Behavior Title box to specify the contents of the plus (+) pop-up menu and the list of applied behaviors.

The initial value in the box is the name you supplied in the New Server Behavior dialog box. As parameters are defined, the name is automatically updated so that the parameters appear inside parentheses after the server behavior name.

Set Session Variable (@@Name@@, @@Value@@)

If the user accepts the default value, everything before the parentheses will appear in the plus (+) pop-up menu (for example, Set Session Variable). The name plus the parameters will appear in the list of applied behaviors—for example, Set Session Variable ("abcd", "5").

Code Block to Select specifies what code block is selected when the user selects the behavior in the Server Behaviors panel.

When you apply a server behavior, one of the code blocks within the behavior is designated the “code block to select.” If you apply the server behavior and then select the behavior in the Server Behaviors panel, in the Document window Dreamweaver automatically selects the designated block. By default, Dreamweaver selects the first code block that is not above the `html` tag. If all the code blocks are above the `html` tag, then Dreamweaver selects the first one. Advanced users can specify which code block is the selected one.

- 14 If you need to create more code blocks, repeat steps 7 through 13 as needed.
- 15 If the server behavior requires that parameters be supplied to it, you will need to create a dialog box that accepts parameters from the person applying the behavior.

To create a dialog box that accepts user input parameters, see “Creating a dialog box for a custom server behavior” on page 568.

- 16 After you have performed the above steps as required by the server behavior you are creating, Click OK.

Once you create a server behavior, it is listed in the Server Behaviors panel. Test the server behavior and ensure that it functions properly.

For additional information, see the following sections:

- “Writing code blocks” on page 561
- “Using parameters in server behaviors” on page 562
- “Making code blocks conditional” on page 563
- “Repeating code blocks” on page 564
- “Coding guidelines” on page 566
- “Positioning code blocks” on page 567

Writing code blocks

The code blocks you create in the Server Behavior Builder are encapsulated in a server behavior, that appears in the Server Behaviors panel. The code can be any valid runtime code for the specified server model. For example, if you choose ColdFusion as the document type for your custom server behavior, then the code you write must be valid ColdFusion code that runs on a ColdFusion application server.

Entering code blocks

You can create the code blocks either directly within the Server Behavior Builder, or you can copy and paste the code from other sources. Each code block you create in the Server Behavior Builder must be a single tag or script block. If you need to insert multiple tag blocks, split them into separate code blocks.

Including Parameters

You can include parameters in your runtime code and let the page designer supply the parameter values. To do so, enter parameter markers in the code, as follows:

```
@@parameterName@@
```

For more information, see “Using parameters in server behaviors” on page 562.

Applying conditions and repeating elements to code blocks

If you want the code block, or a portion of a code block, to be executed only if a certain condition or conditions apply, use the following syntax:

```
<@ if (expression1) @>
  code block1
[<@ elseif (expression2) @>
  code block2]*
[<@ else @>
  code block3]
<@ endif @>
```

The square brackets ([]) denote optional code and the asterisk (*) denotes zero or more instances. The condition expression is any valid JavaScript condition expression, and may contain server behavior parameters. For more information, see “Making code blocks conditional” on page 563.

If you want the code block, or a portion of a code block, to be repeated a number of times, use the following syntax:

```
<@ loop (@@param1@@,@@param2@@) @>
  code block
<@ endloop @>
```

The loop directive takes a comma-separated list of parameter arrays as arguments. The repeating text will be duplicated *n* times, where *n* is the length of the parameter array arguments. If more than one parameter array argument is specified, all the arrays must have the same length. On the *i*th evaluation of the loop, the *i*th elements of the parameter arrays replace the associated parameter instances in the code block. For more information, see “Repeating code blocks” on page 564.

For general information on coding, see “Coding guidelines” on page 566.

Using parameters in server behaviors

You can include parameters in a server behavior’s code, and let the page designer supply the necessary parameter values before inserting the server behavior’s code into the page. To let the page designer supply parameter values, enter parameter markers within the code as shown:

```
@@parameterName@@
```

The example ASP server behavior below contains the parameter `formParam`, which requires the person inserting the behavior to supply a the name of a form object:

```
<% Session(“lang_pref”) = Request.Form(“formParam”); %>
```

To create a parameter that lets the user supply the necessary value:

1 Enclose the `formParam` string in parameter markers:

```
<% Session(“lang_pref”) = Request.Form(“@@formParam@@”); %>
```

2 Create a dialog box that prompts the designer to supply the name of the form object. For more information, see “Creating a dialog box for a custom server behavior” on page 568.

Making code blocks conditional

Dreamweaver allows you to develop code blocks that incorporate control statements that execute conditionally. The Server Behavior Builder uses `if`, `elseif`, and `else` statements, and may contain server behavior parameters. This allows you to insert alternate text blocks based on the values of OR relationships among server behavior parameters. The `if`, `elseif`, and `else` statements appear as shown below. Note that square brackets (`[]`) denote optional code, and the asterisk (*) denotes zero or more instances:

```
<@ if (expression1) @>
    conditional text1
[<@ elseif (expression2) @>
    conditional text2]*
[<@ else @>
    conditional text3]
<@ endif @>
```

Condition expressions can be any JavaScript expression that can be evaluated using the JavaScript `eval()` function, and may include a server behavior parameter marked by `@@`'s. (The `@@`'s are necessary to distinguish the parameter from JavaScript variables and keywords.)

You can nest any number of conditionals or a loop directive (see “Repeating code blocks” on page 564) within a conditional directive. For example, you can specify that if an expression is true to execute a loop.

Note: New lines after each “`@>`” are ignored.

Effectively using conditional expressions

When using `if`, `else`, and `elseif` directives within the `insertText` XML tag, the participant text is preprocessed to resolve the `if` directives and to determine which text to include in the result. The `if` and `elseif` directives take the expression as an argument. The condition expression is the same as that for JavaScript condition expressions, and can also contain server behavior parameters. Directives such as this allow you to choose between alternative code blocks based on the values of, or relationships between, server behavior parameters.

For example, the JSP code shown below comes from a Dreamweaver MX server behavior that uses the conditional code block:

```
@@rsName@@.close();
<conditional_code>
@@rsName@@_hasData = @@rsName@@.next();
```

If the server behavior uses a normal recordset, the `<conditional_code>` placeholder is replaced with:

```
@@rsName@@ = Statement@@rsName@@.executeQuery();
```

If the server behavior uses a recordset from a callable object, it uses the following code instead.

```
@@callableName@@.execute();
@@rsName@@ = @@callableName@@.getResultSet();
```

If the server behavior is added for a callable object, the user would enter a value for the `@@callableName@@` parameter in the server behavior's Parameter dialog box. Otherwise, the `@@callableName@@` parameter would be empty. Thus, you can rewrite the previous insert text using `@@callableName@@` as the `if` argument. In this example, if the `@@callableName@@` parameter is supplied with a value, and first conditional code block (containing the `getResultSet()` method) is selected:

```
@@rsName@@.close();
<@ if ( @@callableName@@ != '' ) @>
@@callableName@@.execute();
@@rsName@@ = @@callableName@@.getResultSet();@ else @>
@@rsName@@ = Statement@@rsName@@.executeQuery();
<@ endif @>
@@rsName@@_hasData = @@rsName@@.next();
```

Repeating code blocks

When creating server behaviors, you can use looping constructs to repeat a code block a specified number of times. The loop syntax is:

```
<@ loop ( @@param1@@, @@param2@@, @@param3@@, @@param_n@@ ) @>
    code block
<@ endloop @>
```

The loop directive accepts a comma-separated list of parameter arrays as arguments. In this case, parameter array arguments allows a users to supply multiple values for a single parameter. The repeating text will be duplicated *n* times, where *n* is the length of the parameter array arguments. If more than one parameter array argument is specified, all the arrays must have the same length. On the *i*th evaluation of the loop, the *i*th elements of the parameter arrays replace the associated parameter instances in the code block.

When you later create a dialog box for the server behavior (see “Creating a dialog box for a custom server behavior” on page 568), you can add a control to the dialog box that allows the page designer to create parameter arrays. Dreamweaver includes a simple array control that you can use to create dialog boxes. This control, called Text Field Comma Separated List, is available through the Server Behavior builder. To create user interface elements of greater complexity, see the API documentation to create a dialog box with a control to create arrays (a grid control, for example).

Loop directives cannot be nested, but conditional directives (see “Making code blocks conditional” on page 563) can be nested within a loop directive.

The following example shows how such repeating code blocks can be used to create server behaviors (the example is a ColdFusion behavior used to access a stored procedure):

```
<CFSTOREDPROC procedure="AddNewBook"
    datasource=#MM_connection_DSN#
    username=#MM_connection_USERNAME#
    password=#MM_connection_PASSWORD#>
<CFPROCPARAM type="IN" dbvarname="@CategoryId" value="#Form.CategoryID#"
    cfsqltype="CF_SQL_INTEGER">

<CFPROCPARAM type="IN" dbvarname="@ISBN" value="#Form.ISBN#"
    cfsqltype="CF_SQL_VARCHAR">

</CFSTOREDPROC>
```

In this example, the CFSTOREDPROC tag can include zero or more CFPROCPARAM tags. However, without support for the loop directive, there is no way to include the CFPROCPARAM tags within the inserted CFSTOREDPROC tag. If this were to be created as a server behavior without the use of the loop directive, you would need to divide this example into two participants: a main CFSTOREDPROC tag, and a CFPROCPARAM tag whose participant type is multiple.

Using the loop directive, the same procedure can be written as follows:

```
<CFSTOREDPROC procedure="@@procedure@"
datasource=#MM_@@conn@@_DSN#
username=#MM_@@conn@@_USERNAME#
password=#MM_@@conn@@_PASSWORD#>
<@ loop (@@paramName@@,@@value@@,@@type@@) @>
  <CFPROCPARAM type="IN"
    dbvarname="@@paramName@"
    value="@@value@"
    cfsqltype="@@type@">
<@ endloop @>
</CFSTOREDPROC>
```

In the above example, and in the case of conditional code blocks as well, newlines after @> are ignored.

If the user entered the following parameter values in the server behavior dialog box:

```
procedure = "proc1"
conn = "connection1"
paramName = ["@CategoryId", "@Year", "@ISBN"]
value = ["#Form.CategoryId#", "#Form.Year#", "#Form.ISBN#"]
type = ["CF_SQL_INTEGER", "CF_SQL_INTEGER", "CF_SQL_VARCHAR"]
```

The server behavior would insert the following runtime code in the page:

```
<CFSTOREDPROC procedure="proc1"
datasource=#MM_connection1_DSN#
username=#MM_connection1_USERNAME#
password=#MM_connection1_PASSWORD#>
<CFPROCPARAM type="IN" dbvarname="@CategoryId" value="#Form.CategoryId#"
  cfsqltype="CF_SQL_INTEGER">
<CFPROCPARAM type="IN" dbvarname="@Year" value="#Form.Year#"
  cfsqltype="CF_SQL_INTEGER">
<CFPROCPARAM type="IN" dbvarname="@ISBN" value="#Form.ISBN#"
  cfsqltype="CF_SQL_VARCHAR">
</CFSTOREDPROC>
```

Note: Parameter arrays cannot be used outside of a loop except as part of a conditional directive expression.

Using the loop directive's `_length` and `_index` variables

The loop directive includes two built-in variables that you can use for embedded `if` conditions. The variables are: `_length` and `_index`. The `_length` variable evaluates to the length of the arrays processed by the loop directive, while the `_index` variable evaluates to the current index of the 'loop' directive. To ensure that the variables are only recognized as directives, and not as actual parameters to be passed into the loop, do not enclose either variable in `@`'s.

An example of using built-in variables is to apply them to the `import` attribute of the `page` directive. The `import` attribute requires comma separation of packages. If the `loop` directive extends around the entire `import` attribute, you will only want to output the attribute name `import=` on the first iteration of the loop (This would include the closing double quote (") , and not output a comma on the last iteration of the loop). Using the built-in variable, you can express this as:

```
<@loop (@@Import@@)@>
<@ if(_index == 0)@>import="
<@endif@>@@Import@@<@if (_index == _length-1)@>"<@else@>,
<@ endif @>
<@endloop@>
```

Coding guidelines

In general, your server behavior's code should be compact and robust. Web application developers are very sensitive to the code added to their pages. Follow generally accepted coding practices for the document type's language (ColdFusion, ASP.NET, JavaScript, VBScript, PHP, Visual Basic or Java). When writing comments, consider the different technical audiences that might need to understand the code, such as web and interaction designers, or other web application developers. Include comments that accurately describe the purpose of the code, and any special instructions for including it within a page.

Error checking

An important requirement is error checking. The server behavior's code should handle error cases gracefully. Try to foresee every possibility. For example, what if a parameter request fails? What if no records are returned from a query?

Use unique names

Your code should be clearly identifiable and avoid name collisions with existing code. For example, if the page contains a function called `hideLayer()` and a global variable called `ERROR_STRING`, and your server behavior inserts code that uses those names too, the server behavior may conflict with the existing code.

Create a prefix for your code's functions and global variables

Create your own prefix for runtime functions and global variables that you insert in a page. One convention is to use your initials. Never use the `MM_` prefix: it is reserved for Macromedia use only. Macromedia precedes all functions and global variables with the prefix `MM_` to prevent them from conflicting with your code.

```
var MM_ERROR_STRING = "...";
function MM_hideLayer() {
```

Avoid similar code blocks

Make sure the code in your blocks don't resemble too closely the code in other blocks. If a code block looks too much like another code block on the page, the Server Behaviors panel might mistakenly identify the first code block as an instance of the second code block (or conversely). A simple solution is to add a comment to a code block to make it more unique.

Positioning code blocks

After writing code blocks in the Server Behavior Builder, you must specify where to insert them in the page's HTML source code.

In the Insert Code pop-up menu, you can choose to insert the code block above the opening `<html>` tag, below the closing `</html>` tag, relative to another tag in the page, or relative to a tag selected by the page designer.

To position a code block above the `<html>` tag:

- 1 In the Insert Code pop-up menu, choose Above the `<html>` Tag.
- 2 Specify a location above the tag by choosing an option in the Relative Position pop-up menu.
You can insert the block at the beginning of the file, just before code blocks that open recordsets, just after code blocks that open recordsets, or just above the `<html>` tag. You can also specify a custom position.
- 3 If you want to specify a custom position, choose Custom Position from the Relative Position pop-up menu, then assign a weight to the code block.

Dreamweaver assigns a weight of 50 to all recordset-opening code blocks inserted above the `<html>` tag. If the weight of two or more blocks match, Dreamweaver randomly sets the order among the blocks.

Use the Custom Position option when you need to insert more than one code block in a particular order. For example, if you want to insert an ordered series of three code blocks after the code blocks that open recordsets, you could enter a weight of 60 for the first block, 65 for the second, and 70 for the third.

To position a code block below the closing `</html>` tag:

- 1 In the Insert Code pop-up menu, choose Below the `</html>` Tag.
- 2 Specify a location below the tag by choosing an option in the Relative Position pop-up menu.
You can insert the block just after the `</html>` tag, just before code blocks that close recordsets, just after code blocks that close recordsets, or just before the end of the file. You can also specify a custom position.
- 3 If you want to specify a custom position, choose Custom Position from the Relative Position pop-up menu, then assign a weight to the code block.

Dreamweaver assigns a weight of 50 to all recordset-closing code blocks inserted below the `</html>` tag. If the weight of two or more blocks match, Dreamweaver randomly sets the order among the blocks.

Use the Custom Position option when you need to insert more than one code block in a particular order. For example, if you want to insert an ordered series of three code blocks before the code blocks that close recordsets, you could enter a weight of 30 for the first block, 35 for the second, and 40 for the third.

To position a code block relative to another tag on the page:

- 1 In the Insert Code pop-up menu, choose Relative To a Specific Tag.
- 2 In the Tag text box, enter the tag or select one from the pop-up menu.
If you enter a tag, don't include the angled brackets (< >).
- 3 Specify a location relative to the tag by choosing an option in the Relative Position pop-up menu.

You can insert your code block just before or just after the opening or closing tags. You can also replace the tag with the code, insert the code as the value of an attribute of the tag (a box appears to let you choose the attribute), or insert the code inside the opening tag.

To position a code block relative to a tag selected by the page designer:

- 1 In the Insert Code pop-up menu, choose Relative To the Selection.
- 2 Specify a location relative to the selection by choosing an option in the Relative Position pop-up menu.

You can insert your code block just before or just after the selection. You can also replace the selection with your code block, or you can wrap the code block around the selection.

If you want to wrap the code block around a selection, the selection must consist of an opening and closing tag with nothing in between, as shown below:

```
<CFIF Day="Monday"></CFIF>
```

The opening tag piece of the code block is inserted before the selection's opening tag and the closing tag piece of the code block is inserted after the selection's closing tag.

Creating a dialog box for a custom server behavior

Server behaviors often require that the page designer supply a parameter value. This value must be inserted before the server behavior's code is inserted into the page. To do this, you can create a dialog box that prompts the person implementing the server behavior for a parameter value.

You create the dialog box by defining the designer-supplied parameters in the code. After defining all the parameters, you can generate a dialog box for the server behavior.

Note: A parameter is added to your code block without your intervention if you specify that your code should be inserted relative to a specific tag chosen by the page designer (that is, you chose Relative to a Specific Tag in the Insert Code pop-up menu). The parameter adds a tag menu to the behavior's dialog box to let the page designer choose a tag.

To create a parameter in the server behavior's code:

Enter a parameter marker at the point in the code where you want to insert the supplied parameter value. The syntax for the parameter is:

```
@@parameterName@@
```

For example, if the server behavior contains the following code block:

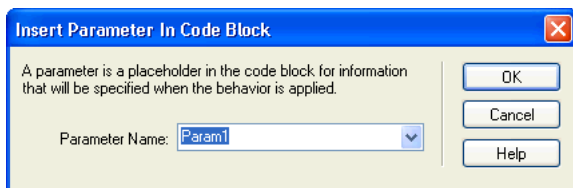
```
<% Session("lang_pref") = Request.Form("Form_Object_Name"); %>
```

To let the page designer supply the value of *Form_Object_Name*, enclose the string in parameter markers (@@):

```
<% Session("lang_pref") = Request.Form("@@Form_Object_Name@@"); %>
```


You can also highlight the string, then click the Insert Parameter In Code Block button. Enter a parameter name and click OK. Dreamweaver replaces every instance of the highlighted string with the specified parameter name enclosed in parameter markers.

Dreamweaver uses the strings you enclose in parameter markers to label the controls in the dialog box it generates (see procedure below). In the above example, Dreamweaver creates a dialog box with the following label:

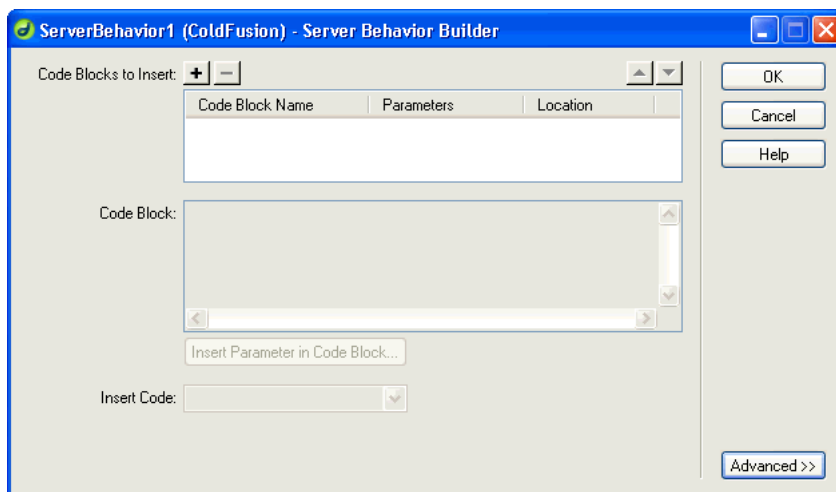


Note: Parameter names in the server behavior code cannot have any spaces. Therefore, the dialog box labels cannot have any spaces. If you want to include spaces in the label, you can edit the generated HTML file.

To create a dialog box for a server behavior:

- 1 In the Server Behavior Builder, click Next.

A dialog box appears listing all of the designer-supplied parameters you defined in your code.



- 2 If you want, change the display order of the dialog box controls by selecting a parameter and clicking the up and down arrows.
- 3 If you want, change a parameter's control by selecting the parameter and choosing another control in the Display As column.
- 4 Click OK.

Dreamweaver generates a dialog box with a labelled control for each designer-supplied parameter you defined. To view the dialog box, click the plus (+) button in the Server Behaviors panel (Window > Server Behaviors), and select your server behavior from the pop-up menu.

To edit the dialog box of a server behavior you created:

- 1 In the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button and choose Edit Server Behaviors from the pop-up menu.
- 2 Select your server behavior from the list, and click Open.
The Server Behavior Builder appears with your server behavior.
- 3 Click Next.
A dialog box appears listing all the designer-supplied parameters you defined in your code.
- 4 If you want, change the display order of the dialog box controls by selecting a parameter and clicking the up and down arrows.
- 5 If you want, change a parameter's control by selecting the parameter and choosing another control in the Display As column.
- 6 Click OK.

Testing server behaviors

The Macromedia Exchange recommends you perform the following tests on each server behavior you create:

- Apply the behavior from the Server Behaviors panel. If it has a dialog box, enter valid data in each field and click OK. Verify that no error occurs when the behavior is applied. Verify that the runtime code for the server behavior appears in the Code inspector.
- Apply the server behavior again and enter invalid data in each field of the dialog box. Try leaving the field blank, using large or negative numbers, using invalid characters (such as /, ?, :, *, and so on), and using letters in numeric fields. You can write form validation routines to handle invalid data (validation routines involve hand-coding, which is beyond the scope of this book).

After successfully applying your server behavior to the page, verify the following:

- Check the Server Behaviors panel to make sure the name of the server behavior appears in the list of behaviors added to the page.
- If applicable, verify that server-side script icons show up on the page. The generic server-side script icons are gold shields. To see the icons, turn on Invisible Elements (View > Visual Aids > Invisible Elements).
- In Code View, (View > Code) verify that no invalid code is generated.

In addition, if your server behavior inserts code in the document establishing a connection to a database, create a test database to test the code inserted in the document. Verify the connection by defining queries that produce different sets of data, and different sizes of data sets.

Finally, upload the page to the server and open it in a browser. View the page's HTML source code and verify that no invalid HTML has been generated by the server-side scripts.

Editing and modifying server behavior code

You can edit any server behavior created with the Server Behavior Builder, including server behaviors you download from the Macromedia Exchange website, and other third-party developers.

If you apply a server behavior to a page and then edit the behavior in Dreamweaver, instances of the old behavior will no longer appear in the Server Behaviors panel. The Server Behaviors panel searches the page for code that matches the code of known server behaviors. If the code of a server behavior known to the panel changes, the panel will no longer recognize earlier versions of the behavior on that page.

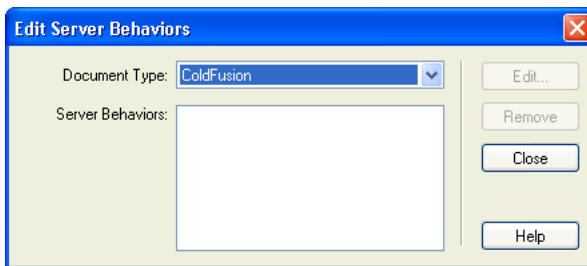
If you want both the old and new versions of the behavior to appear in the panel, click the plus (+) button on the Server Behaviors panel, choose New Server Behavior, and create a copy of the old server behavior.

Editing server behaviors

To edit the code of a server behavior created with the Server Behavior Builder:

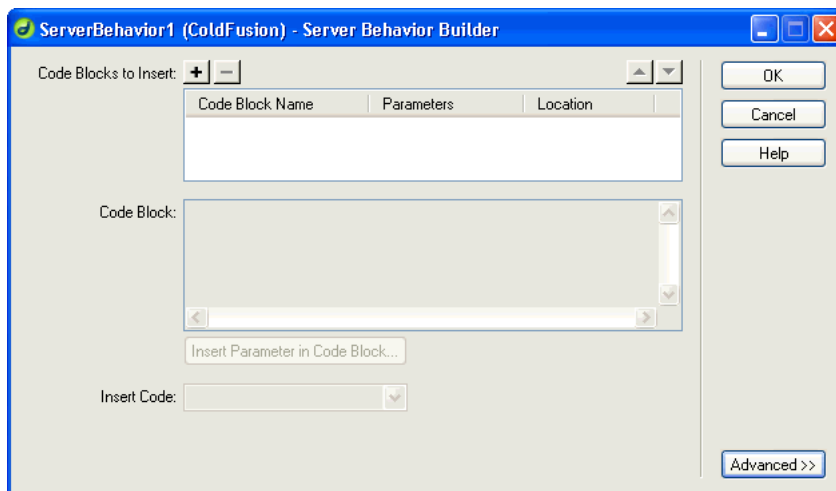
- 1 In the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button and choose Edit Server Behaviors from the pop-up menu.

The Edit Server Behaviors dialog box appears, displaying all the behaviors for the current server technology.



- 2 Select the server behavior and click Edit.

The Server Behavior Builder dialog box appears.



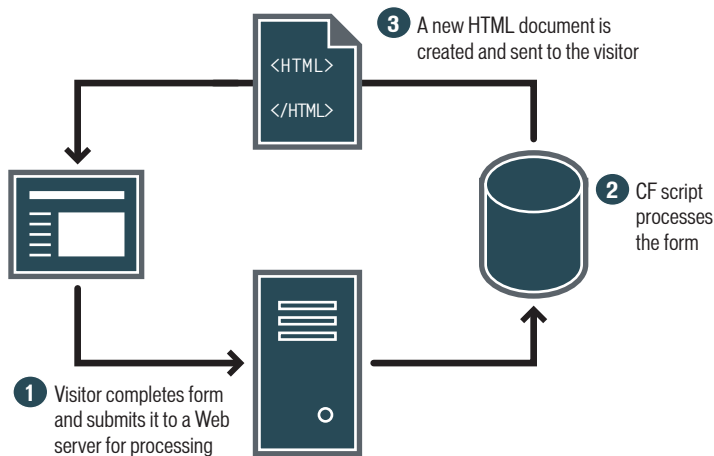
- 3 Select the appropriate code block and modify the code to be inserted in pages.
- 4 If you want, change or add parameter markers to the code.
For instructions, see “Creating a dialog box for a custom server behavior” on page 568.
- 5 If you want, change where the code block is inserted in the page’s HTML source code by choosing another option in the Insert Code pop-up menu.
For instructions, see “Positioning code blocks” on page 567.
- 6 If the modified code does not contain any designer-supplied parameters, click OK.
Dreamweaver regenerates the server behavior without a dialog box. The new server behavior appears in the plus (+) pop-up menu of the Server Behaviors panel.
- 7 If the modified code does contain designer-supplied parameters, click Next.
Dreamweaver asks you whether you want to create a new dialog box, overwriting the old one. Make your changes and click OK.
Dreamweaver saves all changes in the server behavior’s EDML file.

CHAPTER 38

Creating Interactive Forms

Forms allow you to interact with or gather information from visitors to a website. Forms collect information from the user and submit this information to the server for processing. Forms can contain various objects that enable user interaction. These form objects include text fields, list boxes, check boxes, and radio buttons. The `form` tag includes parameters that let you specify a path to the server-side script or application that will process the form data, and which HTTP method to use when transmitting data from the browser to the server.

When a visitor enters information into a website form and clicks the submit button, the information is sent to the server where a server-side script or application processes it. The server responds by sending requested information back to the user, or performing some action based on the forms contents. Typically, the information is processed by a Common Gateway Interface (CGI) script, ColdFusion page, JavaServer Page (JSP), or Active Server Page (ASP). Note that you can't collect form data without using a server-side script or application to process the data.



Macromedia Dreamweaver MX lets you create a variety of form objects, including text fields, password fields, radio buttons, checkboxes, pop-up menus, and clickable images (such as a Submit button).

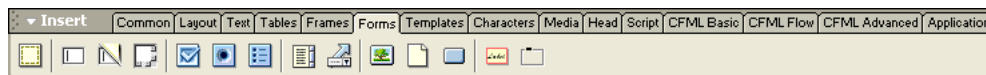
Dreamweaver also includes a Validate Form behavior to verify the information a visitor provides. For example, you can check that an e-mail address contains an “@” symbol, or that a required field contains an entry.

This chapter contains the following sections:

- About form objects
- Creating a form
- Understanding form objects
- Inserting checkboxes and radio buttons
- Adding lists and menus
- Adding form buttons
- About form design
- Using a client-side JavaScript function to process a form
- Using behaviors with forms
- Creating dynamic form objects

About form objects

In Dreamweaver, form input types are called **form objects**. You can insert form objects by choosing Insert > Form Objects, or by accessing the form objects from the Forms panel of the Insert bar shown below.



The Forms bar provides the following buttons:

Form inserts a form in the document. Dreamweaver inserts opening and closing `form` tags in the HTML source code. Any additional form objects, such as text fields, buttons, and so on must be inserted between the `form` tags for the data to be processed correctly by all browsers.

Text Field inserts a text field in a form. Text fields accept any type of alphanumeric entries. The entered text can be displayed as a single line, as multiple lines, or as bullets or asterisks (for password protection).

Hidden Field inserts a field in the document in which user data can be stored. Hidden fields let you store information entered by a user, such as a name, e-mail address, or purchase preference, and then use that data when the user next visits the site.

Check Box inserts a check box in a form. Check boxes allow multiple responses in a single group of options, a user can select as many options as apply.

Radio Button inserts a radio button in a form. Radio buttons represent exclusive choices. Selecting a button within a group deselects all others in the group. For example a user can select Yes or No.

Radio Group inserts a collection of radio buttons which share the same name.

List/Menu allows you to create user choices in a list. The List option displays the option values in a scrolling list and allows users to select multiple options in the list. The Menu option displays the option values in a pop-up menu and allows users to select only a single choice.

Jump Menu inserts a navigational list or pop-up menu. Jump menus let you insert a menu in which each option links to a document or file. See “Creating jump menus” on page 412.

Image Field allows you to insert an image in a form. Image fields can be used in place of Submit buttons to make graphical buttons.

File Field inserts a blank text field and a Browse button in a document. File fields let users browse to files on their hard disks and upload the files as form data.

Button inserts a text button within a form. Buttons perform tasks when clicked, such as submitting or resetting forms. You can add a custom name or label to a button, or use one of the predefined “Submit” or “Reset” labels.

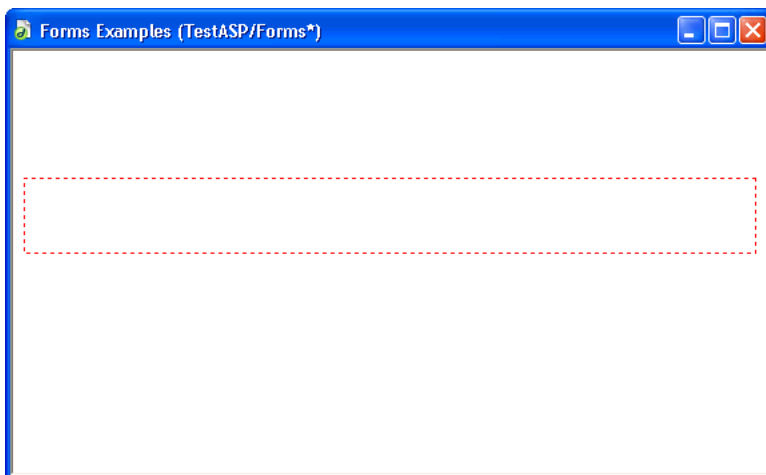
Creating a form

Dreamweaver’s Insert bar lets you create a form and add objects such as text fields, buttons, list boxes, and radio buttons to the form.

To add a form to a document:

- 1 Place the insertion point where you want the form to appear.
- 2 Choose Insert > Form or select the Forms category on the Insert bar and click the Form icon.

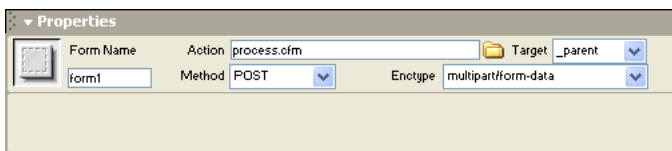
Dreamweaver inserts a form. With a page in Design view, forms are indicated by a dotted red outline. If you don’t see this outline, check that View > Visual Aids > Invisible Elements is selected.



- 3 In the Document window, click the form outline to select the form, or select the `<form>` tag in the tag selector, located in the bottom left corner of the Document window.

- 4 In the Property inspector's Form Name field, type a unique name to identify the form.

Naming a form makes it possible to reference or control the form with a scripting language, such as JavaScript or VBScript. If you do not name the form, Dreamweaver generates a name using the syntax `formn`, and increments the value of *n* for each form added to the page.



- 5 In the Property inspector's Action field, specify the path to the dynamic page or script that will process the form. You can either type the complete path into the Action field, or click the folder icon to navigate to the appropriate folder containing the script or application page.

If you're specifying a path to a dynamic page, the path to the URL will look similar to this example:

```
http://www.mysite.com/application_name/process.cfm
```

- 6 In the Method pop-up menu, choose the method that will transmit the form data to the server. The form Methods are:

POST	Embeds the form data in the HTTP request.
GET	Appends the value to the URL requesting the page.
Default	Uses the browser's default setting to send the form data to the server. Typically the default is the GET method.

Note that the method you choose may be dictated by the web or application server you are using. Contact your organization's server administrator for more information.

Do not use the GET method to send long forms. URLs are limited to 8,192 characters. If the amount of data sent is too large, data will be truncated, leading to unexpected or failed processing results. Also, do not use GET method when sending confidential user names and password, credit card numbers, or other confidential information. GET is not a secure method for passing information.

- 7 The Enctype pop-up menu lets you specify the MIME encoding type of the data submitted to the server for processing.

The default setting of `application/x-www-form-urlencoded` is typically used in conjunction with the POST method. If you are creating a file-upload field, specify the `multipart/form-data` MIME type.

- 8 The Target pop-up menu specifies the window in which to display the data returned by the invoked program.

If the named window is not already open, a new window with that name opens. The target values are:

_blank opens the destination document in a new unnamed window.

_parent opens the destination document in the parent window of the one displaying the current document.

_self opens the destination document in the same window as the one in which the form was submitted.

_top opens the destination document in the body of the current window. This value can be used to ensure that the destination document takes over the full window even if the original document was displayed in a frame.

Understanding form objects

Form objects are the mechanisms that allow users to input data. Before you create a form object, you must first insert a form into the page.

Tip: If you attempt to insert a form object without first creating the form, Dreamweaver will display the message “Add form tags?” Choose Yes to let Dreamweaver create form tags for the object.

The form objects you can add to a form are:

Text Fields accept any type of alphanumeric text entry. The text can be displayed as a single line, multiple lines, and as a series of dots or asterisks for private information (e.g. for password protection).

Hidden Fields store information entered by a user, such as a name, e-mail address, or viewing preference, and then use that data when the user next visits the site.

Buttons perform actions when clicked. Typically these actions include submitting or resetting a form. You can add a custom name or label for a button, or use one of the predefined “Submit” or “Reset” labels.

Check Boxes allow multiple responses within a single group of options. A user can select as many options as apply.

Radio Buttons represent exclusive choices. Selecting a button within a radio button group deselects all others in the group (a group consists of two or more buttons that share the same name). For example a user subscription form might require that you select “Male” or “Female.”

List/Menus display option values within a scrolling list that allows users to select multiple options. The Menu option displays the option values in a pop-up menu that allows users to select only a single item.

Jump Menus are navigational lists or pop-up menus, that let you insert a menu in which each option links to a document or file. See “Creating jump menus” on page 412.

File Fields let users browse to a file on their computer and upload the file as form data.

Image Fields let you insert an image in a form. Image fields can be used to make graphical buttons (e.g. Submit or Reset button).

Using form fields

Form fields let users enter text into a form. There are three types of form fields:

- Text fields (in which a user enters a response).
- File fields (in which a user enters the path of a file to upload to your server).
- Hidden fields (which store information a user entered in another field).

When you add a field to a form, you can specify the field's length, number of lines the field contains, maximum number of characters the user can enter, and whether the field is a password field.

Creating text fields

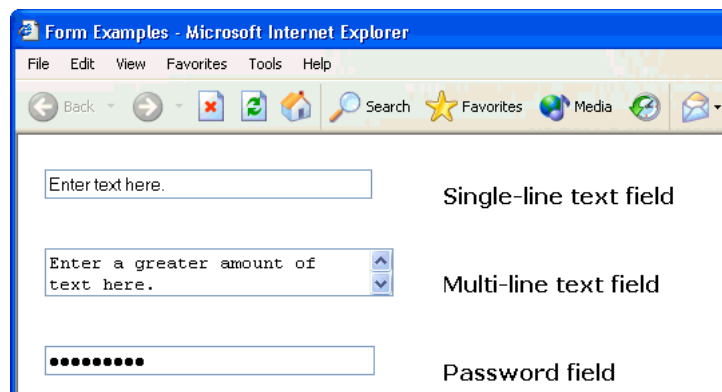
A text field is a form object in which users enter a response. There are three types of text fields:

Single-line text fields typically provide a single word or short phrase response, such as a name, or an address.

Multiple-line text fields provide the visitor with a larger area in which to enter a response. You can specify the maximum number of lines the visitor can enter, and the character width of the object. If text is entered that exceeds these settings, the field will scroll, following the setting specified in the wrap attribute.

Password fields are a special type of text field. When a user types in a password field, the entered text is replaced by asterisks or bullets to obscure the text, and protect the information from being viewed.

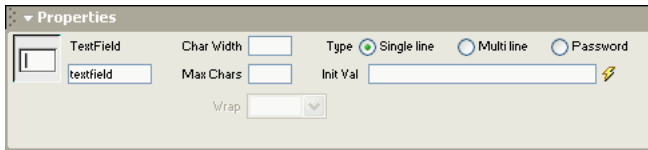
Note that passwords and other information sent to a server using a password field are not encrypted. The transferred data can be intercepted and read as alphanumeric text. For this reason, you should always provide encryption for data you want to keep secure.



To create a single-line or a password text field:

- 1 Place the insertion point inside the form outline.
- 2 Choose Insert > Form Objects > Text Field.

A text field appears in the document, and the TextField property inspector is displayed.



- 3 In the Property inspector's TextField field, enter a unique name for the text field.

The name you choose must uniquely identify the text field within the form. Form object names cannot contain spaces or special characters. You may use any combination of alphanumeric characters and an underscore (_). Note that the label you assign to the TextField is the variable name that will store the value (the entered data) of the field. This is the value sent to the server for processing.

- 4 In the Char Width field, do one of the following:

- Accept the default setting which sets the text field's length to 20 characters.
- Specify the maximum length of the text field. The maximum length of a text field is the maximum number of characters the field will display at a time. For example, if the Char Width is set to 20 (the default value) and a user enters 100 characters, only 20 of those characters will be viewable in the text field. Note that even though you cannot view the characters in the field, they are recognized by the field object and will be sent to the server for processing.

- 5 In the Max Chars field enter a value for the maximum number of characters the user can enter in the text field. This value defines the text field's size limits, and is used to validate the form.

If you leave the Max Chars field blank, users can enter any amount of text. If the text exceeds the character width of the field the text will scroll. If a user exceeds the maximum number of characters, the form produces an alert sound.

- 6 Select either Single-line or Password to specify the type of text field you want to create.

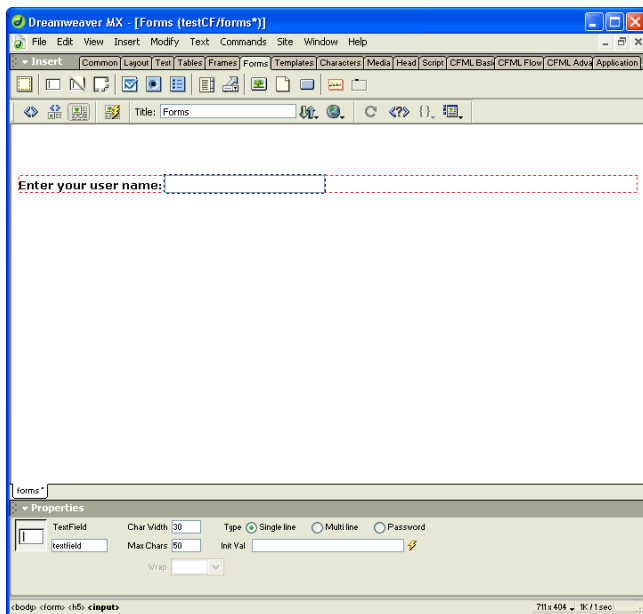
You can also choose to create a multiple-line text field. For instructions on creating multiple-line text fields, see "Creating multiple-line text fields" on page 580.

- 7 If you want to display a default text value in the field, enter the default text in the Property inspector's Init Value field.

This text will be displayed in the text field when the form first loads in the user's browser. For example, you might indicate that the user enter information in the field by including a note or example value.

- 8 To label the field within the page, place the insertion point next to the object and enter any text you want.

You can apply text formatting to form object labels. For more information, see “Setting and changing fonts and styles” on page 274. The example below shows a single-line text field with the label “Enter your user name:”. Note that the TextField Property inspector specifies that the field display up to 30 characters in the Char Width field, but that it can hold a maximum of 50 characters.



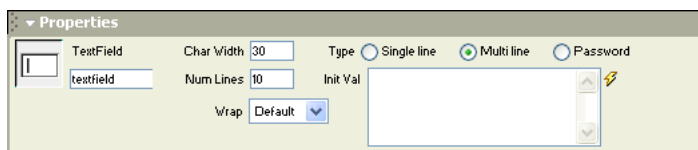
Creating multiple-line text fields

Multiple-line text fields let you enter multiple lines of text. When you create a multiple-line text field, you can specify the number of lines of text the user can enter.

To create a multiple-line text field:

- 1 Place the insertion point inside the form outline.
- 2 Choose Insert > Form Objects > Text Field.

A text field is inserted into the document, and the TextField property inspector is displayed.



- 3 In the Property inspector's TextField field, enter a unique name for the text field.

The name you choose must uniquely identify the text field within the form. Form object names cannot contain spaces or special characters. You may use any combination of alphanumeric characters and an underscore (_). Note that the label you assign to the TextField is the variable name that will store the value (the entered data) of the field. This is the value sent to the server for processing.

- 4 In the Char Width field, do one of the following:

- Accept the default setting which sets the text field's length to 20 characters.
- Specify the maximum length of the text field. The maximum length of a text field is the maximum number of characters the field will display at a time. For example, if the Char Width is set to 20 (the default value) and a user enters 100 characters, only 20 of those characters will be viewable in the text field. Note that even though you cannot view the characters in the field, they are recognized by the field object and will be sent to the server for processing.

- 5 Select Multi line to specify the type of text field you want to create.

- 6 In the Num Lines field specify the maximum number of lines to display.

The default setting sets a two-line text field.

- 7 From the Wrap pop-up menu, select a setting to set how the user's input will be displayed when the user enters more information than can be displayed within the defined text area. The wrap options are:

Off or Default prevents text from wrapping to the next line. When the user's input exceeds the right boundary of the text area, text will scroll toward the left. User's must press Return to move the insertion point to the next line in the text area.

Virtual sets word wrap in the text area. When the user's input exceeds the right boundary of the text area, text wraps to the next line. When data is submitted for processing, word wrap isn't applied to the data. It is submitted as one string of data.

Physical to set word wrap in the text area, as well as to the data when it is submitted for processing.

- 8 If you want to display a default text value in the field, enter the default text in the Property inspector's Init Value field.

This text will be displayed in the text field when the form first loads in the user's browser. For example, you might indicate that the user enter information in the field by including a note or example value.

- 9 To label the field within the page, place the insertion point next to the object and enter any text you want.

You can apply text formatting to form object labels. For more information, see "Setting and changing fonts and styles" on page 274.

Creating a file upload field

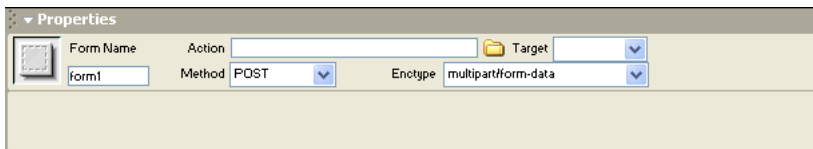
You can create a file upload field that lets users select a file on their computer—such as a word processing document or graphics file—and upload the file to the server. A file field looks like other text fields except it also contains a Browse button. The user either manually enters the path to the file they want to upload, or uses the Browse button to locate and select the file.

File fields require that you use the `POST` method to transmit files from the browser to the server. The file is posted to the address you specify in the form's Action field.

Note: Contact your server's administrator to confirm that anonymous file uploads are allowed before using the file field.

To create a file field in a form:

- 1 Insert a form in the page (Insert > Form).
- 2 Select the form. The form Property inspector is displayed.
- 3 Set the form Method to `POST`.
- 4 From the Enctype pop-up menu, select `multipart/form-data`.



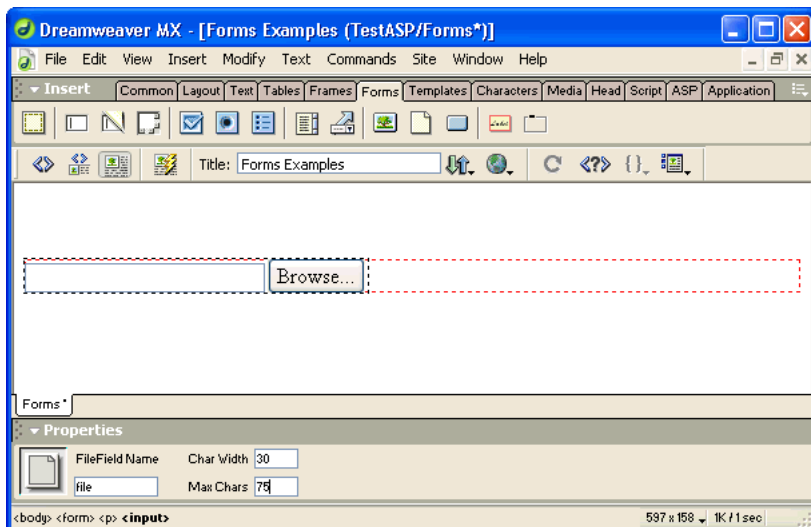
- 5 Place the insertion point inside the form outline, then choose Insert > Form Objects > File Field or click the Insert File Field icon in the Form tab of the Insert bar.

A file field is inserted within the form.

- 6 In the Property inspector's FileField Name field, enter a name for the file field object.
- 7 In the Char Width field, specify the maximum number of characters you want the field to display.
- 8 In the Max Char field, specify the maximum number of characters the field will hold.

If the user browses to locate the file, the filename and path can exceed the specified Max Chars value. However, if the user attempts to type in the filename and path, the file field will only allow the number of characters specified by the Max Chars value.

The example shown below depicts a simple file field. Note that the Property inspector's Max Chars field has been set to 75, while the Char Width is set to 30. This means that only 30 characters of the file path will be viewable to the user. Carefully consider the types of files the user will upload, and if they need to be able to view the entire file path within the file field's text box.



Creating a hidden field

Hidden fields are text fields that you use to collect information about users. Information stored in the field is sent back to the server when the form is submitted.

Dreamweaver creates a marker in the document when you insert a hidden field. If you insert a hidden field and don't see a marker, choose View > Visual Aids > Invisible Elements to see the marker.

To create a hidden field:

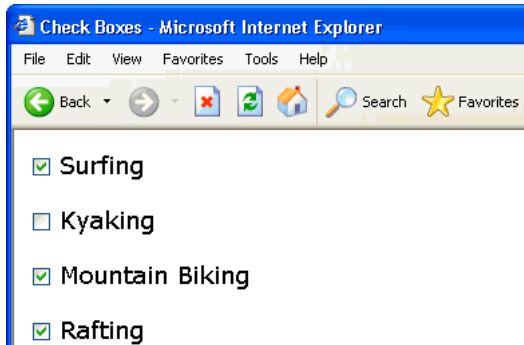
- 1 Place the insertion point inside the form outline.
- 2 Choose Insert > Form Objects > Hidden Field.
A marker appears in the document.
- 3 In the Property inspector's HiddenField field, type a unique name for the field.
- 4 In the Value field, type the value you want to assign the field.

Inserting checkboxes and radio buttons

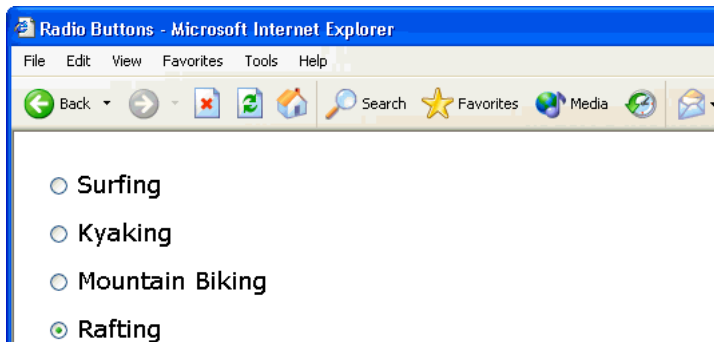
Check box and radio button form objects set up predefined selection objects. The user clicks a checkbox or radio button to make a choice.

Understanding checkboxes and radio buttons

Check boxes toggle each individual response “off” and “on.” Thus, a user can select more than one option from a checkbox group. The example below illustrates this by showing three checkbox items selected: Surfing, Mountain Biking, and Rafting.



Radio buttons work as a group and provide mutually exclusive selection values. A user can select only one option within a radio button group. In the example below, “Rafting” is the currently selected option. If the user were to click “Surfing,” the “Rafting” button would be automatically deselected.



Inserting a checkbox

Check boxes let users select more than one option from a set of options.

To insert a checkbox:

- 1 Place the insertion point inside the form outline, then do one of the following:
 - Choose Insert > Form Objects > Check Box.
 - In the Insert bar’s Forms category, click the Insert Checkbox icon.

- 2 In the Property inspector's CheckBox Name field, type a unique, descriptive name for the checkbox.
- 3 In the Checked Value field, type a value for the checkbox.
For example, in a survey you might set a value of 4 for strongly agree and a value of 1 for strongly disagree.
- 4 For Initial State, click Checked if you want an option to appear selected when the form first loads in the browser.

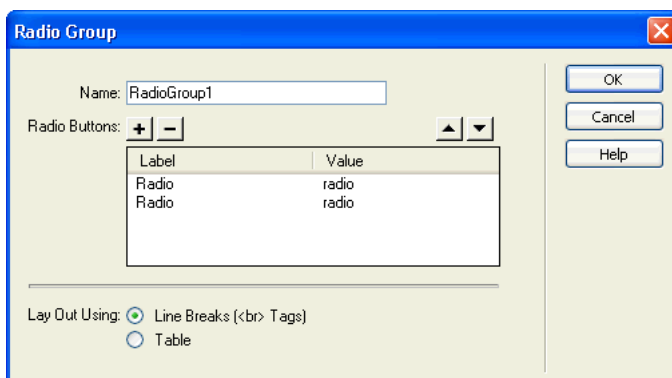
Inserting a radio button

Use radio buttons when the user must select only one choice from a set of options. Radio buttons are typically used in groups. All radio buttons in a group must have the same name and must contain different field values.

To insert radio buttons as a group:

- 1 Place the insertion point inside the form outline:
- 2 Insert the Radio Group form object by either:
 - Clicking the Forms tab and click the Radio Group icon.
 - Choose Insert > Form Objects > Radio Group.

The Radio Group dialog box appears.



- 3 Complete the Radio Group dialog box and click OK.

For instructions on completing the Radio Group dialog box, click the Radio Group dialog box's Help button.

Dreamweaver inserts the radio button group within the HTML form. If you haven't inserted a form on your page yet, Dreamweaver inserts one for you. If you want, you can change the layout of the group. You can also edit the radio buttons using the Property inspector or directly in Code view.

To insert radio buttons one at a time:

- 1 Place the insertion point inside the form outline.
- 2 Insert a radio button form object by choosing Insert > Form Objects > Radio button.

- 3 In the Property inspector's RadioButton Name field, type a descriptive name for the group of choices.

If you are creating multiple radio button interactions in a form, be certain that each set of radio buttons has a unique name. If however you are inserting individual radio buttons to create a group, you must label each

- 4 In the Checked Value field, enter the value you want sent to the server-side script or application when a user selects this radio button. For example, you might type **skiing** in the CheckedValue field to indicate a user chose skiing.
- 5 For Initial State, click Checked if you want an option to appear selected when the form first loads in the browser.

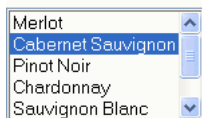
Adding lists and menus

Lists and menus let you present users with many choices within a limited space. Lists provide a scroll bar that lets users navigate through many items, and make multiple selections. Drop-down menus display a single item, which is also the active selection. Users can choose only one item from a menu.

Creating a scrolling list

Scrolling lists allow you to display multiple options in a confined space. The user can scroll through the list, and select multiple items.

What wine varietal are you looking for?



To create a scrolling list:

- 1 Place the insertion point inside the form outline.
- 2 Choose Insert > Form Objects > List/Menu and select the resulting List/Menu element if necessary.

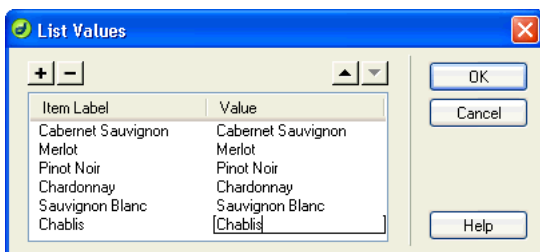
The List/Menu form object will appear within the form outline, and the List/Menu Property inspector will display. In the illustration below, the List/Menu Property inspector displays the Dynamic button. This button is used to create dynamic list/menu form objects, and only displays if a dynamic document type has been previously selected.



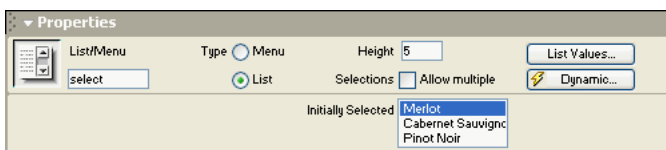
- 3 In the Property inspector's List/Menu field, enter a unique name for the list.
- 4 Under Type, select List.
- 5 In the Height field, enter a number to specify the number of lines (or items) the list will display. Scroll bars appear when the number you specify is less than the number of options the list contains.

- 6 If you want to allow users to select more than one item in the list, select Allow Multiple (next to Selections).
- 7 Click List Values to add the option choices.

The List Values dialog box appears.



- 8 With the insertion point in the Item Label field, enter the text you want to appear in the list.
- 9 In the Value field, enter data you want sent to the server when a user selects the item.
- 10 To add another item to the options list, click the plus (+) button, then repeat steps 7 and 8.
- 11 When you finish adding items to the list, click OK to close the List Values dialog box.
The Property inspector appears. The option choices are visible in the Initially Selected field.
- 12 To have one of the items in the list be selected by default, select that item in the Initially Selected field of the Property inspector.



Creating pop-up menus

A pop-up menu lets a visitor select a single item from a list of many. Pop-up menus are useful when you have a limited amount of space, but need to display many items. Note that only one option choice is visible when the menu form is displayed in a browser. To display the other choices, the user clicks the down arrow to display the entire list.

What wine varietal are you looking for?

Note: The pop-up menu form element is not the same as a graphical pop-up menu. For information on creating, editing, and showing and hiding a graphical pop-up menu, see “Show Pop-Up Menu” on page 370.

To create a pop-up menu:

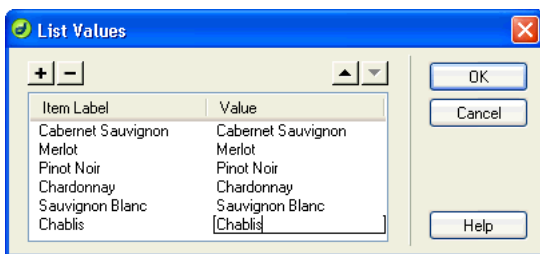
- 1 Place the insertion point inside the form outline.
- 2 Choose Insert > Form Objects > List/Menu and select the resulting List/Menu element if necessary.

The List/Menu form object will appear within the form outline, and the List/Menu Property inspector will display.



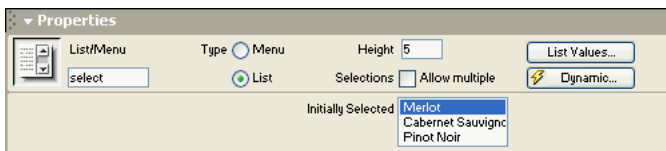
- 3 In the Property inspector's List/Menu field, enter a unique name for the menu.
- 4 Under Type, select Menu.
- 5 Click List Values to add the option choices.

The List Values dialog box appears.



- 6 With the insertion point in the Item Label field, enter the text you want to appear in the menu.
- 7 In the Value field, enter data you want sent to the server when a user selects the item.
- 8 To add another item to the options list, click the plus (+) button, then repeat steps 6 and 7.
- 9 When you finish adding items to the menu, click OK to close the List Values dialog box.

- 10 To have one of the items in the list be selected by default, select that item in the Initially Selected field of the Property inspector.



Adding form buttons

Form buttons control form operations. Use a form button to submit data entered into a form to the server, or to reset the form. You can also assign other processing tasks that you've defined in a script to a button. For example, the form button might calculate the total cost of items selected based on assigned values.

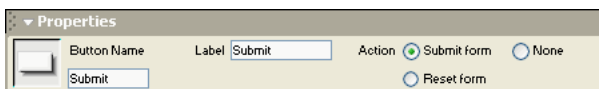
Inserting standard form buttons

Standard form buttons are the browser's default button style, which contain text you want to display. Standard form buttons are typically labeled Submit, Reset, or Send.

To create a text button:

- 1 Place the insertion point inside the form outline.
- 2 Choose Insert > Form Objects > Button, or click the Button icon in the Form panel.

Dreamweaver inserts the Button form object into the form, and displays the Button Property inspector.



- 3 Enter a name for the button in the Property inspector's Button Name field.
Note that there are two reserved names: "Submit," which submits the form data for processing, and "Reset," which resets all the form fields to their original values.
- 4 Enter the text you want to appear on the button in the Property inspector's Label field.
- 5 Select an action from the Action section. The available actions are:
 - Submit** the form for processing when the button is clicked.
 - Reset** the form when the button is clicked.
 - None** to activate an action based on the processing script when the button is clicked. To specify an action, select the Form tag from the Document window's status bar to select the form and display the Form Property inspector. In the Action pop-up menu, select a script or page to process the form.

Creating a graphical Submit button

You can use images you specify as button icons. Using an image to perform tasks other than submitting data requires attaching a behavior to the form object. You can assign a behavior to the button using the Dreamweaver Behaviors panel, or you can use client-side JavaScript to perform an action.

To learn how to attach a behavior to an object, see “Applying a behavior” on page 351.

To create a graphical Submit button:

- 1 In the document, place the insertion point inside the form outline.
- 2 Choose Insert > Form Objects > Image Field.
- 3 In the Property inspector for the Image field, change the text in the ImageField field to Submit.
- 4 Select the image you want to use as a button by clicking the SRC field’s folder icon, and browsing to the image file.
- 5 In the Alt field, enter any text you want to appear in place of the image for text-only browsers or for browsers set to download images manually.

About form design

You can use line breaks, paragraph breaks, preformatted text, or tables to format your forms. You cannot insert a form in another form (that is, you cannot overlap tags), but you can include more than one form in a page.

When designing forms, remember to label the form fields with descriptive text to let users know what they’re responding to—for example, “Type your name” to request name information.

Inserting tables in forms

You can design more attractive forms by inserting a table in a form. Use tables to provide structure for form objects and field labels. Tables make it easier to line up choices both vertically and horizontally. When using tables in forms make sure all the `table` tags are included between the `form` tags.

<input type="checkbox"/> Cabernet Sauvignon	<input type="checkbox"/> Chablis
<input type="checkbox"/> Merlot	<input type="checkbox"/> Chardonnay
<input type="checkbox"/> Pinot Noir	<input type="checkbox"/> Sauvignon Blanc

Using a client-side JavaScript function to process a form

Forms are processed by the script or application specified in the `action` attribute of the `form` tag. To see what action is associated with a form, select it and look in the Property inspector. The Action field will point to the associated action.

The simplest forms use JavaScript or VBScript to perform form processing on the client side (as opposed to sending the form data to the server for processing). For example, you might have a small form at the bottom of a page that contains only two radio buttons labeled Yes and No, plus a Submit button. The form action might be a JavaScript function defined in the head section of the document that displays one alert if the user selects Yes and another alert if the user selects No:

```
function processForm(){
    if (document.forms[0].elements[0].checked){
        alert('Yes');
    }else{
        alert('No');
    }
}
```

To use a client-side JavaScript function as the form action:

- 1 Select a Submit button in a form.
- 2 Attach the Call JavaScript behavior to the button (see “Call JavaScript” on page 354).
- 3 In the JavaScript text box that appears while attaching the behavior, enter `processForm()`.
- 4 Add a `processForm()` JavaScript function (like the one shown above) to the head section of the document.

You can handle many form-processing tasks using client-side scripting, however, you can't save the data entered by the user or send it to someone else. To save the data or transmit it to another application for processing, you need to use a server-side script or application. Common Gateway Interface (CGI) scripts are the most popular form of server-side scripting mechanisms to process form data. There are several sites on the web that offer free CGI scripts that you can use. The Internet Service Provider (ISP) hosting your site may provide CGI scripts that perform many common tasks such as collecting e-mail addresses, or allowing visitors to send you comments through a web form.

Using behaviors with forms

You can attach behaviors to forms and form objects by using any of the behaviors that appear in the Behaviors panel when the form or form object is selected. The Validate Form and Set Text of Text Field behaviors are available only if a text field has been inserted into the document. When attaching the Validate Form behavior to a form object, you must specify which text field is to be validated. For example, if you attach Validate Form to the Submit button, you might specify a text field created for “Name” to check that the user entered text in the Name field.

When applying behaviors, you need to make sure that every form object in your document (and every other object) has a unique name. If you use the same name for two different objects, behaviors may not work properly—even if the objects are in different forms.

The two form-specific behaviors are explained below. For information on other behaviors, see Chapter 24, “Using JavaScript Behaviors,” on page 349.

Validate Form checks the contents of specified text fields to ensure that the user has entered the correct type of data. (See “Validate Form” on page 376.)

Set Text of Text Field replaces the content of a specified text field with the content you specify. (See “Set Text of Text Field” on page 368.)

Creating dynamic form objects

You can create a dynamic HTML form to both display records from a database, and to collect information from users. For example, a form could display suppliers' contact information retrieved from a database table.

Dynamic forms can simplify site maintenance. For example, many sites use HTML form menus to let users jump to other areas of the site. Typically, this kind of menu appears in several places on the site. If the menu is dynamic, you can add a new menu item once—to the database table—to update all the menus on the site.

List/menus are common dynamic form objects. You can also create and use dynamic radio buttons, checkboxes, text fields, and image fields.

This section describes how to make form objects dynamic.

Note: Dreamweaver MX does not provide the following dynamic form objects for ASP.NET: Dynamic Text Field, Dynamic CheckBox, Dynamic List/Menu, or Dynamic Radio Group.

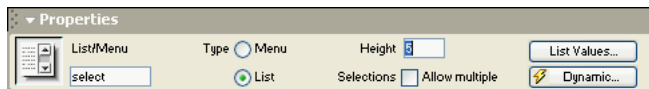
Creating a dynamic list/menu

You can dynamically populate a list or menu form object with entries from a database.

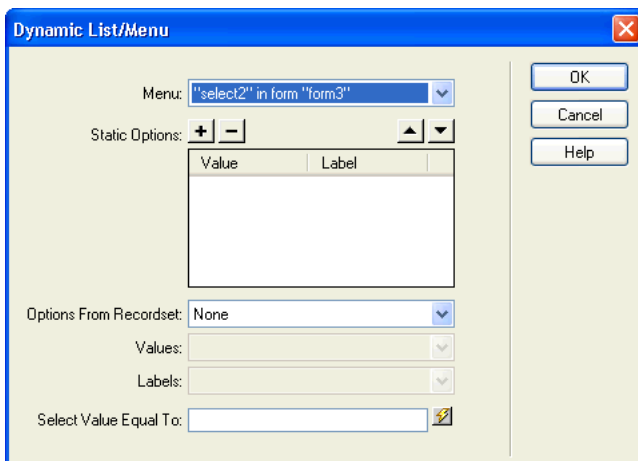
To add a dynamic list/menu:

- 1 Click inside the HTML form on your page.
- 2 Choose Insert > Form Objects > List/Menu
Dreamweaver inserts a List/Menu form object into the page.
- 3 Select the List/Menu form object.

The List/Menu property inspector displays.



- 4 In the The List/Menu property inspector, click the Dynamic button to display the Dynamic List/Menu dialog box. Complete the dialog box and click OK.



For instructions on completing the Dynamic List/Menu dialog box fields, click the Dynamic List/Menu dialog box's Help button.

- 5 Click OK.

Dreamweaver inserts the dynamic list/menu into the HTML form. You can edit the menu using the Server Behaviors panel or the Property inspector, or you can edit the menu's source code directly.

Making an existing list/menu dynamic

You can make an existing list/menu form dynamic.

To make an existing list/menu dynamic:

- 1 Select the list/menu form object you want to make dynamic.
- 2 In property inspector click the Dynamic button.
The Dynamic List/Menu dialog box appears.
- 3 Complete the dialog box and click OK.

For instructions on completing the Dynamic List/Menu dialog box, click dialog box's Help button.

Making text and image fields dynamic

You can make text and image fields on a form dynamic.

To make text fields dynamic:

- 1 Select a text field in the HTML form on your page.
- 2 Choose Window > Server Behaviors to display the Server Behaviors panel.
- 3 Click the plus (+) button and select Dynamic Elements > Dynamic Text Field from the pop-up menu.

The Dynamic Text Field dialog box appears.

- 4 Complete the dialog box and click OK.

For instructions in completing the dialog box, click its Help button.

The text field will display dynamic content when the form is viewed in a browser.

To make image fields dynamic:

- 1 Place the insertion point where you want the image field to appear on the page, then select Insert > Form Objects > Image Field.

The Select Image Source dialog box appears.

- 2 Click the Data Sources option (Windows) or the Data Source button (Macintosh).

A list of data sources appears.

- 3 Select a data source from the list.

The data source should be a recordset containing the paths to your image files. Depending on the file structure of your site, the paths can be absolute, document relative, or root relative.

Note: Dreamweaver does not currently support binary images stored in a database.

If no recordsets appear in the list, or if the available recordsets don't meet your needs, define a new recordset. For instructions, see "Defining a recordset" on page 502.

- 4 Click OK.

The image field will display a dynamically selected image when the form is viewed in a browser.

Making checkboxes dynamic

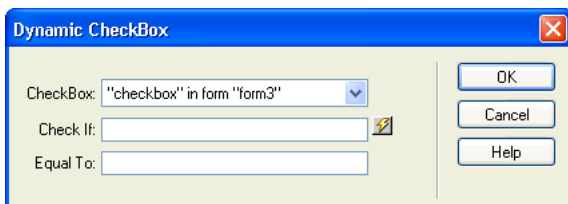
Use a checkbox in a form to let users choose multiple items from the list of available items. To create a dynamic checkbox you must have specified a dynamic document type, and associated a recordset or other source of dynamic content with the page. The data source must be in a Boolean form, such as Yes/No or true/false.

To dynamically pre-select a checkbox:

- 1 Select a checkbox form object.

- 2 In the Property inspector, click the Dynamic button.

The Dynamic Check Box dialog box appears.



- 3 Complete the dialog box and click OK.

For instructions on completing the Dynamic Check Box dialog box, click the dialog box's Help button.

The checkbox will appear selected or un-selected, depending on the data, when the form is viewed in a browser.

Making radio buttons dynamic

You can dynamically pre-select a radio button based on a value in a recordset. To create a dynamic checkbox you must have specified a dynamic document type, and associated a recordset or other source of dynamic content with the page.

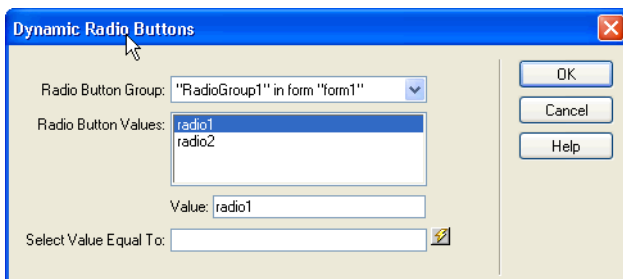
To dynamically pre-select a radio button:

- 1 Make sure the page has at least one group of radio buttons.

You can create a radio button group by selecting the Radio Group icon on the Forms tab of the Insert bar. For more information, see “Inserting a radio button” on page 585.

- 2 In the Server Behaviors Panel, click the plus (+) button and select Dynamic Form Elements > Dynamic Radio Group.

The Dynamic Radio Group dialog box appears. The dialog box that displays may appear different from the example shown below depending on the document type you are working with.



- 3 Complete the dialog box and click OK.

For instructions on completing the Dynamic Radio Buttons dialog box, click the dialog box's Help button.

Part IX

Developing Applications Rapidly

Web applications often feature pages that let users search a database, pages that let users insert, update, or delete data in a database, and pages that restrict access to a website. Quickly build any of these pages with Dreamweaver.

This part contains the following chapters:

- Chapter 39, “Building Master/Detail Pages”
- Chapter 40, “Building Pages that Search Databases”
- Chapter 41, “Building Pages that Modify Databases”
- Chapter 42, “Building Pages that Restrict Access to Your Site”

CHAPTER 39

Building Master/Detail Pages

With Macromedia Dreamweaver MX, you can create sets of pages that present information in two levels of detail. This chapter describes how to build master/detail pages in the following sections:

- “About master/detail pages” on page 599
- “Building master/detail pages rapidly” on page 600
- “Building master/detail pages block by block” on page 602
- “Modifying master/details pages” on page 605

About master/detail pages

A master page is a page that lists records and corresponding links for each record. When the user clicks a link, a detail page opens displaying more information about the record. For example, here’s a master page from the intranet site of a fictional company:



employees 1-5 of 5

Name	Department	Ext.	
BATES, Chris	Operations	3476	VIEW
DAVIS, Welan	Operations	3459	VIEW
GRANDEL, David	Operations	3458	VIEW
RIELY, Dan	Operations	6799	VIEW
SMITH, Ken	Operations	3479	VIEW

[ADD NEW EMPLOYEE](#)

When a user clicks one of the linked View icons, a detail page opens:



A results page is a common type of master page. However, unlike the master page described in this section, the list of records on a results page is determined not by you, the designer, but by the user. (The user determines the list by conducting a database search.) For more information on this type of master page, see “Building Pages that Search Databases” on page 607.

A detail page can be used to update or delete the record displayed. For more information on update and delete pages, see “Building a page to update records” on page 622 and “Building a page to delete a record” on page 626.

A master page consists of the following building blocks:

- A recordset
- A page layout to display multiple records
- A Go to Detail Page server behavior to open the detail page and pass the ID of the record the user clicked

A detail page consists of the following building blocks:

- A page layout to display a single record
- A recordset to hold the record’s details
- Either a recordset filter to retrieve a specific record from the database table, or a Move To Specific Record server behavior to move to a specific record in the recordset.

Building master/detail pages rapidly

You can quickly build master/detail pages using the Master/Detail Page Set live object.

Note: The live object is not supported in PHP or ASP.NET.

To complete the master/detail page set with a live object:

- 1 Create a blank master page and add a recordset to it.

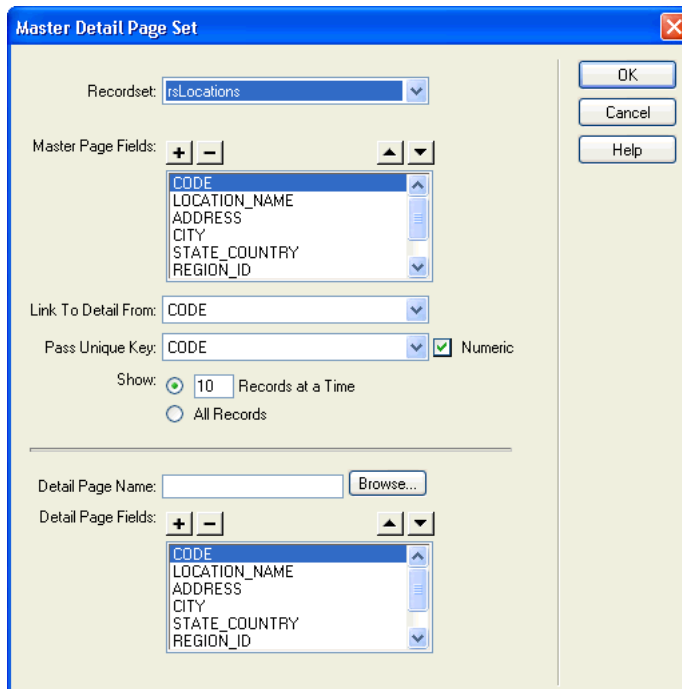
Make sure the recordset contains not only all the columns you'll need for the master page, but also all the columns you'll need for the detail page.

The recordset can be defined by you at design time (see “Defining a recordset” on page 502) or by the user at runtime (see “Building Pages that Search Databases” on page 607).

Typically, the recordset on the master page extracts a few columns from a database table while the recordset on the detail page extracts more columns from the same table to provide the extra detail.

- 2 Open the master page in Design view, and choose Insert > Application Objects > Master Detail Page Set.

The Master Detail Page Set dialog box appears.



- 3 Complete the dialog box.

For more information, click the Help button in the dialog box.

- 4 Click OK.

The live object creates a detail page (if you didn't already create one) and adds dynamic content and server behaviors to both the master and detail pages.

- 5 Customize the layout of the master and detail pages to suit your needs.

You can fully customize the layout of each page using the Macromedia Dreamweaver MX page-design tools. You can also edit the server behaviors by double-clicking them in the Server Behaviors panel.

Building master/detail pages block by block

This section describes how to build a set of master/detail pages without using the Master/Detail Page Set live object. For instructions on using the live object, see “Building master/detail pages rapidly” on page 600.

To complete the master/detail page set, you need to accomplish the following tasks:

- Create a blank master page and define a recordset for it.
- Display the records on the master page.
- Pass the ID of the record the user selected to the detail page.
- Find the record in the database and display it on the detail page.

Creating a master page and defining a recordset for it

The first step is to create a blank master page and add a recordset to it.

The recordset can be defined by you at design time (see “Defining a recordset” on page 502) or by the user at runtime (see “Building Pages that Search Databases” on page 607).

Make sure the recordset contains not only all the columns you’ll need for the master page, but also all the columns you’ll need for the detail page.

Typically, the recordset on the master page extracts a few columns from a database table while the recordset on the detail page extracts more columns from the same table to provide the extra detail.

Displaying the records on the master page

After you create a blank master page and define a recordset for it, you must display the records on the page.

To display the records on the master page:

- 1 Create a page layout to display multiple records and bind recordset columns to the page.

A common approach is to create a two-row HTML table on the master page and to drag a limited number of recordset columns from the Bindings panel (Window > Bindings) into the table’s second row. (Use the first row to display the table’s column headings.)

- 2 Create a repeated region to display more than one record at a time.

The repeated region is normally applied to the table row containing the dynamic content.

Opening the detail page and passing a record ID to it

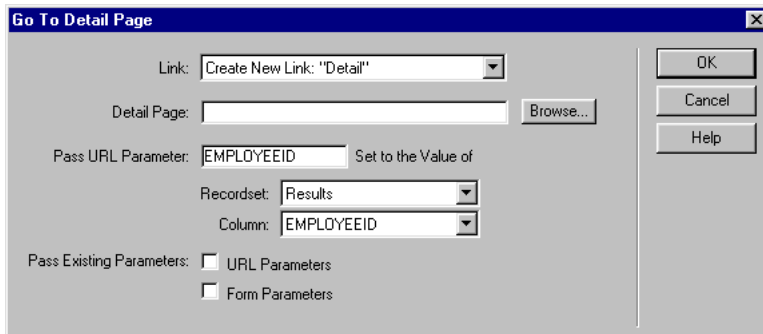
After creating the master page and displaying records on it, you must create links that open the detail page and pass the ID of the record the user selected to it.

PHP, ASP.NET, and ColdFusion users must create a link with URL parameters in the code. For more information, see “Creating URL parameters using HTML links” on page 495. Place the anchor tags around the dynamic content you want to serve as the link.

To open the detail page and pass it a URL parameter that contains the ID of the record the user clicked (ASP and JSP users only):

- 1 In the repeated region on the master page, select the dynamic content to double as a link.
- 2 In the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button, and choose Go to Detail Page from the pop-up menu.

The Go to Detail Page dialog box appears.



- 3 Complete the dialog box.

For more information, click the Help button in the dialog box.

- 4 Click OK.

Dreamweaver places a special link around the selected text. When the user clicks the link, the Go To Detail Page server behavior passes a URL parameter containing the record ID to the detail page. For example, if the URL parameter is called `id` and the detail page is called `customerdetail.asp`, then the URL will look something like the following when the user clicks on the link:

```
http://www.mysite.com/customerdetail.asp?id=43
```

The first part of the URL, `http://www.mysite.com/customerdetail.asp`, opens the detail page. The second part, `?id=43`, is the URL parameter. It tells the detail page what record to find and display. The term `id` is the name of the URL parameter and `43` is its value. In this example, the URL parameter contains the record's ID number, `43`.

For more information on URL parameters, see “Understanding URL parameters” on page 494.

Finding and displaying the requested record on the detail page

After completing the master page, you must find the requested record in the database and display it on the detail page. The procedure consists of defining a recordset to hold a single record—the record requested by the master page—and binding the recordset columns to the page.

To find and display the requested record on the detail page:

- 1 Switch to the detail page.
- 2 In the Bindings panel, click the plus (+) button and choose Recordset or DataSet (ASP.NET) from the pop-up menu.

The simple dialog box appears. If the advanced dialog box appears instead, click Simple to switch.

3 Name the recordset, then choose a connection and database table that will provide data to your recordset.

4 In the Column area, select the table columns to include in the recordset.

The recordset can be identical to or different from the recordset on the master page. Usually a detail page recordset has more columns to display more detail.

If the recordsets are different, the recordset on the detail page should contain at least one column in common with the master page. The common column is usually the record ID column, but it can also be the join field of related tables.

To include only some of the table's columns in the recordset, click Selected and choose the desired columns by Control-clicking (Windows) or Command-clicking (Macintosh) them in the list.

5 Complete the Filter section as follows to find and display the record specified in the URL parameter passed by the results page:

- From the first pop-up menu in the Filter area, select the column in the database table containing values that match the value of the URL parameter passed by the master page.

For example, if the URL parameter contains a record ID number, choose the column containing record ID numbers.

- From the pop-up menu beside the first menu, select the equal sign (it should already be selected).
- From the third pop-up menu, select URL Parameter.

The master page passes information identifying the user's selection to the detail page in a URL parameter.

- In the fourth text box, enter the name of the URL parameter passed by the master page.

For example, if the URL the master page used to open the detail page is `www.mysite.com/customerdetail.asp?id=43`, then enter `id`.

You can also get the name by switching to the master page, opening the Server Behaviors panel (Window > Server Behaviors), and double-clicking the Go to Detail Page server behavior. Check the Pass URL Parameter name.

6 Click OK.

The recordset appears in the Bindings panel.

7 Bind the recordset columns to the detail page by selecting the columns in the Bindings panel (Window > Bindings) and dragging them onto the page.

Modifying master/details pages

After creating master/detail pages with server behaviors, use the Server Behaviors panel to modify the various building blocks on the pages. For more information, see “Editing server behaviors” on page 571.

If you created pages with the live object, you must also use the Server Behaviors panel to edit the building blocks the live object inserts into the pages. The live object adds the following building blocks to the master page:

- A basic table with a repeated region to display multiple records
- A recordset navigation bar
- A record counter
- A Go to Detail Page server behavior to open the detail page and pass the ID of the record the user clicked

The live object also adds the following building blocks to the detail page:

- A basic table to display a single record
- A filtered recordset to find and display the record the user clicked on the master page

CHAPTER 40

Building Pages that Search Databases

You can use Macromedia Dreamweaver MX to build a set of pages to let users search your database.

This chapter contains the following sections:

- “Creating the search page” on page 608
- “Building the results page” on page 609
- “Creating a detail page for a results page” on page 613
- “Working with related pages (ASP and JSP users only)” on page 617

About search/results pages

You need at least two pages to add this feature to your web application. The first page is a page containing an HTML form in which users enter search parameters. Although this page doesn't do any actual searching, it is referred to as “the search page.”

The second page you need is the results page, the workhorse of the page set. The results page performs the following tasks:

- Reads the search parameters submitted by the search page
- Connects to the database and searches for records
- Builds a recordset with the records found
- Displays the contents of the recordset

Optionally, you can add a detail page. A detail page gives users more information about a particular record on the results page.

If you have only one search parameter, Dreamweaver lets you add search capabilities to your web application without using SQL queries and variables. Simply design your pages and complete a few dialog boxes. If you have more than one search parameter, you need to write a SQL statement and define multiple variables for it.

Creating the search page

A search page on the web typically contains form fields in which the user enters search parameters. When the user clicks the form's Search button, the search parameters are sent to a results page on the server. The results page on the server, not the search page on the browser, is responsible for retrieving records from the database.

To get started on this part of your web application, create two pages: a search page that lets users enter the search parameters, and a results page to display the records found. You can even combine the two pages into one. At minimum, your search page must have an HTML form with a Submit button.

To add an HTML form to a search page:

- 1 Open the search page and select Form from the Insert menu.

An empty form is created on the page. You may have to turn on Invisible Elements (View > Visual Aids > Invisible Elements) to see the form's boundaries, which are represented by thin red lines.

- 2 Add form objects for users to enter their search parameters by choosing Form Objects from the Insert menu.

Form objects include text fields, list menus, checkboxes, and radio buttons. You can add as many form objects as you want to help users refine their searches. However, keep in mind that the greater the number of search parameters on the search page, the more complex your SQL statement will be.

For more information on form objects, Chapter 38, "Creating Interactive Forms," on page 573.

- 3 Add a Submit button to the form (Insert > Form Objects > Button).
- 4 If you wish, change the label of the Submit button by selecting the button, opening the Property inspector (Window > Properties), and entering a new value in the Label box.

For example, here's the Property inspector of a button labeled "search":



Next, you'll tell the form where to send the search parameters when the user clicks the Submit button.

- 5 Select the form by selecting the `<form>` tag in the tag selector at the bottom of the Document window, as shown:



- 6 In the Action box on the form's Property inspector, enter the filename of the results page that will conduct the database search.

7 In the Method pop-up menu, choose one of the following methods to determine how the form sends data to the server:

- `GET` sends the form data by appending it to the URL as a query string. Because URLs are limited to 8192 characters, don't use the `GET` method with long forms.
- `POST` sends the form data in the body of a message.
- `Default` uses the browser's default method (usually `GET`).

The search page is done. Next comes the results page.

Building the results page

After the user enters the search parameters, your application must retrieve the records from the database. This work is done by the results page.

Here's the job of a results page:

- Get the search parameters from the search page.
- Connect to the database and search for records.
- Build a recordset with the records found.
- Display the contents of the recordset.

If your search page has only one search parameter (a single text field, for example), you can build the results page without SQL queries and variables. You simply create a basic recordset, then add a filter to it to exclude records that don't meet the search parameters sent by the search page. For instructions, see "Searching with only one search parameter" on page 609.

If your search page has more than one search parameter, you need to write a SQL statement and define multiple variables for it. For instructions, see "Searching with multiple search parameters" on page 611.

Searching with only one search parameter

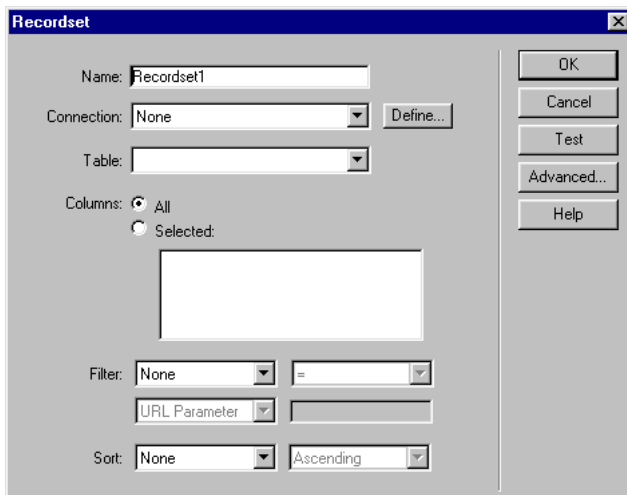
If the search page submits a single search parameter to the server, then you can build the results page without SQL queries and variables. You create a basic recordset with a filter that excludes records that don't meet the search parameter submitted by the search page.

Note: If you have more than one search condition, you must use the advanced Recordset dialog box to define your recordset. The simple Recordset dialog box only supports one search condition. For more information, see "Searching with multiple search parameters" on page 611.

To create the recordset to hold the search results:

- 1 Open your results page in the Document window.
- 2 Create a new recordset by opening the Bindings panel (Window > Bindings), clicking the plus (+) button, and selecting Recordset or DataSet (ASP.NET) from the pop-up menu.

- 3 Make sure the simple Recordset or DataSet dialog box appears.



If the advanced dialog box appears instead, switch to the simple dialog box by clicking the Simple button.

- 4 Enter a name for the recordset and choose a connection.

The connection should be to a database containing data you want the user to search.

- 5 In the Table pop-up menu, select the table to be searched in the database.

Note: In a single-parameter search, you can search for records in only a single table. To search more than one table at a time, you must use the advanced Recordset dialog box and define a SQL query.

- 6 To include only some of the table's columns in the recordset, click Selected and choose the desired columns by Control-clicking (Windows) or Command-clicking (Macintosh) them in the list.

Choose columns containing information you want to display on the results page.

Leave the Recordset or DataSet dialog box open for now. You'll use it next to fetch the parameters sent by the search page and create a recordset filter to exclude records that don't meet the parameters.

To create the recordset filter:

- 1 From the first pop-up menu in the Filter area, select a column in the table to compare against the search parameter sent by the search page.

For example, if the value sent by the search page is a city name, select the column in your table that contains city names.

- 2 From the pop-up menu beside the first menu, select the equal sign (it should be the default).

This choice states that the user wants only those records in which the selected table column is exactly the same as the one specified on the search page.

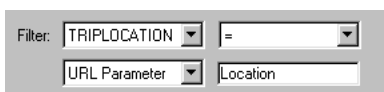
- 3 From the third pop-up menu, select Form Variable if the form on your search page uses the POST method, or URL Parameter if it uses the GET method.

This pop-up menu specifies where the value sent by the search page is currently stored on the server. In ASP, if the search form uses the POST method, the value is stored in the Request.Form collection. If the search form uses the GET method, the value is stored in the Request.QueryString collection.

- 4 In the fourth box, enter the name of the form object that accepts the search parameter on the search page.

You can get the name by switching to the search page, clicking the form object on the form to select it, and checking the object's name in the Property inspector.

For example, suppose you want to create a recordset that includes only adventure trips to a specific country. Assume you have a column in the table called TRIPLOCATION. Also assume the HTML form on your search page uses the GET method and contains a Menu/List object called Location that displays a list of countries. Here's how your Filter section should look:



The image shows a screenshot of a 'Filter' section in a software interface. It consists of several elements: a label 'Filter:', a dropdown menu with 'TRIPLOCATION' selected, an equals sign, another dropdown menu, a second dropdown menu with 'URL Parameter' selected, and a text input field containing the word 'Location'.

- 5 If you want to, click Test, enter a test value, and click OK to connect to the database and create an instance of the recordset.

The test value simulates the value that would otherwise have been returned from the search page. Click OK to close the recordset.

- 6 If you're satisfied with the recordset, click OK.

Dreamweaver inserts a server-side script on your page that, when run on the server, checks each record in the database table. If the specified field in a record meets the filtering condition, the record is included in a recordset. The script in effect builds a recordset containing only the search results.

The next step is to display the recordset on the results page. For more information, see "Displaying the records" on page 613.

Searching with multiple search parameters

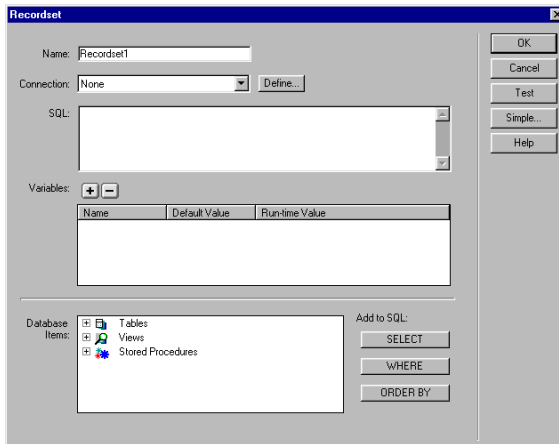
If the search page submits more than one search parameter to the server, then you must write a SQL query for the results page and use the search parameters in SQL variables.

After Dreamweaver inserts the SQL query in your page and the page runs on the server, each record in the database table is checked. If the specified field in a record meets your SQL query conditions, the record is included in a recordset. The SQL query in effect builds a recordset containing only the search results.

For example, field sales staff may have the ability to tell which customers in a certain area have incomes above a certain level. In a form on a search page, the sales associate enters a geographical area and a minimum income level, then clicks the Submit button to send the two values to a server. On the server, the values are passed to the results page's SQL statement, which then creates a recordset containing only customers in the specified area with incomes above the specified level.

To search for records in a database using SQL:

- 1 Open the results page in Dreamweaver, then create a new recordset by opening the Bindings panel (Window > Bindings), clicking the plus (+) button, and selecting Recordset or DataSet (ASP.NET) from the pop-up menu.
- 2 Make sure the advanced Recordset or DataSet dialog box appears.



If the simple dialog box appears instead, switch to the advanced dialog box by clicking the Advanced button.

- 3 Enter a name for the recordset and choose a connection.

The connection should be to a database containing data you want the user to search.

- 4 Enter a Select statement in the SQL text area.

Make sure the statement includes a Where clause with variables to hold the search parameters. In the following example, the variables are called `varLastName` and `varDept`:

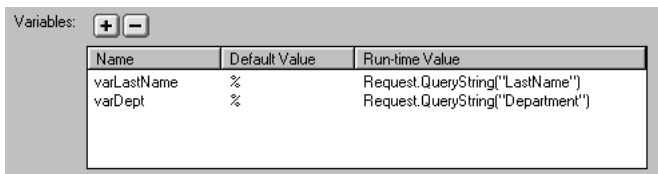
```
SELECT EMPLOYEEID, FIRSTNAME, LASTNAME, DEPARTMENT, EXTENSION
FROM EMPLOYEE WHERE LASTNAME LIKE 'varLastName' AND DEPARTMENT
LIKE 'varDept'
```

To reduce the amount of typing, you can use the tree of database items at the bottom of the advanced Recordset dialog box. For instructions, see “Creating an advanced recordset by writing SQL” on page 504.

For help on SQL syntax, see “SQL Primer” on page 663.

- 5 Give the SQL variables the values of the search parameters by clicking the plus (+) button in the Variables area and entering the variable's name, default value (the value the variable should take if no runtime value is returned), and runtime value (usually a server object holding a value sent by a browser, such as an request variable).

In the following ASP example, the HTML form on the search page uses the GET method and contains one text field called "LastName" and another called "Department".



In Macromedia ColdFusion, the runtime values would be `#LastName#` and `#Department#`. In JSP, the runtime values would be `request.getParameter("LastName")` and `request.getParameter("Department")`.

- 6 If you want to, click Test to create an instance of the recordset using the default variable values. The default values simulate the values that would otherwise have been returned from the search page. Click OK to close the test recordset.
- 7 If you're satisfied with the recordset, click OK.

Displaying the records

After creating a recordset to hold the search results, you may want to display the information on the results page. Displaying the records is a simple matter of dragging individual columns from the Bindings panel to the results page. You can add navigation links to move forward and backward through the recordset, or you can create a repeated region to display more than one record on the page. You can also add links to a detail page.

For more information about displaying dynamic content on a page, see:

- "Creating a detail page for a results page" on page 613
- "Adding Dynamic Content to Web Pages" on page 519
- "Displaying Database Records" on page 527

Creating a detail page for a results page

Your set of search pages can include a detail page to display more information about specific records on the results page. On the results page, the records are typically displayed in a repeated region, and each record has a link. When a user clicks one of the links, the detail page opens and displays more information about the selected record.

This section contains the following topics:

- "Modifying a results page to work with a detail page" on page 614
- "Building the detail page using a server behavior (ASP and JSP users only)" on page 615
- "Building the detail page using a filtered recordset" on page 616

Modifying a results page to work with a detail page

Your results page should have a repeated region to display more than one record at a time, and each record in the repeated region should have a link to the detail page. The link must not only open the detail page, it must also tell the detail page what record the user selected. The Go to Detail Page server behavior creates this kind of link (for ASP and JSP users only).

PHP, ASP.NET, and ColdFusion users must create a link with URL parameters in the code. For more information, see “Creating URL parameters using HTML links” on page 495.

To modify the results page to work with a detail page (ASP and JSP users only):

- 1 Create a blank detail page (File > New) and name the file.

Set aside the page for now. You'll work on it later.

- 2 Open the results page in Dreamweaver.

- 3 Make sure the results are displayed in a repeated region.

For instructions, see “Displaying multiple behaviors” on page 533.

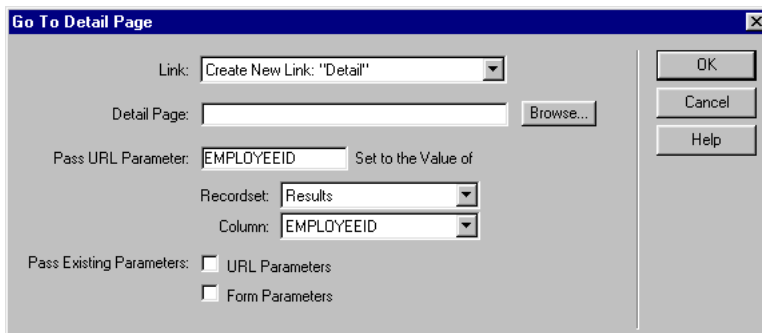
- 4 In the repeated region, select the text or image to double as a link.

If you're working in the Live Data window, select the text or image in the first region displayed.

Note: The text or image you select can be dynamic.

- 5 Create the link to the detail page by opening the Server Behaviors panel (Window > Server Behaviors), clicking the plus (+) button, and choosing Go to Detail Page from the pop-up menu.

The Go to Detail Page dialog box appears.



- 6 Complete the dialog box.

For more information, click the Help button in the dialog box.

- 7 Click OK.

The results page passes the value to the detail page in a URL parameter, which is simply a variable appended to the end of the URL used to open the detail page. For example, if the URL parameter is called `id` and the detail page is called `customerdetail.asp`, then the URL will look something like the following when the user clicks on the link:

`http://www.mysite.com/customerdetail.asp?id=43`

The first part of the URL, `http://www.mysite.com/customerdetail.asp`, opens the detail page. The second part, `?id=43`, is the URL parameter. It tells the detail page what record to find and display. The term `id` is the name of the URL parameter and `43` is its value. In this example, the URL parameter contains the record's ID number, `43`.

Building the detail page using a server behavior (ASP and JSP users only)

After modifying the results page to work with your detail page, complete the detail page. You can build the detail page with a combination of a regular recordset and a server behavior, or you can build one with only a filtered recordset. This section describes how to build the first kind of detail page (for ASP and JSP users only). To learn how to build the second kind, see “Building the detail page using a filtered recordset” on page 616.

First, lay out the detail page using the Dreamweaver design tools. For more information, see “Building Master/Detail Pages” on page 599.

Second, define a recordset for the page, or copy and paste the recordset from the results page. The detail page will extract the record details from this recordset. For instructions, see “Defining a recordset” on page 502 and “Copying a recordset from one page to another page” on page 517.

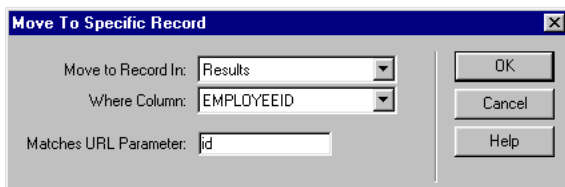
Third, bind the recordset columns to the page. In the Bindings panel (Window > Bindings), select columns in the recordset and drag them onto the page.

Fourth, add a server behavior that reads the record ID in the URL parameter passed by the results page and retrieves the record. If you omit this step, the server will retrieve the first record in the recordset.

To retrieve a specific record using a server behavior (ASP and JSP users only):

- 1 In the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button, choose Move to Record from the pop-up menu, then choose Move to Specific Record.

The Move to Specific Record dialog box appears.



- 2 Complete the dialog box.

For more information, click the Help button in the dialog box.

- 3 Click OK.

The next time the page is requested by a browser, it will read the record ID in the URL parameter passed by the browser and move to the specified record in the recordset.

Building the detail page using a filtered recordset

Another approach to building a detail page is to filter your recordset so that only a single record remains—the record the user selected on the results page. This method can improve the performance of your application because the recordset contains only one record.

First, lay out the page using the Dreamweaver design tools. For more information, see “Building Master/Detail Pages” on page 599.

Second, define a recordset for the page, or copy and paste the recordset from the results page. The detail page will extract the record details from this recordset. For instructions, see “Defining a recordset” on page 502 and “Copying a recordset from one page to another page” on page 517.

Third, create a recordset filter to retrieve the record specified on the results page. If you create a recordset using the simple Recordset dialog box, you can use the Filter boxes to create the filter. If you use the advanced Recordset dialog box, you can modify your SQL query to create the filter.

To retrieve a specific record using a recordset filter:

- 1 Make sure the detail page contains a recordset.
- 2 Open the recordset by double-clicking its name in the Bindings panel (Window > Bindings).
- 3 Make sure the simple Recordset or DataSet dialog box appears.

If the advanced dialog box appears instead, switch to the simple dialog box by clicking Simple. If Dreamweaver informs you that it can't switch (usually because your query is too complex to display in the simple dialog box), you must use a SQL query to find the record; please skip to the next procedure in this section.

- 4 Complete the Filter section as follows to find and display the record specified in the URL parameter passed by the results page:

- From the first pop-up menu in the Filter area, select the column in the database table containing values that match the value of the URL parameter passed by the results page.

For example, if the URL parameter contains a record ID number, choose the column containing record ID numbers.

- From the pop-up menu beside the first menu, select the equal sign (it should already be selected).
- From the third pop-up menu, select URL Parameter.

The results page passes information identifying the user's selection to the detail page in a URL parameter.

- In the fourth text box, enter the name of the URL parameter passed by the results page.

For example, if the URL the results page used to open the detail page is `www.mysite.com/customerdetail.asp?id=43`, then enter `id`.

If you used the Go To Detail Page server behavior on the results page, you can also get the name by switching to the results page, opening the Server Behaviors panel (Window > Server Behaviors), and double-clicking the Go to Detail Page server behavior. Check the Pass URL Parameter name.

- 5 Click OK.
- 6 If not already done, bind the recordset columns to the page by selecting the columns in the Bindings panel (Window > Bindings) and dragging them onto the page.

To retrieve a specific record using a SQL query:

- 1 Make sure the detail page contains a recordset.
- 2 Open the recordset by double-clicking its name in the Bindings panel (Window > Bindings).
- 3 Make sure the advanced Recordset dialog box appears.

If the simple Recordset dialog box appears instead, switch to the advanced Recordset dialog box by clicking the Advanced button.

- 4 Add a Where clause in your SQL statement to find the record the user selected on the results page.

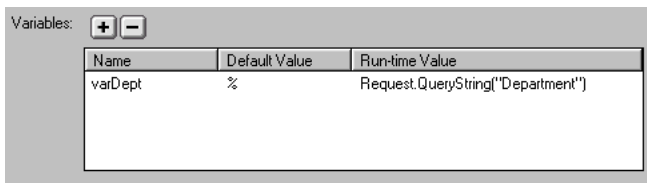
The Where clause should contain a variable to hold the value passed in the URL parameter. In the following example, the variable is called `varDept`:

```
SELECT * FROM EMPLOYEES  
WHERE DEPARTMENT = 'varDept'
```

For help on SQL syntax, see “SQL Primer” on page 663.

- 5 Give the variable the value the results page passed in the URL parameter by clicking the plus (+) button in the Variables area and entering the variable’s name, default value (the value the variable should take if no runtime value is returned), and runtime value.

In the following ASP example, the results page passes a URL parameter called `Department`.



- 6 Click OK.
- 7 If not already done, bind the recordset columns to the page by selecting the columns in the Bindings panel (Window > Bindings) and dragging them onto the page.

Working with related pages (ASP and JSP users only)

In some situations, you may want to display a page that is not a search page, a results page, or a detail page, but you don’t want to lose the information the page has received from an HTML form or a URL parameter. Instead of using a standard link to open the related page, create the link using the Go to Related Page server behavior. The resulting link opens the related page, and passes existing parameters to that page. For example, you can pass search parameters from one page to another and save the user from entering the search parameters again.

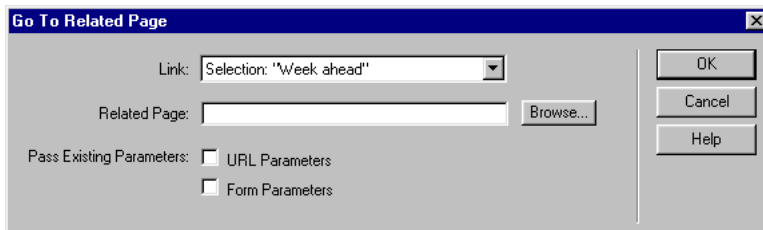
Note: This server behavior is only available for ASP and JSP pages.

Before adding a Go to Related Page server behavior to a page, make sure the page receives parameters from an HTML form (in other words, the form’s `ACTION` attribute specifies the page), or from a URL parameter, such as when the page is the link destination of another page with a Go to Related Page server behavior.

To create a link that passes existing form parameters to a related page:

- 1 On the page, select the text string or images to act as a link to the related page.
- 2 In the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button and choose Go to Related Page from the pop-up menu.

The Go to Related Page dialog box appears.



- 3 Complete the dialog box.

For more information, click the Help button in the dialog box.

- 4 Click OK.

When a user clicks the new link, the page passes the parameters to the related page using URL parameters. For example, suppose a form parameter is called “lastname” and the related page is called special_offer.asp. The URL will look something like the following when the user clicks the link:

`http://www.mysite.com/special_offer.asp?lastname=Anderson`

The first part of the URL, `http://www.mysite.com/special_offer.asp`, opens the related page. The second part, `?lastname=Anderson`, is the URL parameter that passes the original form parameter to the related page.

CHAPTER 41

Building Pages that Modify Databases

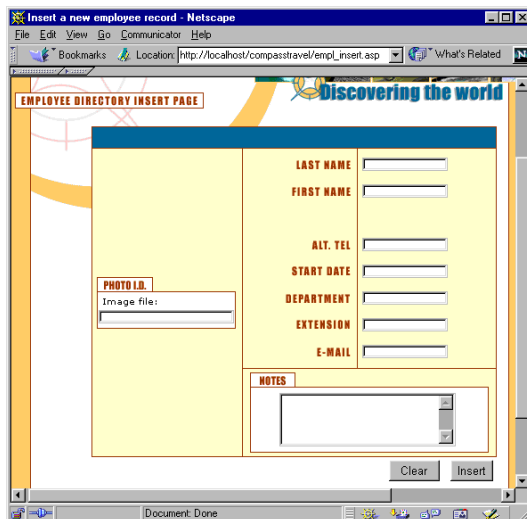
Macromedia Dreamweaver MX comes with a set of server behaviors that let users add, update, and delete records with their web browsers. You can also use application objects to create fully functional HTML forms for inserting or updating records.

This chapter contains the following sections:

- “Building a page to insert records” on page 619
- “Building a page to update records” on page 622
- “Building a page to delete a record” on page 626
- “Using stored procedures to modify databases” on page 629
- “Using ASP commands to modify a database” on page 634
- “Using JSP prepared statements to modify a database” on page 636

Building a page to insert records

Your application can contain a page that lets users insert new records in a database. For example, the following page inserts a new record in an employee database:



An insert page requires two building blocks:

- An HTML form that lets users enter data
- An Insert Record server behavior that updates the database

You can add these building blocks in a single operation using the Record Insertion Form live object, or you can add them separately using the Dreamweaver form tools and the Server Behaviors panel.

Note: The insert page can contain only one record-editing server behavior at a time. For example, you cannot add an Update Record or a Delete Record server behavior to the insert page.

Building an insert page rapidly

You can add the basic building blocks of an insert page in a single operation using the Record Insertion Form live object. The live object adds both an HTML form and an Insert Record server behavior to the page.

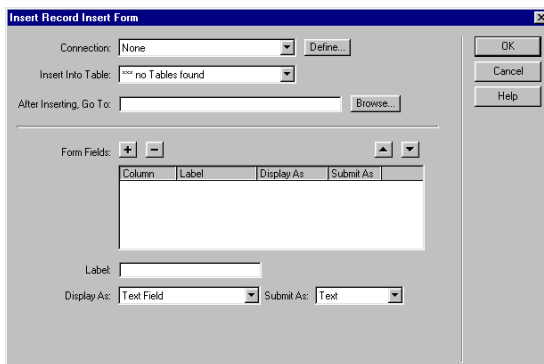
You can also add the building blocks separately using the form tools and the Server Behaviors panel. For more information, see “Building an insert page block by block” on page 621.

After placing the building blocks on the page, you can use the Dreamweaver design tools to customize the form as you desire, or the Server Behaviors panel to edit the Insert Record server behavior.

To build the insert page with the Record Insertion Form live object:

- 1 Open the page in Design view, then choose Insert > Application Objects > Record Insertion Form.

The Insert Record Insertion Form dialog box appears.



- 2 Complete the dialog box.

For instructions, click the Help button in the dialog box.

- 3 Click OK.

Dreamweaver adds both an HTML form and an Insert Record server behavior to your page. The form objects are laid out in a basic table, which you can freely customize using the Dreamweaver page design tools. (Make sure all the form objects remain within the form’s boundaries.)

To edit the server behavior, open the Server Behaviors panel (Window > Server Behaviors) and double-click the Insert Record behavior.

Building an insert page block by block

You can add the basic building blocks of an insert page separately using the form tools and the Server Behaviors panel.

You can also add the building blocks all at once using the Record Insertion Form live object. For more information, see “Building an insert page rapidly” on page 620.

The first step is to add an HTML form to the page to let users enter data.

To add an HTML form to an insert page:

- 1 Create a new page (File > New) and lay out your page using the Dreamweaver design tools.
- 2 Add an HTML form by placing the insertion point where you want the form to appear and choosing Form from the Insert menu.

An empty form is created on the page. You may have to turn on Invisible Elements (View > Visual Aids > Invisible Elements) to see the form’s boundaries, which are represented by thin red lines.

- 3 Name the HTML form by clicking the `<form>` tag at the bottom of the Document window to select the form, opening the Property inspector (Window > Properties), and entering a name in the Form Name box.

You don’t have to specify an `action` or `method` attribute for the form to tell it where and how to send the record data when the user clicks the Submit button. The Insert Record server behavior sets these attributes for you.

- 4 Add a form object (Insert > Form Objects) for each column in the database table you want to insert records into.

The form objects are for data entry. Text fields are common for this purpose, but you can also use list/menus, checkboxes, and radio buttons.

For more information on form objects, see “Creating Interactive Forms” on page 573.

- 5 Add a Submit button to the form (Insert > Form Objects > Button).
- 6 If you wish, change the label of the Submit button by selecting the button, opening the Property inspector (Window > Properties), and entering a new value in the Label box.

For example, here’s the Property inspector of a button labeled “Insert Record”:

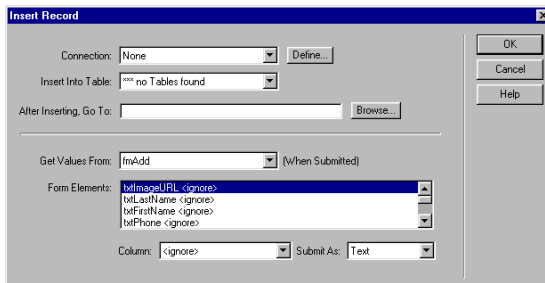


The next step is to add the Insert Record server behavior to insert records in a database table.

To add a server behavior to insert records in a database table:

In the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button and choose Insert Record from the pop-up menu (ASP.NET users choose Insert Record on form Submit).

The Insert Record dialog box appears.



1 Complete the dialog box.

For instructions, click the Help button in the dialog box.

2 Click OK.

Dreamweaver adds a server behavior to the page that lets users insert records in a database table by filling out the HTML form and clicking the Submit button.

Building a page to update records

Your application can contain a page that lets users update existing records in a database table. An update page is usually a detail page working in tandem with a results page. The results page lets the user choose a record to update, then passes the choice to the update page.

An update page has three building blocks:

- A filtered recordset to retrieve the record from a database table
- An HTML form to let users modify the record's data
- An Update Record server behavior to update the database table

You can add the HTML form and the server behavior to the page in a single operation using the Record Update Form live object, or you can add them separately using the Dreamweaver form tools and the Server Behaviors panel.

Note: The update page can contain only one record-editing server behavior at a time. For example, you cannot add an Insert Record or a Delete Record server behavior to the update page.

Identifying the record to update

When users want to update a record, they must first find that record in the database. Accordingly, you need a search and a results page to work with the update page. For instructions on creating a search and a results page, see “Building Pages that Search Databases” on page 607.

The results page tells the update page which record to update by passing it a URL parameter. Therefore, make sure the results page has a Go to Detail Page server behavior that names the update page as the detail page. For instructions, see “Modifying a results page to work with a detail page” on page 614.

Retrieving the record to update

After the results page passes a URL parameter to the update page identifying the record to update, the update page must read the parameter, retrieve the record from the database table, and store it temporarily in a recordset.

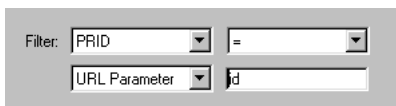
To retrieve the record to update:

- 1 In the Bindings panel (Window > Bindings), click the plus (+) button and choose Recordset or DataSet (ASP.NET).

If the advanced dialog box appears, click the Simple button to open the simple dialog box.

- 2 Name the recordset and specify where the data you want to update is located using the Connection and Table pop-up menus.
- 3 Click the Selected option and select a key column (usually the record ID column) and the columns containing the data to be updated.
- 4 Configure the Filter area so that the value of your key column equals the value of the corresponding URL parameter passed by the results page.

This kind of filter creates a recordset that contains only the record specified by the results page. For example, if your key column contains record ID information and is called PRID, and if the results page passes the corresponding record ID information in the URL parameter called `id`, then here's how your Filter area should look:



For more information, see “Building the detail page using a filtered recordset” on page 616.

- 5 Click OK.

When the user selects a record on the results page, the update page will generate a recordset containing only the selected record.

Completing the update page rapidly

You can add the final two building blocks of an update page in a single operation using the Record Update Form live object. The live object adds both an HTML form and an Update Record server behavior to the page.

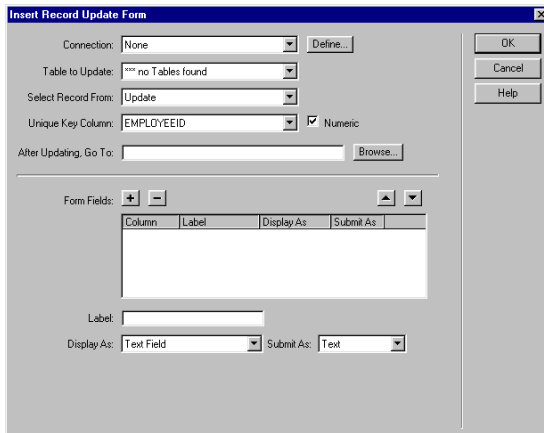
Before you can use the live object, your web application must be able to identify the record to update, and your update page must be able to retrieve it. (See “Identifying the record to update” on page 622 and “Retrieving the record to update” on page 623.)

You can also add the HTML form and the Update Record server behavior separately using the form tools and the Server Behaviors panel. (See “Completing the update page block by block” on page 624.)

After the live object places the building blocks on the page, you can use the Dreamweaver design tools to customize the form to your liking, or the Server Behaviors panel to edit the Update Record server behavior.

To build the update page with the Record Update Form live object:

- 1 Open the page in Design view, then choose Insert > Application Objects > Record Update Form. The Record Update Form dialog box appears.



- 2 Complete the dialog box.

For instructions, click the Help button in the dialog box.

- 3 Click OK.

The live object adds both an HTML form and an Update Record server behavior to your page. The form objects are laid out in a basic table, which you can customize using the Dreamweaver page design tools. (Make sure all the form objects remain within the form's boundaries.)

To edit the server behavior, open the Server Behaviors panel (Window > Server Behaviors) and double-click the Update Record behavior.

Completing the update page block by block

You can add the final two basic building blocks of an update page separately using the form tools and the Server Behaviors panel.

Before you can add the building blocks, your web application must be able to identify the record to update, and your update page must be able to retrieve it. (See “Identifying the record to update” on page 622 and “Retrieving the record to update” on page 623.)

You can also add the remaining building blocks all at once using the Record Update Form live object. (See “Completing the update page rapidly” on page 623.)

The first step is to add an HTML form to the page to let users modify the data.

To add an HTML form to an update page:

- 1 Create a new page (File > New).
- 2 Lay out your page using the Dreamweaver design tools.
- 3 Add an HTML form by placing the insertion point where you want the form to appear and choosing Form from the Insert menu.

An empty form is created on the page. You may have to turn on Invisible Elements (View > Visual Aids > Invisible Elements) to see the form's boundaries, which are represented by thin red lines.

- 4 Name the HTML form by clicking the `<form>` tag at the bottom of the Document window to select the form, opening the Property inspector (Window > Properties), and entering a name in the Form Name box.

You don't have to specify an `action` or `method` attribute for the form to tell it where and how to send the record data when the user clicks the Submit button. The Update Record server behavior sets these attributes for you.

- 5 Add a form object (Insert > Form Objects) for each column you want to update in the database table.

The form objects are for data entry. Text fields are common for this purpose, but you can also use list/menus, checkboxes, and radio buttons.

Each form object should have a corresponding column in the recordset you defined earlier. The only exception is the unique key column, which should have no corresponding form object.

For more information on form objects, see “Creating Interactive Forms” on page 573.

- 6 Add a Submit button to the form (Insert > Form Objects > Button).
- 7 If you wish, change the label of the Submit button by selecting the button, opening the Property inspector (Window > Properties), and entering a new value in the Label box.

For example, here's the Property inspector of a button labeled “Update Record”:



The next step is to display the record in the form by binding the form objects to database table columns.

To display the record in the form:

- 1 Make sure you defined a recordset to hold the record the user wants to update.
For more information, see “Retrieving the record to update” on page 623.
- 2 Drag a database table column from the Bindings panel (Window > Bindings) to its corresponding form object on the page.

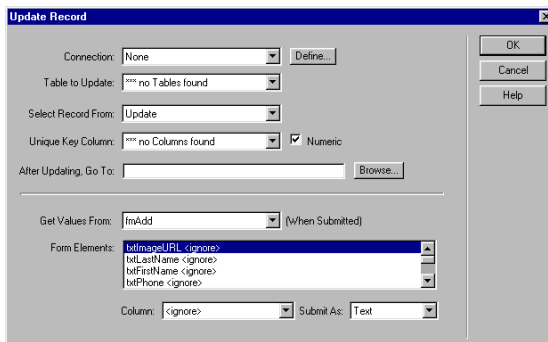
For more information, see “Making ActiveX, Flash, and other object parameters dynamic” on page 525.

The final step is to add the Update Record server behavior to update the database table after the user modifies the record.

To add a server behavior to update the database table:

- 1 In the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button and choose Update Record from the pop-up menu.

The Update Record dialog box appears.



- 2 Complete the dialog box.

For instructions, click the Help button in the dialog box.

- 3 Click OK.

Dreamweaver adds a server behavior to the page that lets users update records in a database by modifying the information displayed in the HTML form and clicking the Submit button.

Building a page to delete a record

Your application can contain a page that lets users delete existing records in a database table. A delete page is usually a detail page working in tandem with a results page. The results page lets the user choose a record to delete, then passes the choice to the delete page.

A delete page has four building blocks:

- A filtered recordset to retrieve the record from a database table
- A read-only display of the data about to be deleted
- A Submit button to send the delete command to the server
- A Delete Record server behavior to update the database table

Note: The delete page can contain only one record-editing server behavior at a time. For example, you cannot add an Insert Record or an Update Record server behavior to the delete page.

Identifying the record to delete

When users want to delete a record, they must first find that record in the database. Accordingly, you need a search and a results page to work with the delete page. For instructions on creating a search and a results page, see “Building Pages that Search Databases” on page 607.

The results page tells the delete page which record to delete by passing it a URL parameter. Therefore, make sure the results page has a Go to Detail Page server behavior that names the delete page as the detail page. For instructions, see “Modifying a results page to work with a detail page” on page 614.

Retrieving the record to delete

After the results page passes a URL parameter to the delete page identifying the record to delete, the delete page must read the parameter, retrieve the record from the database table, and temporarily store the record in a recordset.

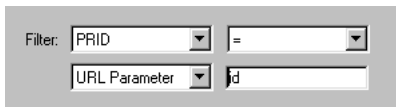
To retrieve the record to delete:

- 1 In the Bindings panel (Window > Bindings), click the plus (+) button and choose Recordset or DataSet (ASP.NET).

If the advanced dialog box appears, click the Simple button to open the simple dialog box.

- 2 Name the recordset and specify where the data you want to delete is located using the Connection and Table pop-up menus.
- 3 In the Columns area, select the All option to select all the columns in the database table.
- 4 Configure the Filter area so that the value of your key column equals the value of the corresponding URL parameter passed by the results page.

This kind of filter creates a recordset that contains only the record specified by the results page. For example, if your key column contains record ID information and is called PRID, and if the results page passes the corresponding record ID information in the URL parameter called `id`, then here's how your Filter area should look:



For more information, see “Building the detail page using a filtered recordset” on page 616.

- 5 Click OK.

When the user selects a record on the results page, the delete page generates a recordset containing only the selected record.

Next you'll add a read-only display of the data to be deleted.

Displaying the data to be deleted

It is good practice to display the record before the user deletes it to confirm that the user wants to delete it.

To add a read-only display of the record to be deleted:

- 1 Make sure you defined a recordset to hold the record the user wants to delete.

For more information, see “Retrieving the record to delete” on page 627.

- 2 Drag a column from the Bindings panel (Window > Bindings) to the page.

Dynamic content appears on the page. You can drop the dynamic content on the page as is, or you can drop it in an HTML table. For more information, see “Making text dynamic” on page 520.

Sending the delete command to the server

The delete page uses a Submit button to send the delete command to the server. To add a Submit button to your page, you must create an HTML form. The form can consist of only the Submit button.

To add a Submit button to a delete page:

- 1 In Design view, place the insertion point where you want the Submit button to appear and choose Form from the Insert menu.

An empty form is created on the page. You may have to turn on Invisible Elements (View > Visual Aids > Invisible Elements) to see the form's boundaries, which are represented by thin red lines.

- 2 Name the HTML form by clicking the `<form>` tag at the bottom of the Document window to select the form, opening the Property inspector (Window > Properties), and entering a name in the Form Name box.

You don't have to specify an `action` or `method` attribute for the form to tell it where and how to send the record data when the user clicks the Submit button. The Delete Record server behavior sets these attributes for you.

- 3 Add a Submit button to the form (Insert > Form Objects > Button).
- 4 If you wish, change the label of the Submit button by selecting the button, opening the Property inspector (Window > Properties), and entering a new value in the Label box.

For example, here's the Property inspector of a button labeled "Delete Record":



Next, you add the Delete Record server behavior to update the database table after the user clicks the Submit button.

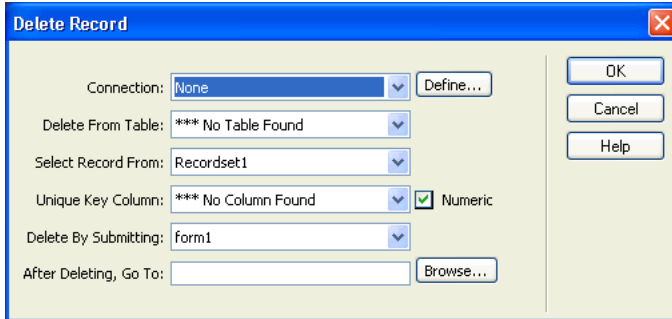
Deleting the record from the database table

The final step is to add the Delete Record server behavior to update the database table after the user clicks the Submit button.

To add a server behavior to delete the database table:

- 1 In the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button and choose Delete Record from the pop-up menu.

The Delete Record dialog box appears.



- 2 Complete the dialog box.

For more information, click the Help button in the dialog box.

- 3 Click OK.

Dreamweaver adds a server behavior to the page that lets users delete records in a database table by clicking the Submit button on the form.

Using stored procedures to modify databases

You can use a stored procedure to modify a database. A stored procedure is a reusable database item that performs some operation on the database. A stored procedure contains SQL code that can, among other things, insert, update, or delete records. Stored procedures can also alter the structure of the database itself. For example, you can use a stored procedure to add a table column or even delete a table.

A stored procedure can also call another stored procedure, as well as accept input parameters and return multiple values to the calling procedure in the form of output parameters.

A stored procedure is reusable in the sense that you can reuse a single compiled version of the procedure to execute a database operation a number of times. If you know a database task will be executed more than a few times—or the same task will be executed by different applications—using a stored procedure to execute that task can make database operations more efficient.

Note: MySQL and Microsoft Access databases do not support stored procedures.

Before you use a stored procedure to modify a database, make sure the stored procedure contains SQL that modifies the database in some way. To create and store one in your database, consult your database documentation and a good Transact-SQL manual.

The procedure for using a stored procedure varies depending on your server model.

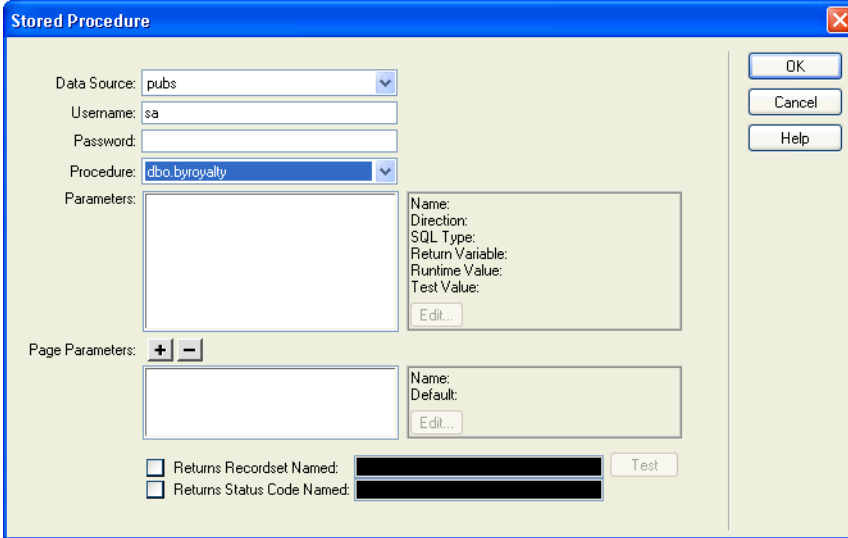
Running the stored procedure in ColdFusion

With Macromedia ColdFusion pages in Dreamweaver, you must add a Stored Procedure server behavior to a page to run a stored procedure.

To add a stored procedure to a ColdFusion page:

- 1 In Dreamweaver, open the page that will run the stored procedure.
- 2 In the Bindings panel, click the plus (+) button, and then select Stored Procedure.

The Stored Procedure dialog box appears.



The screenshot shows the "Stored Procedure" dialog box with the following fields and controls:

- Data Source:** A dropdown menu set to "pubs".
- Username:** A text field containing "sa".
- Password:** An empty text field.
- Procedure:** A dropdown menu set to "dbo.byroyalty".
- Parameters:** An empty list box.
- Page Parameters:** A section with a "+" and "-" button and an empty list box.
- Properties:** Two sections, each with "Name:" and "Default:" labels and an "Edit..." button.
- Options:** Two checkboxes: "Returns Recordset Named:" and "Returns Status Code Named:", each followed by a blacked-out text field.
- Buttons:** "OK", "Cancel", and "Help" buttons on the right side.
- Test:** A "Test" button located at the bottom right of the dialog.

- 3 Complete the dialog box.

The dialog boxes for ColdFusion 4 and ColdFusion MX are different.

For more information, click the Help button in the dialog box.

- 4 Click OK.

After you close the Stored Procedure dialog box, Dreamweaver inserts ColdFusion code in your page that, when run on the server, calls a stored procedure in the database. The stored procedure in turn performs a database operation, such as inserting a record.

If the stored procedure takes parameters, you can create a page that gathers the parameter values and submits them to the page with the stored procedure. For example, you may create a page that uses URL parameters or an HTML form to gather parameter values from users.

Running the stored procedure in ASP

With ASP pages in Dreamweaver, you must add a command object to a page to run a stored procedure. For more information on command objects, see “Understanding ASP command objects” on page 634.

To add a stored procedure to an ASP page:

- 1 In Dreamweaver, open the page that will run the stored procedure.
- 2 In the Bindings panel, click the plus (+) button, and then select Command (Stored Procedure).
The Command dialog box appears.
- 3 Enter a name for the command, choose a connection to the database containing the stored procedure, then choose Stored Procedure from the Type pop-up menu.
- 4 Choose your stored procedure by expanding the Stored Procedures branch in the Database Items box, choosing the stored procedure from the list, and clicking the Procedure button.
- 5 Enter any required parameters in the Variables table.
You don't need to enter any parameters for any `RETURN_VALUE` variable.
- 6 Click OK.

After you close the dialog box, Dreamweaver inserts ASP code in your page that, when run on the server, creates a command object that runs a stored procedure in the database. The stored procedure in turn performs a database operation, such as inserting a record.

By default, the code sets the Prepared property of the Command object to `true`, which makes the application server reuse a single compiled version of the object every time the stored procedure is run. If you know the command will be executed more than a few times, having a single compiled version of the object can improve the efficiency of database operations. However, if the command will only be executed one or two times, using one might actually slow down your web application because the system has to pause to compile the command. To change the setting, switch to Code view and change the Prepared property to `false`.

Note: Not all database providers support prepared commands. If your database does not support it, you might get an error message when you run the page. Switch to Code view and change the Prepared property to `false`.

If the stored procedure takes parameters, you might create a page that gathers the parameter values and submits them to the page with the stored procedure. For example, you may create a page that uses URL parameters or an HTML form to gather parameter values from users.

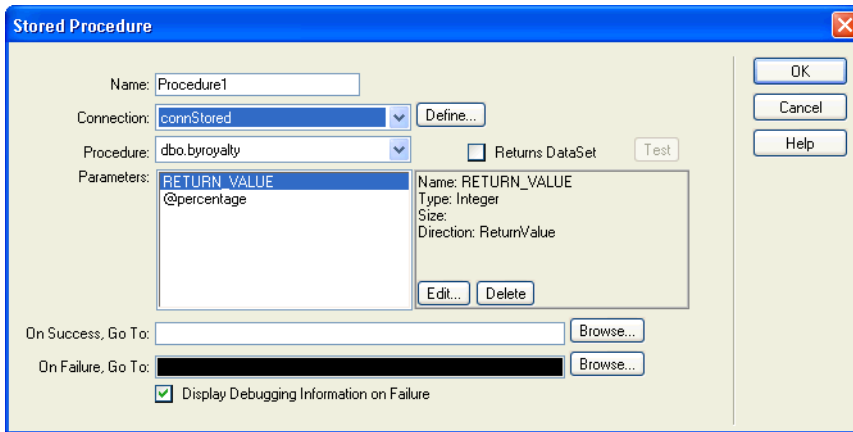
Running the stored procedure in ASP.NET

With ASP.NET pages in Dreamweaver, you must add a Stored Procedure server behavior to a page to execute a stored procedure.

To add a stored procedure to a ASP.NET page:

- 1 In Dreamweaver, open the page that will run the stored procedure.
- 2 In the Bindings panel, click the plus (+) button, and then select Stored Procedure.

The Stored Procedure dialog box appears.



- 3 Complete the dialog box.

For more information, click the Help button in the dialog box.

- 4 Click OK.

After you close the dialog box, Dreamweaver inserts ASP.NET code in your page that, when run on the server, executes a stored procedure in the database. The stored procedure in turn performs a database operation, such as inserting a record or executing a query.

If the stored procedure takes parameters, you might create a page that gathers the parameter values and submits them to the page with the stored procedure. For example, you may create a page that uses URL parameters or an HTML form to gather parameter values from users.

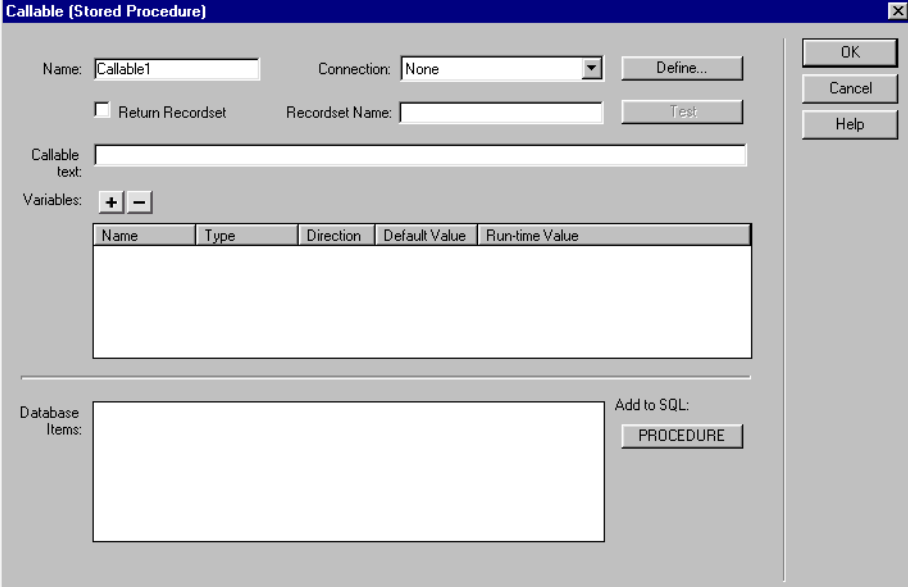
Running the stored procedure in JSP

With JSP pages in Dreamweaver, you must add a Callable server behavior to a page to run a stored procedure.

To add a stored procedure to a JSP page:

- 1 In Dreamweaver, open the page that will run the stored procedure.
- 2 In the Bindings panel, click the plus (+) button, and choose Callable (Stored Procedure).

The Callable (Stored Procedure) dialog box appears.



The screenshot shows the 'Callable (Stored Procedure)' dialog box. It has a title bar with the text 'Callable (Stored Procedure)' and a close button. The dialog is divided into several sections:

- Name:** A text box containing 'Callable1'.
- Connection:** A dropdown menu set to 'None' with a 'Define...' button to its right.
- Return Recordset:** A checkbox that is currently unchecked.
- Recordset Name:** An empty text box with a 'Test' button to its right.
- Callable text:** A large empty text area.
- Variables:** A section with '+' and '-' buttons above a table.
- Table:** A table with five columns: 'Name', 'Type', 'Direction', 'Default Value', and 'Run-time Value'. The table is currently empty.
- Database Items:** A large empty text area.
- Add to SQL:** A button labeled 'PROCEDURE'.
- Buttons:** On the right side, there are three buttons: 'OK', 'Cancel', and 'Help'.

- 3 Complete the dialog box.

For instructions, click the Help button in the dialog box.

- 4 Click OK.

After you close the Callable (Stored Procedure) dialog box, Dreamweaver inserts JSP code in your page that, when run on the server, calls a stored procedure in the database. The stored procedure in turn performs a database operation, such as inserting a record.

If the stored procedure takes parameters, you can create a page that gathers the parameter values and submits them to the page with the stored procedure. For example, you may create a page that uses URL parameters or an HTML form to gather parameter values from users.

Using ASP commands to modify a database

You can use Dreamweaver to create ASP command objects that insert, update, or delete records in a database. You supply the command object with the SQL statement that performs the operation on the database.

You can also supply the object with a stored procedure that performs the operation. For more information, see “Running the stored procedure in ASP” on page 631.

Understanding ASP command objects

A command object is a server object that performs some operation on a database. The object can contain any valid SQL statement, including one that returns a recordset, or one that inserts, updates, or deletes records in a database. A command object can alter the structure of a database if the SQL statement adds or deletes a column in a table. You can also use a command object to run a stored procedure in a database.

A command object can be reusable, in the sense that the application server can reuse a single compiled version of the object to execute the command a number of times. You make a command reusable by setting the Prepared property of the Command object to `true`, as in the following VBScript statement:

```
mycommand.Prepared = true
```

If you know the command will be executed more than a few times, having a single compiled version of the object can make database operations more efficient.

Note: Not all database providers support prepared commands. If your database does not support it, it might return an error when you set this property to `true`. It might even ignore the request to prepare the command and set the Prepared property to `false`.

A command object is created by scripts on an ASP page, but Dreamweaver lets you create command objects without writing a line of ASP code. The rest of this section describes how to use rapid application development (RAD) tools in Dreamweaver to create ASP command objects that edit database records.

Creating a command that uses SQL to edit a database

You can use an ASP command object that uses a SQL statement to insert, update, or delete records in a database.

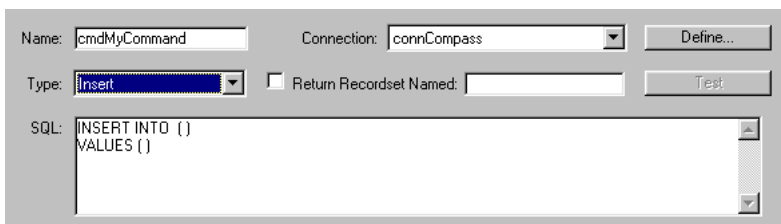
To create the command object:

- 1 In Dreamweaver, open the ASP page that will run the command.
- 2 Open the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button, and choose Command.

The Command dialog box appears.

- 3 Enter a name for the command, choose a connection to the database containing the records you want to edit, and choose the editing operation you want the command to perform—Insert, Update, or Delete.

Dreamweaver starts the SQL statement for you based on the type of operation you choose. For example, here's the dialog box after choosing Insert:



Name: Connection:

Type: Return Recordset Named:

SQL:

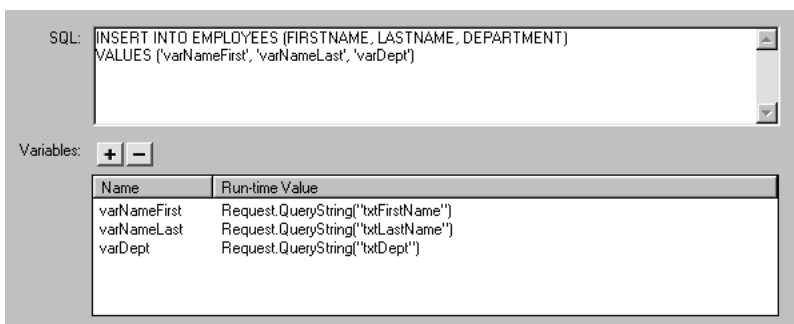
```
INSERT INTO ()
VALUES ()
```

- 4 Complete the SQL statement.

For information on writing SQL statements that modify databases, consult a Transact-SQL manual.

- 5 Use the Variables area to define any SQL variables.

For example, below is an Insert statement that contains three SQL variables. The values of these variables are provided by URL parameters passed to the page, as defined in the Run-time Value column of the Variables area.



SQL:

```
INSERT INTO EMPLOYEES (FIRSTNAME, LASTNAME, DEPARTMENT)
VALUES ('varNameFirst', 'varNameLast', 'varDept')
```

Variables:

Name	Run-time Value
varNameFirst	Request.QueryString("txtFirstName")
varNameLast	Request.QueryString("txtLastName")
varDept	Request.QueryString("txtDept")

After you close the dialog box, Dreamweaver inserts ASP code in your page that, when run on the server, creates a command that inserts, updates, or deletes records in the database.

By default, the code sets the Prepared property of the Command object to `true`, which makes the application server reuse a single compiled version of the object every time the command is run. To change this setting, switch to Code view and change the Prepared property to `false`.

In the above example, next you would probably create a page with an HTML form so users could enter record data. The HTML form would contain three text fields (`txtFirstName`, `txtLastName`, and `txtDept`) and a submit button. As well, the form would use the GET method and submit the text field values to the page containing your command.

Using JSP prepared statements to modify a database

You can use Dreamweaver to create JSP prepared statements that insert, update, or delete records in a database. You supply the prepared statement with the SQL that performs the operation on the database.

Understanding JSP prepared statements

A JSP prepared statement is a reusable server object that contains a SQL statement. You can place any valid SQL statement in a prepared statement. For example, a prepared statement can contain a SQL statement that returns a recordset, or one that inserts, updates, or deletes records in a database.

A prepared statement is reusable in the sense that the application server uses a single instance of the prepared statement object to query the database a number of times. Unlike the JSP statement object, a new instance of the prepared statement object is not created for each new database query. If you know the statement will be executed more than a few times, having a single instance of the object can make database operations more efficient and take up less server memory.

A prepared statement object is created by a Java scriptlet on a JSP page. However, Dreamweaver lets you create prepared statements without writing a single line of Java code.

If you're interested in the code, the following scriptlet creates a prepared statement:

```
String myquery = "SELECT * FROM EMPLOYEES WHERE DEPARTMENT = ?";
PreparedStatement mystatement = connection.prepareStatement(myquery);
```

The first line stores the SQL statement in a string variable called `myquery`, with a question mark serving as a placeholder for the SQL variable value. The second line creates a prepared statement object called `mystatement`.

Next, you assign a value to the SQL variable, as follows:

```
mystatement.setString(1, request.getParameter("myURLparam"));
```

The `setString` method assigns the value to the variable and takes two arguments. The first argument specifies the affected variable by its position (here, the first position in the SQL statement). The second argument specifies the variable's value. In this example, the value is provided by a URL parameter passed to the page.

Note: You must use different methods to assign non-string values to SQL variables. For example, to assign an integer to the variable, you would use the `mystatement.setInt()` method.

Finally, you generate the recordset, as follows:

```
ResultSet myresults = mystatement.execute();
```

The rest of this section describes how to create JSP prepared statements using rapid application development (RAD) tools in Dreamweaver. These tools let you create prepared statements without writing a line of JSP code.

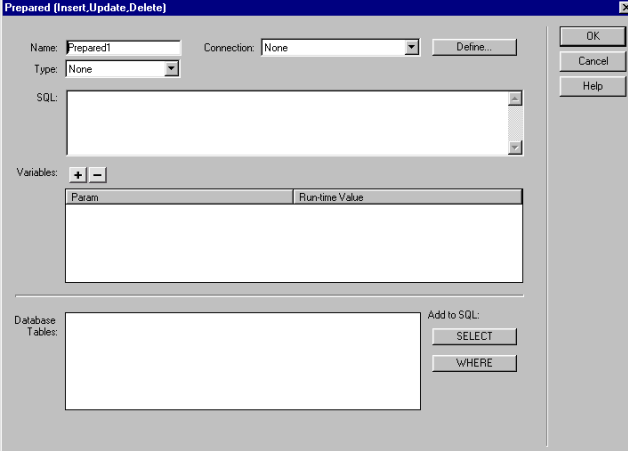
Creating a prepared statement that edits a database record

You can use a JSP prepared statement to insert, update, or delete records in a database.

To create the prepared statement:

- 1 In Dreamweaver, open the JSP page that will run the command.
- 2 Open the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button, and choose Prepared (Insert, Update, Delete).

The Prepared (Insert, Update, Delete) dialog box appears:



The screenshot shows the 'Prepared (Insert, Update, Delete)' dialog box. It has a title bar with the text 'Prepared (Insert, Update, Delete)'. The dialog contains the following fields and controls:

- Name:** A text field containing 'Prepared1'.
- Connection:** A dropdown menu set to 'None'.
- Type:** A dropdown menu set to 'None'.
- SQL:** A large text area for entering the SQL statement.
- Variables:** A section with a '+' and '-' button, and a table with two columns: 'Param' and 'Run-time Value'.
- Database Tables:** A text area for listing database tables.
- Add to SQL:** Two buttons labeled 'SELECT' and 'WHERE'.
- Buttons:** 'OK', 'Cancel', and 'Help' buttons on the right side.

- 3 Complete the dialog box.

For instructions, click the Help button in the dialog box.

- 4 Click OK.

After you close the dialog box, Dreamweaver inserts JSP code in your page that, when run on the server, creates a prepared statement that inserts, updates, or deletes records in the database.

CHAPTER 42

Building Pages that Restrict Access to Your Site

You can use Macromedia Dreamweaver MX to build the following pages to restrict access to your site:

- A page that requires users to register the first time they visit the site (see “Building a registration page” on page 639)
- A page that lets registered users log in to the site (see “Building a login page” on page 643)
- Pages that only authorized users can view (see “Building a page only authorized users can access” on page 645)

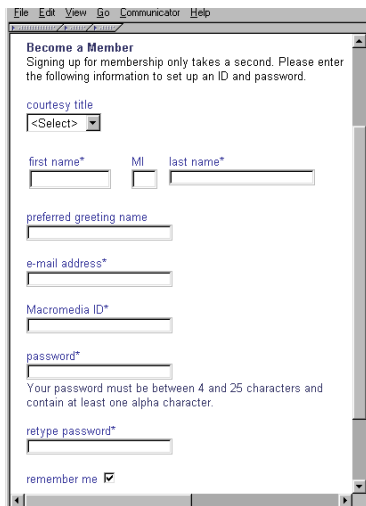
Note: Dreamweaver does not have authentication server behaviors for ASP.NET or PHP pages.

Building a registration page

Your web application can contain a page that requires users to register the first time they visit your site.

Note: Dreamweaver does not have authentication server behaviors for ASP.NET or PHP pages.

For example, the following page asks first-time users to register:



The screenshot shows a web browser window with a registration form. The form is titled "Become a Member" and includes the following fields and instructions:

- Sign up for membership only takes a second. Please enter the following information to set up an ID and password.**
- courtesy title**: A dropdown menu with "<Select>" as the current selection.
- first name***: A text input field.
- MI**: A small text input field for the middle initial.
- last name***: A text input field.
- preferred greeting name**: A text input field.
- e-mail address***: A text input field.
- Macromedia ID***: A text input field.
- password***: A text input field.
- Your password must be between 4 and 25 characters and contain at least one alpha character.**
- retype password***: A text input field.
- remember me**: A checked checkbox.

A registration page is made up of the following building blocks:

- A database table to store login information about the users.
- An HTML form that lets users choose a user name and password.
You can also use the form to obtain other personal information from users.
- An Insert Record server behavior to update the database table of site users.
- A Check New Username server behavior to make sure the user name entered by the user is not taken by another user.

Note: You can delete or change the properties of any server behavior you add to a page (see “Editing server behaviors” on page 571).

Storing login information about users

A registration page requires a database table to store the login information entered by users. Make sure your database table contains a user name and a password column. If you want logged-in users to have different access privileges, include an access privilege column (see “Storing access privileges in the user database” on page 647).

If you want to set a common password for all users of the site, configure your database application (Microsoft Access, Microsoft SQL Server, Oracle, and so on) to enter the password in each new user record by default. In most database applications, you can set a column to a default value each time a new record is created. Set the default value to the password.

You can also use the database table to store other useful information about the user.

Letting users choose a user name and password

You add an HTML form to the registration page to let users choose a user name and password (if applicable).

To let users choose a user name and password:

- 1 Create a new page (File > New) and lay out your registration page using the Dreamweaver design tools.
- 2 Add an HTML form by placing the insertion point where you want the form to appear and choosing Form from the Insert menu.

An empty form is created on the page. You may have to turn on Invisible Elements (View > Visual Aids > Invisible Elements) to see the form’s boundaries, which are represented by thin red lines.

- 3 Name the HTML form by clicking the `<form>` tag at the bottom of the Document window to select the form, opening the Property inspector (Window > Properties), and entering a name in the Form Name box.

You don’t have to specify an `action` or `method` attribute for the form to tell it where and how to send the record data when the user clicks the Submit button. The Insert Record server behavior sets these attributes for you (see “Updating the table of users in the database” on page 641).

- 4 Add text fields (Insert > Form Objects > Text Field) to let the user enter a user name and password. The form can also have more form objects to record other personal data. You should add labels (either as text or images) beside each form object to tell users what they are. You should also line up the form objects by placing them inside an HTML table. For more information on form objects, see “Creating Interactive Forms” on page 573.
- 5 Add a Submit button to the form (Insert > Form Objects > Button).
- 6 If you wish, change the label of the Submit button by selecting the button, opening the Property inspector (Window > Properties), and entering a new value in the Label box. For example, here’s the Property inspector of a button labeled “Register”:



The next step is to add the Insert Record server behavior to insert records in the table of users in the database.

Updating the table of users in the database

You add an Insert Record server behavior to update the table of users in the database.

To update the table of users in the database:

- 1 In the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button and choose Insert Record from the pop-up menu. The Insert Record dialog box appears.
- 2 Use the Connection and Insert Into Table pop-up menus to specify the table of users in the database.
- 3 In the After Inserting, Go To dialog box, enter the page to open after the record is inserted into the table.
- 4 In the Get Values From pop-up menu, choose the HTML form used to obtain the user’s user name and password. Dreamweaver automatically chooses the first form on your page.
- 5 Specify what each object on your form will update in the database table by selecting a form object in the Form Elements list, then choosing a table column from the Column pop-up menu and a data type from the Submit As pop-up menu.

The data type is the kind of data the column in your database table is expecting (text, numeric, Boolean checkbox values). Password or user name columns usually expect text.

For example, in the Form Elements list, click the password text field, choose the column in the database table where the password should be stored, and then choose the Text data type.

Repeat the procedure for each form object in the Form Elements list.

- 6 Click OK.

The final step is to make sure the user name is not used by another registered user.

Making sure the chosen user name is unique

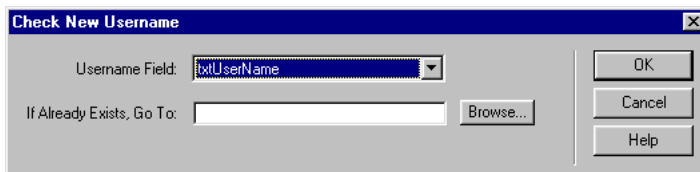
You add a server behavior to make sure the user name entered is not taken by another registered user.

When the user clicks the Submit button on the registration page, the server behavior compares the user name entered by the user against the user names stored in a database table of registered users. If no matching user name is found in the database table, the server behavior carries out the insert record operation normally. If a matching user name is found, the server behavior cancels the insert record operation and opens a new page (usually a page alerting the user that the user name is already taken).

To make sure the chosen user name is unique:

- 1 In the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button and choose User Authentication > Check New Username from the pop-up menu.

The Check New Username dialog box appears.



- 2 Complete the dialog box.

For instructions, click the Help button in the dialog box.

- 3 Click OK.

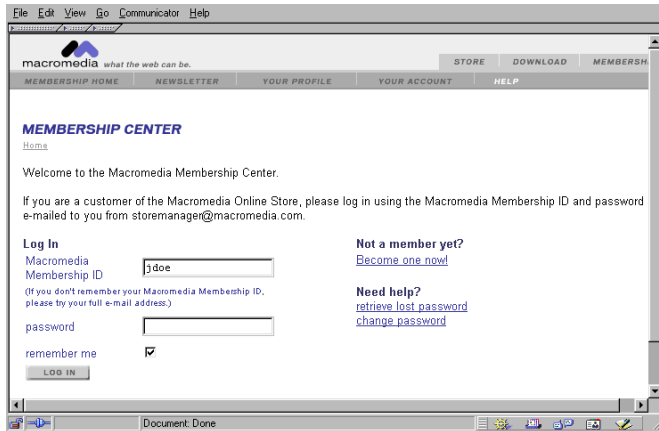
Dreamweaver adds a server behavior to the registration page that checks that the user name submitted by a visitor is unique before adding that visitor's information to the database of registered users.

Building a login page

Your web application can contain a page that lets registered users log in to the site.

Note: Dreamweaver does not have authentication server behaviors for ASP.NET or PHP pages.

For example, the following page asks registered users to log in:



A login page is made up of the following building blocks:

- A database table of registered users
- An HTML form to let users enter a user name and password
- A Log In User server behavior to make sure the entered user name and password are valid

A session variable consisting of the user name is created for the user when the user logs in successfully.

Note: You can delete or change the properties of any server behavior you add to a page (see “Editing server behaviors” on page 571).

Creating a database table of registered users

You need a database table of registered users to verify that the user name and password entered in the login page are valid. Use your database application and a registration page to create the table. For more information, see “Building a registration page” on page 639.

Letting users log in

You add an HTML form to the page to let users log in by entering a user name and password.

To let users log in:

- 1 Create a new page (File > New) and lay out your login page using the Dreamweaver design tools.
- 2 Add an HTML form by placing the insertion point where you want the form to appear and choosing Form from the Insert menu.

An empty form is created on the page. You may have to turn on Invisible Elements (View > Visual Aids > Invisible Elements) to see the form’s boundaries, which are represented by thin red lines.

- 3 Name the HTML form by clicking the `<form>` tag at the bottom of the Document window to select the form, opening the Property inspector (Window > Properties), and entering a name in the Form Name box.

You don't have to specify an `action` or `method` attribute for the form to tell it where and how to send the record data when the user clicks the Submit button. The Log In User server behavior sets these attributes for you (see "Checking the user name and password" on page 644).

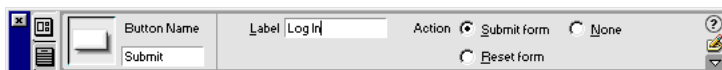
- 4 Add a user name and a password text field (Insert > Form Objects > Text Field) to the form.

Add labels (either as text or images) beside each text field, and line up the text fields by placing them inside an HTML table and setting the table's `border` attribute to 0.

- 5 Add a Submit button to the form (Insert > Form Objects > Button).

- 6 If you wish, change the label of the Submit button by selecting the button, opening the Property inspector (Window > Properties), and entering a new value in the Label box.

For example, here's the Property inspector of a button labeled "Log In":



The next step is to add the Log In User server behavior to make sure the entered user name and password are valid.

Checking the user name and password

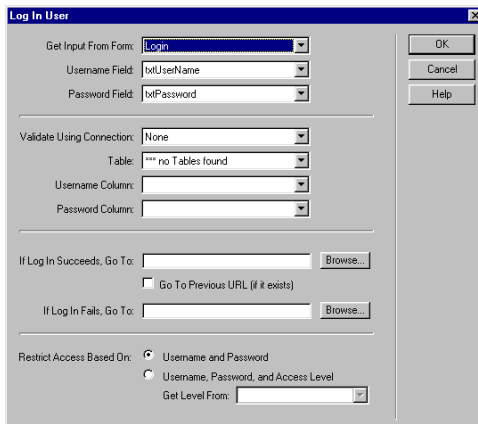
You add a Log In User server behavior to make sure the user name and password users enter are valid.

When a user clicks the Submit button on the login page, the Log In User server behavior compares the values entered by the user against the values for registered users. If the values match, the server behavior opens one page (usually the site's start page). If the values do not match, the server behavior opens another page (usually a page alerting the user that the login attempt failed).

To check the user name and password:

- 1 In the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button and choose User Authentication > Log In User from the pop-up menu.

The Log In User dialog box appears.



- 2 Complete the dialog box.

For instructions, click the Help button in the dialog box.

- 3 Click OK.

Dreamweaver adds a server behavior to the login page that makes sure the user name and password entered by a visitor are valid.

Building a page only authorized users can access

Your web application can contain a protected page that only authorized users can access.

Note: Dreamweaver does not have authentication server behaviors for ASP.NET or PHP pages.

For example, if a user attempts to bypass the login page by typing the protected page's URL in a browser, the user is redirected to another page. Similarly, if you set the authorization level for a page to Administrator, then only users with Administrator access privileges can view the page. If a logged-in user attempts to access the protected page without the proper access privileges, the user is redirected to another page.

You can also use authorization levels to review newly registered users before granting them full access to the site. For example, you may want to receive payment before allowing a user access to the member pages of the site. To do so, you can protect the member pages with a Member authorization level and only grant newly registered users Guest privileges. After receiving payment from the user, you can upgrade the user's access privileges to Member (in the database table of registered users).

If you do not plan to use authorization levels, you can protect any page on your site simply by adding a Restrict Access To Page server behavior to the page. The server behavior redirects to another page any user who has not successfully logged in.

If you do plan to use authorization levels, you can protect any page on your site with the following building blocks:

- An extra column in your users database table to store each user's access privileges
- A Restrict Access To Page server behavior to redirect unauthorized users to another page

In this case, the server behavior redirects to another page any user who does not have the required access privileges.

In either case, you can add a link to the protected page that lets a user log out and clears any session variables. For more information, see “Logging out users” on page 647.

Redirecting unauthorized users to another page

To prevent unauthorized users from accessing a page, you add a Restrict Access To Page server behavior to it. The server behavior redirects the user to another page if the user attempts to bypass the login page by typing the protected page's URL in a browser, or if the user is logged in but attempts to access the protected page without the proper access privileges.

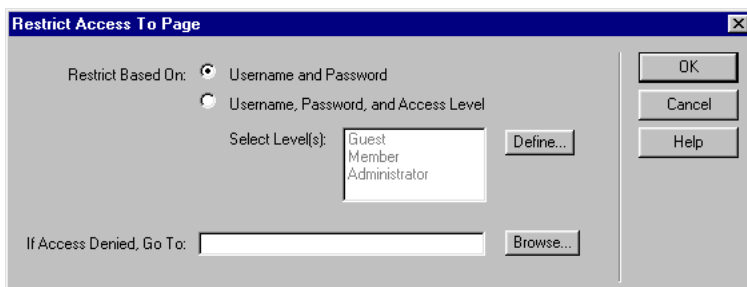
Note: The Restrict Access To Page server behavior can only protect HTML pages. It does not protect other site resources such as image files and audio files.

If you want to give many pages on your site the same access rights, you can copy and paste access rights from one page to another.

To redirect unauthorized users to another page:

- 1 Open the page you want to protect.
- 2 In the Server Behaviors panel (Window > Server Behaviors), click the plus (+) button and choose User Authentication > Restrict Access To Page from the pop-up menu.

The Restrict Access To Page dialog box appears.



- 3 Complete the dialog box.

For instructions, click the Help button in the dialog box.

- 4 Click OK.

Dreamweaver adds a server behavior to the page that allows only authorized users to view the page.

To copy and paste a page's access rights to other pages on the site:

- 1 Open the protected page and select the Restrict Access To Page server behavior listed in the Server Behaviors panel (not the one in the plus (+) pop-up menu).

- 2 Click the arrow button in the top right corner of the panel and choose Copy from the pop-up menu.

The Restrict Access To Page server behavior is copied to your system's Clipboard.

- 3 Open another page you want to protect in the same way.
- 4 In the Server Behaviors panel (Window > Server Behaviors), click the arrow button in the top right corner and choose Paste from the pop-up menu.
- 5 Repeat steps 3 and 4 for each page you want to protect.

Storing access privileges in the user database

This building block is required only if you want certain logged-in users to have different access privileges. If you simply require users to log in, then you don't need to store access privileges.

If you want certain logged-in users to have different access privileges, make sure your database table of users contains a column specifying each user's access privileges (Guest, User, Administrator, and so on). The access privileges of each user should be entered in the database by the site administrator.

In most database applications, you can set a column to a default value each time a new record is created. Set the default value to the most common access privilege on your site (for example, Guest), then manually change the exceptions (for example, changing Guest to Administrator). The user now has access to all administrator pages.

Make sure each user in the database has a single access privilege, such as Guest or Administrator, not multiple privileges like "User, Administrator". If you want to set multiple access privileges for your pages (for example, all guests and administrators can see this page), then set those privileges at the page level, not the database level. For more information, see "Redirecting unauthorized users to another page" on page 646.

Logging out users

When a user logs in successfully, a session variable is created that consists of the user name. When the user leaves your site, you can use the Log Out User server behavior to clear the session variable and redirect the user to another page (usually a "goodbye" or "thank you" page).

You can invoke the Log Out User server behavior when the user clicks a link or when a specific page loads.

To add a link to let users log out:

- 1 On the page, select text or an image to serve as the link.
- 2 In the Server Behaviors panel, click the plus (+) button and choose User Authentication > Log Out User.

The Log Out User dialog box appears.

- 3 Specify a page to open when the user clicks the link.
The page is usually a "goodbye" or "thank you" page.
- 4 Click OK.

To log out a user when a specific page loads:

- 1** Open the page that will load in Dreamweaver.

The page is usually a “goodbye” or “thank you” page.

- 2** In the Server Behaviors panel, click the plus (+) button and choose User Authentication > Log Out User.

The Log Out User dialog box appears.

- 3** Select the “Log Out When Page Loads” option.

- 4** Click OK.

Part X

Appendixes

Get more help to develop your web applications.

This part contains the following chapters:

- Appendix A, “Beginner’s Guide to Databases”
- Appendix B, “SQL Primer”
- Appendix C, “Setting Up a DSN in Windows”
- Appendix D, “Quick Reference: Macromedia ASP.NET Tags”

APPENDIX A

Beginner's Guide to Databases

This appendix is intended for Macromedia Dreamweaver MX users who have little or no experience working with databases or database connections. It explains general concepts, not specific procedures. To see how these concepts apply in practice, see the rest of this guide.

The appendix describes how to design a database but not how to create one in an application such as Microsoft Access or SQL Server. That process is described in the printed or online documentation that came with your database system.

The appendix contains the following sections:

- “About databases” on page 651
- “Database design basics” on page 652
- “Understanding database connections” on page 657

About databases

The building block of a database is the record. A record is a collection of related data treated as a single entity. For example, a hockey trading card could be called a record: it brings together the name, photograph, team, and statistics of one player. Using database terms, each of these related pieces of information is called a field: each hockey card “record” has a name field, a photograph field, a team field, and various statistic fields.

A collection of records that share the same fields is called a table because this kind of information can easily be presented in table format: each column represents a field and each row represents a record. In fact, the word column is synonymous with the word field, and the word row is synonymous with the word record.

Fields (columns)

Number	LastName	FirstName	Position	Goal


Records
(rows)

A database can contain more than one table, each with a unique name. These tables can be related or independent from one another.

A subset of data extracted from one or more tables is called a recordset (or a DataSet in ASP.NET). A recordset is also a table because it's a collection of records that share the same columns. For example, a hockey team roster listing the names and positions of the players could be called a recordset: it consists of a subset of all the possible information about the players, including goals, assists, penalty minutes, and so on.

Number	LastName	FirstName	Position	Goals

Database table



LastName	FirstName	Position

Recordset table

To create a recordset, you run a database query. A query consists of search criteria. For example, the query can specify that only certain columns be included in the recordset, or that only certain records be included. For more information, see “Defining a recordset” on page 502.

Database design basics

Database design is the first step in building any database-driven website. This section presents a case study to explain basic database design principles. The case study involves a web application commissioned by a fictional company called Arrow Aircraft Services, a firm that manages a small fleet of privately owned business jets.

This section contains the following topics:

- “Studying Arrow Aircraft’s business rules and policies” on page 652
- “Studying Arrow Aircraft’s feature request” on page 653
- “What questions will users ask the database?” on page 653
- “Choosing the tables that belong in the database” on page 654
- “Choosing the columns in each table” on page 654
- “Defining relationships between the tables” on page 656
- “Creating the database” on page 657

Studying Arrow Aircraft’s business rules and policies

You have been hired on a contract basis to build a web application for Arrow Aircraft Services. Before you start designing the database, you make sure you understand all the organization’s business rules and policies that will affect the application. This section describes the (simplified) business rules and policies of Arrow Aircraft Services.

Arrow Aircraft manages a fleet of five business jets of varying sizes and models for their owners. Each jet has up to eight fractional owners—corporations or individuals who have purchased a share of the aircraft. This type of arrangement is popular with those who don't need a business jet on a year-round basis.

The fractional owners, or “shareholders”, pay Arrow Aircraft the following fees:

- A monthly management fee proportional to the owner's share of the aircraft to cover pilot, insurance, and hangaring costs
- An “occupied” hourly fee covering all direct costs such as maintenance, engine reserves and catering

In exchange, a shareholder tells Arrow Aircraft when and where they want to go and Arrow Aircraft takes care of the rest, including obtaining flight and ground crews and catering the flight. Arrow Aircraft requires a minimum notice of 8 hours to prepare an aircraft for a flight.

Arrow Aircraft guarantees 800 “occupied hours” per year for each aircraft. Occupied hours are hours where the jet is both in flight and occupied by at least one passenger. Occupied hours begin six minutes (0.1 of an hour) before the aircraft takes off with passengers and end six minutes after the aircraft lands.

Each aircraft can have up to eight owners. For example, an individual or corporation can purchase a 1/5 interest (or a 1/5th share) in a jet. Since Arrow Aircraft guarantees 800 occupied flight hours per year, the owner of the 1/5th share is entitled to 160 occupied hours ($800 \text{ hours} / 5 = 160$).

Studying Arrow Aircraft's feature request

Arrow Aircraft hired you to build a web application with the following features:

- Allow any aircraft shareholder to request an aircraft for a flight
- Provide the company's Flight Operations Department (Flight Ops) with all the information needed to begin preparing the aircraft, including details of the shareholder's request (itinerary, departure date and time, catering needs, etc.), the shareholder's remaining occupied hours, and the availability of the aircraft for the proposed trip
- Allow Flight Ops to book the aircraft to prevent scheduling conflicts

What questions will users ask the database?

After you become familiar with the proposed features of the web application, you sit down with the database users and pose the following question: “What questions will you ask the database?”

You learn from some of the aircraft shareholders that they want to ask the database the following questions:

- How many occupied hours do I have left?
- Is my aircraft available on a certain date or dates?

After a shareholder requests an aircraft, the employees in Flight Ops will ask the database the following questions:

- Where does the shareholder want to go?
- What is the itinerary—one way, return, multi-city? Flight Ops needs this information to start planning the flight (check weather forecasts, file flight plans, and so on) and to estimate the total occupied hours.
- Does the shareholder have sufficient occupied hours remaining for the proposed itinerary?
- When does the shareholder want to leave?
- What is the shareholder's plane?
- Is the plane available for the proposed itinerary?
- How many passengers will accompany the shareholders?
- How much luggage will they bring—light (carry-on), normal (one suitcase per passenger), or heavy (more than one suitcase)?
- What are the shareholder's catering needs?
- What is the occupied hourly fee for the plane?
- Where can I contact the shareholder to confirm the flight and the fee estimate?

Choosing the tables that belong in the database

After learning the questions users will ask the database, you think about how your database should be structured to best answer their questions. The first step is to choose the tables in the database.

In a relational database, all data is represented in the rows and columns of tables. Each table describes a collection of related entities such as persons, objects, or events. Each row describes one occurrence of the entity and each column describes one property of the entity—for example, a person's last name, an object's color, or an event's date.

You choose the following five tables for the Arrow Aircraft database:

- An aircraft table describing all of the business jets managed by Arrow Aircraft
- An aircraft bookings table listing all the dates the jets are booked or otherwise unavailable to fly
- A shareholders table describing all the corporations and individuals who own shares in the jets
- An itineraries table describing all the itineraries requested by the shareholders
- A flights table describing all the legs (“flights”) of the itineraries.

Choosing the columns in each table

The next step is to choose the columns in each table. The columns describe the properties of each entity in the table.

A good rule of thumb when choosing columns is to avoid duplicating data. For example, in the Arrow Aircraft application, you know shareholder names will have to be used in association with the itinerary data to answer the following Flight Ops question: “What are the shareholder's catering needs for a particular itinerary?”

One approach is to have shareholder first and last name columns in both the shareholders table and the itineraries table. However, this would require not only duplicating the name data in two tables, but also several times in the itineraries table (you would have to enter the shareholder's name in the itineraries table every time the shareholder requested a new itinerary). This approach increases data-entry effort, introduces opportunities for errors, and threatens the data's integrity (a change in one part of the database must be done in other parts).

A much better solution is to enter the shareholder's first and last name a single time in a single table, the shareholders table. When the time comes to answer the Flight Ops question above, you can join the shareholders table with the itineraries table using a SQL expression.

Make sure each of your tables has a primary key column. A primary key column contains values that are unique for each row. This allows you to zero in on the exact row you want when searching the database. Most primary key columns consist of ID numbers, but you can use natural primary keys such as government form numbers or aircraft serial numbers.

After some thought, you choose the following properties and primary keys for the tables in the Arrow Aircraft database:

aircraft table

ac_serial (primary key)
model
capacity (passenger capacity)
max_range
occupied_hr_fee (occupied hourly fee)
photo

aircraft bookings table

ac_booking_ID (primary key)
ac_booking_start
ac_booking_end

shareholders table

shareholder_ID (primary key)
fname (first name)
lname (last name)
cname (company name)
user_ID
pwd (password)
share (ownership share of aircraft in %)
occupied_hrs (occupied hours remaining)

itineraries table

itin_ID (primary key)
passengers (no. of passengers on the trip)
luggage (amount of luggage)
catering (catering needs)

flights table

flight_ID (primary key)
destination
leg_no (leg number in the itinerary)
dep_date (requested departure date)
dep_time (requested departure time)
to_time (takeoff time)
td_time (touchdown time)

Defining relationships between the tables

After defining the basic columns and primary keys in your tables, you can start defining relationships between the tables. Once the relationships are defined, you can write SQL statements in Dreamweaver MX to combine data from two tables (see “Joining tables” on page 669).

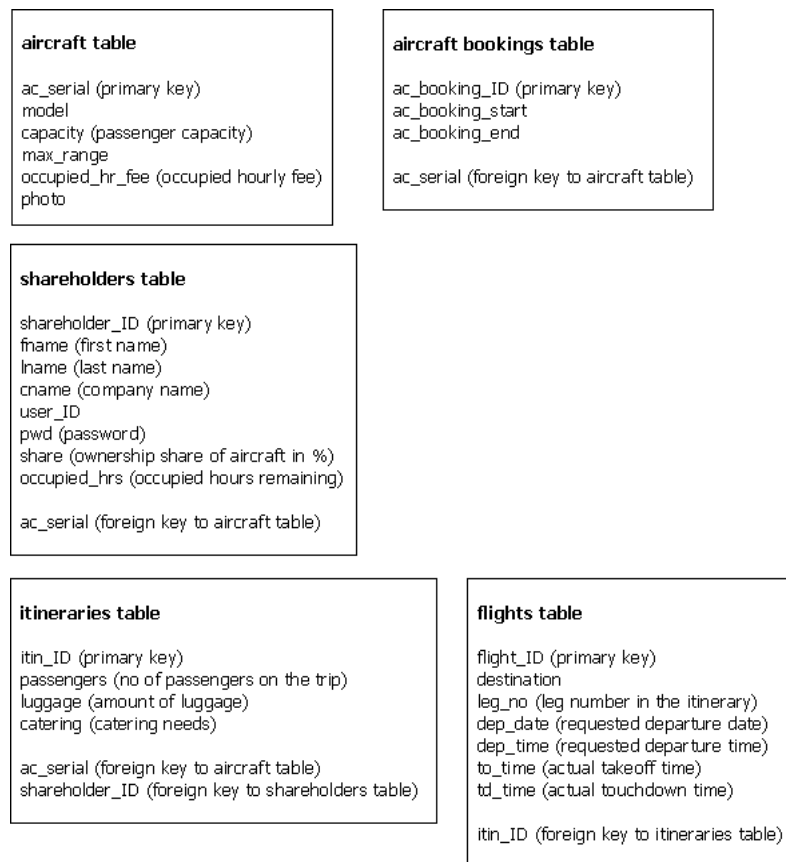
For example, each plane managed by Arrow Aircraft has several shareholders. You would like to establish a similar “one-to-many” relationship between each plane in your aircraft table and the shareholders in the shareholders table. This would save you from entering and tracking redundant aircraft data in the shareholders table.

In a one-to-many database relationship, a single row in one table is related to several rows in the another table. You can define this kind of relationship by including a foreign key in the table providing the many rows—in the above example, the shareholders table. A foreign key is a column containing values matching those in the primary key column of another table. Your aircraft table’s primary key is called `ac_serial`. Therefore, including a foreign key called `ac_serial` in the shareholders table would define a “one-aircraft-to-many-shareholders” relationship.

With your knowledge of the client’s feature request and of the company’s business rules and policies, you decide to define the following one-to-many relationships in your database:

- Each aircraft can have many shareholders
- Each aircraft has many bookings
- Each aircraft has many itineraries
- Each shareholder has many itineraries
- Each itinerary can have many flights (or legs)

Here are the revised table diagrams after you add the foreign keys defining these relationships:



The diagrams are known as E-R diagrams, or entity-relationship diagrams.

Creating the database

The final design step is creating the database using a database system like Microsoft Access, SQL Server, Oracle9i, or MySQL. Consult your database system's documentation for more information.

Understanding database connections

If you plan to use a database with your web application, you need to create at least one database connection. Without one, the application won't know where to find the database or how to connect to it. You create a database connection in Dreamweaver by providing the information—or the “parameters”—the application needs to establish contact with the database.

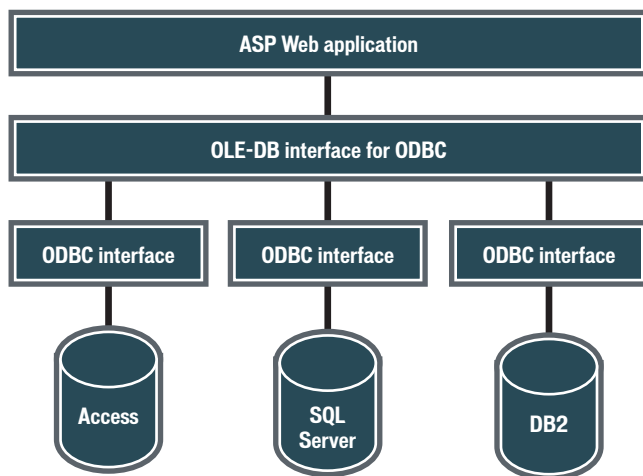
Note: You don't need to create database connections for ColdFusion pages in Dreamweaver MX. You connect using ColdFusion data sources defined in ColdFusion Administrator.

This section discusses database connections in general terms. For specific instructions on creating connections in Dreamweaver MX, see “Connecting to a database” on page 135.

The job of these interfaces is to act like interpreters. For example, when a speech is given in English at the United Nations, one interpreter translates for French-speaking delegates while another interpreter translates for German-speaking delegates. Similarly, you use one interface for OLE DB-speaking applications, another interface for ODBC-speaking web applications, and still another interface for JDBC-speaking applications. ColdFusion MX and JSP applications are JDBC speakers, ASP and ASP.NET applications speak OLE DB, and ColdFusion 4 or 5 applications speak ODBC and OLE DB.

ASP applications (but not ASP.NET applications) are fluent ODBC speakers thanks to a built-in OLE DB/ODBC interpreter. For example, suppose you want your application to communicate with a Microsoft Access database by using an ODBC interface. In ASP, if you specify only the ODBC interface and no OLE DB interface, by default the application will use an OLE DB/ODBC interpreter to translate the OLE DB into ODBC, then it will use the ODBC/Access interpreter you specified to translate the ODBC into something Access can understand.

The following illustration gives you an idea of the process:



Note: SQL Server and DB2 are server-based database systems from Microsoft and IBM, respectively.

Using database drivers to interface with your database

The ODBC, OLE DB, and JDBC interfaces are implemented by database drivers (or “data providers” in OLE DB), which are simply pieces of software. When your web application communicates with your database, it does so through the intermediary of a driver.

Database drivers are database-specific. For example, you can use Microsoft Access, SQL Server, and dBase drivers. Similarly, you can use OLE DB providers such as the OLE DB provider for SQL Server. Your choice depends on your database.

Drivers are written by database vendors such as Microsoft and Oracle, and by a variety of third-party software vendors. Microsoft offers a number of ODBC drivers and OLE DB providers for the most popular database packages, such as Microsoft Access, Microsoft SQL Server, and Oracle. The ODBC drivers, which only run on the Windows platform, are automatically installed with Microsoft Office and with Windows 2000. They are also installed with the Microsoft Data Access Components (MDAC) 2.5, 2.6, and 2.7 packages, which you can download for free from the Microsoft website at <http://www.microsoft.com/data/download.htm>. MDAC 2.6 installs a number of OLE DB providers.

Note: Install MDAC 2.5 first, then install MDAC 2.6.

To find out which ODBC drivers are installed on your Windows system, see “Viewing the ODBC drivers installed on a Windows system” on page 660.

Because the Macintosh is rarely used as a database platform, few ODBC drivers exist for it.

Some common JDBC drivers include the i-net JDBC drivers for Microsoft SQL Server databases, the Oracle Thin driver for Oracle databases, and the JDBC Driver for DB2 for IBM DB2 databases. For more information on JDBC drivers and their vendors, see the searchable database of JDBC drivers on the Sun website at <http://industry.java.sun.com/products/jdbc/drivers>.

Here are the database interfaces for each type of web application and some common database drivers for each:

Web application	Database interface	Common drivers
ColdFusion MX JSP	JDBC	Sun JDBC-ODBC driver i-net Sprinta JDBC driver for SQL Server Oracle Thin JDBC driver
ColdFusion 4 or 5	ODBC or OLE DB	ColdFusion native drivers Microsoft Access Driver Microsoft SQL Server Driver
ASP	ODBC or OLE DB	Microsoft Access Driver Microsoft SQL Server Driver Microsoft SQL Server Provider Microsoft ODBC for Oracle
ASP.NET	OLE DB	Microsoft Jet Provider Microsoft SQL Server Provider Microsoft Oracle provider
PHP	MySQL specific	MySQL driver

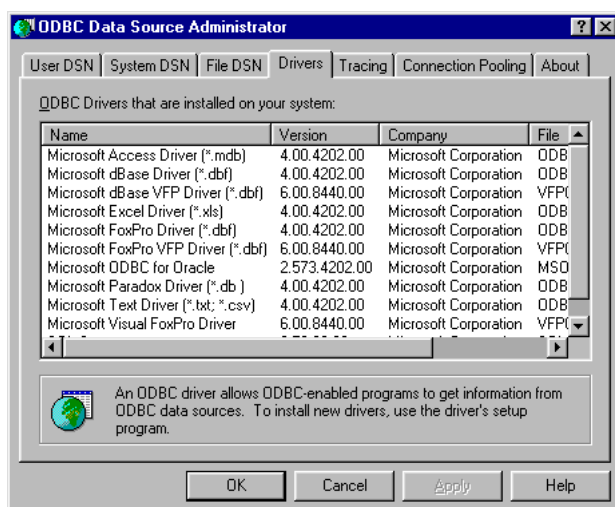
Viewing the ODBC drivers installed on a Windows system

If you need a specific ODBC driver and your web server runs on a Windows system, you can easily find out whether or not the ODBC driver you need is installed on your system.

To view the ODBC drivers installed on a Windows system:

- 1 Open the ODBC Data Source Administrator as follows:
 - In Windows 95, 98, or NT, choose Start > Settings > Control Panel, then double-click the ODBC Data Sources icon. Depending on your system, the icon could also be called ODBC or 32bit ODBC.
 - In Windows 2000, choose Start > Settings > Control Panel > Administrative Tools > Data Sources.

- In Windows XP, choose Start > Control Panel > Performance and Maintenance > Administrative Tools > Data Sources (ODBC).
- 2 Click the Drivers tab.
A list of ODBC drivers installed on the system appears.



Invoking database drivers

An application must invoke a database driver to establish two-way communications with a database. A web application invokes a driver by using a connection string. A connection string consists of all the information (or parameters) required to establish a connection to a database. In its simplest form, a connection string specifies a driver and a database, as in this example:

```
Driver={Microsoft Access Driver (*.mdb)};
DBQ=C:\inetpub\wwwroot\Scaal\scaalcoffee.mdb
```

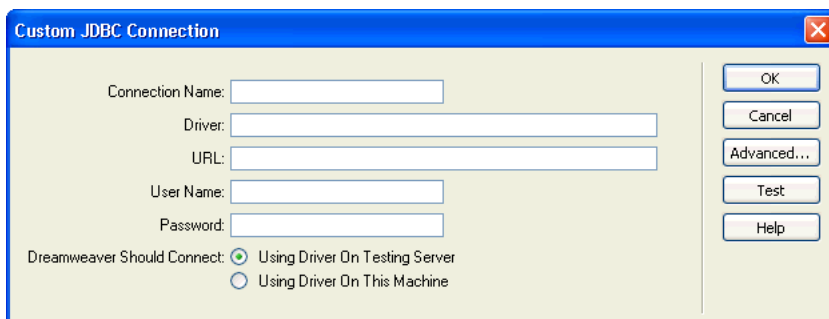
ASP connection strings can contain a Provider parameter specifying an OLE DB provider. If you omit this parameter, by default ASP uses the OLE DB provider for ODBC drivers. In the above example, the OLE DB provider for ODBC drivers would communicate with the ODBC driver, Microsoft Access Driver, which in turn would communicate with the Access database, scaalcoffee.mdb.

The parameters in a connection string may vary depending on the driver. Here's a connection string for a SQL Server database called Cases located on a server named Hoover:

```
Driver={SQL Server};Server=Hoover;Database=Cases;
UID=DanaS;PWD=Queuequeg
```

Note: UID stands for user ID; PWD for password.

Dreamweaver simplifies the process of inserting connection strings into your pages by providing you with a dialog box in which to enter the different connection parameters. For example, here's how the dialog box to define a connection looks when you're developing a JSP application:



After you complete the dialog box and click OK, Dreamweaver inserts the connection string in an include file in your site.

Using a DSN in a connection string

You can specify data source names (DSNs) in some connections. A DSN is a type of shortcut you create in Windows for a connection string. Once defined, you can simply refer to the connection string by name. For example, a connection string may consist of the following parameters:

```
Driver={SQL Server};Server=Clinic-6;Database=Patients;  
UID=dholmes;PWD=stetson2
```

After defining a DSN called `patients` in Windows using the above parameters, you can use the connection string in your application by specifying a single parameter:

```
dsn=patients
```

If your application server is running on a Windows system and you defined a DSN on that system, then you can use the DSN to define an ASP or ColdFusion 4 or 5 connection.

If you do not have physical access to a server—and so are unable to define a DSN on it—then you must use a connection string to connect to the database.

For more information, see “Setting Up a DSN in Windows” on page 671.

APPENDIX B

SQL Primer

This appendix describes how to use Structured Query Language (SQL) to create recordsets for your dynamic pages. SQL (pronounced *sequel*) is a language that lets you read and write data from a database. The language has only a few keywords and simple syntax rules, but still allows you to perform sophisticated database operations.

This appendix contains the following sections:

- “Syntax basics” on page 663
- “Defining the columns of a recordset” on page 665
- “Limiting the records in a recordset” on page 665
- “Sorting the records in a recordset” on page 668
- “Joining tables” on page 669

Note: Macromedia does not provide technical support for third-party technologies such as SQL.

Syntax basics

One of the most frequently used SQL statements is the `SELECT` statement, which extracts specified columns from one or more database tables to build a recordset. Here’s the basic syntax for a `SELECT` statement:

```
SELECT ColumnName FROM TableName
```

You can add line breaks, tabs, and other white space to your statements to clarify the logic: SQL ignores all white space. The following example shows a valid statement:

```
SELECT PaidDues  
FROM Members
```

The following keywords identify commonly used SQL commands:

Keyword	Description
SELECT	Retrieves the specified records from a database
INSERT	Adds a new record in a database table
UPDATE	Changes values in specified database records
DELETE	Removes specified database records

The following keywords are used to refine SQL statements:

Keyword	Description
FROM	Names the data source for an operation
WHERE	Sets one or more conditions for the operation
ORDER BY	Sorts the recordset rows in a specified order
GROUP BY	Groups the recordset by the specified select list items

The following operators specify conditions and perform logical and numeric functions:

Operator	Meaning
=	Equal to
LIKE	Like (wildcards OK)
<>	Not equal to
NOT LIKE	Not like (wildcards OK)
<	Less than
>	Greater than
<=	Less than or equal to
>=	Greater than or equal to
AND	Both conditions must be met, such as Louisiana AND Texas
OR	At least one condition must be met, such as Smith OR Smyth
NOT	Exclude the condition following, such as Paris NOT France

If the item being compared is text, place it in single quotes as in the following example:

```
...WHERE Country = 'Germany'
```

If the item being compared is a date and you're working with a Microsoft Access database, enclose it with # signs:

```
...WHERE DateOfBirth < #01/01/1970#
```

Other databases may have their own date conventions. Consult the database system's documentation.

Some database systems may use non-standard SQL syntax in their products. Check your database system's documentation.

Defining the columns of a recordset

You can use SQL to define recordsets for your pages. A recordset is a subset of records extracted from a database. For more information, see “About databases” on page 651.

Here’s the basic SQL syntax to define the columns of a recordset:

```
SELECT ColumnName1, ColumnName2, ColumnNameX FROM TableName
```

If you want to include all the columns of a table in the recordset, you can use the wildcard character *, as follows:

```
SELECT * FROM TableName
```

For example, suppose you have a table called `Customers`. To extract all the columns, you would type the following `SELECT` statement:

```
SELECT * FROM Customers
```

Suppose you only need the data contained in two columns of the `Customers` table: the `YearBorn` column and the `DateLastPurchase` column. To create a recordset containing only the data from these two columns, you would type the following `SELECT` statement:

```
SELECT YearBorn, DateLastPurchase  
FROM Customers
```

Limiting the records in a recordset

Use a `WHERE` clause to limit the number of records in the recordset. For example, you may want to include only those customers who earn more than \$50,000 a year. Assume you have a column in your table called `Earnings` that tells you how much each customer earns. Your `SELECT` statement would read as follows:

```
SELECT YearBorn, DateLastPurchase FROM Customers  
WHERE Earnings > 50000
```

You specify one or more conditions in a `WHERE` clause to filter out records in the database. The following sections describe ways to filter records with the `WHERE` clause:

- “Filtering records based on the equality of two values” on page 665
- “Filtering records based on the likeness of two values” on page 666
- “Filtering records based on a range of values” on page 667
- “Filtering records based on a combination of search conditions” on page 668

Filtering records based on the equality of two values

You can filter records in a database based on the equality of a parameter’s value with a record column’s value.

Suppose you decide to let users search the database by department. The following logic is required to build the search results recordset:

- Check a record in the database table.
- If the value in the “department” column of the record is equal to the department name submitted by the user, then include that record in the search results recordset.
- Check the next record in the table.

You can express this logic with the following WHERE clause:

```
WHERE ColumnName = ParameterValue
```

ParameterValue is a SQL variable containing a search parameter. In a web application, the user typically supplies this parameter using an HTML form.

This database query could be expressed fully in SQL as follows:

```
SELECT FIRSTNAME, LASTNAME, DEPARTMENT, EMAIL  
FROM EMPLOYEES  
WHERE DEPARTMENT = 'varDept'
```

This SQL statement finds all the records in the employee table with a DEPARTMENT value equal to the value contained in the varDept variable. For example, if the user specifies Operations as the department name, the SQL statement might generate the following recordset:

FIRSTNAME	LASTNAME	DEPARTMENT	EMAIL
David	Grandel	Operations	david@compasstravel
Welan	Davis	Operations	wenlan@compasstravel.com
Ken	Smith	Operations	ken@compasstravel.com
Chris	Bates	Operations	chrisb@compasstravel.com
Dan	Riely	Operations	dan@compasstravel.com

Filtering records based on the likeness of two values

You can filter records in a database based on the likeness of a parameter's value with a record column's value.

Using likeness instead of equality gives users more flexibility when specifying the value of search parameters. For example, search words don't need to be case sensitive. If the user enters "ohio" and the table column contains the value "Ohio", the match is made.

Also, likeness lets you use wildcard characters so users can perform alphabetical and partial-word searches. For example, if the user enters "m" and the table column contains the values "Morgan", "Macy", and "Michelson", then you can use a wildcard character in the SQL statement so that all three matches are made.

The standard wildcard character is the percentage sign (%):

```
...WHERE LastName LIKE 'Mc%'
```

Suppose you decide to let users search the database by last names. The following logic is required to build the search results recordset:

- Check a record in the database table.
- If the value in the "last name" column of the record contains a value like the value submitted by the user, then include that record in the results recordset.
- Check the next record in the table.

You can express this logic with the following WHERE clause:

```
WHERE ColumnName LIKE ParameterValue
```

ParameterValue is a SQL variable containing a search parameter. In a web application, the user typically supplies this parameter using an HTML form.

This database query could be expressed fully in SQL as follows:

```
SELECT FIRSTNAME, LASTNAME, DEPARTMENT, EMAIL
FROM EMPLOYEES
WHERE DEPARTMENT LIKE 'varLastName'
```

If you want to give users the ability to perform partial-word searches, combine the variable with a wildcard character. The SQL wildcard character to use in this case is the percentage sign (%). Here's an example:

```
...WHERE LASTNAME LIKE 'varLastName%'
```

For example, if the user types “s” as the search parameter, all records with last names starting with the letter “s” are included in the recordset, as in the following example:

FIRSTNAME	LASTNAME	DEPARTMENT	EMAIL
Mike	Smith	Administration	mike@compasstravel.com
Ken	Smith	Operations	ken@compasstravel.com
Dalbir	Stone	Administration	dalbir@compasstravel.com
Terry	Silver	Marketing	terry@compasstravel.com

If the user specifies “sm” as the search parameter, then only those records with last names that start with the letters “sm” are included in the recordset:

FIRSTNAME	LASTNAME	DEPARTMENT	EMAIL
Mike	Smith	Administration	mike@compasstravel.com
Ken	Smith	Operations	ken@compasstravel.com

Filtering records based on a range of values

You can filter records in a database based on whether a record column's value falls within the range of two parameter values.

Suppose you decide to let users search the database by a date range. The following logic is required to build the search results recordset:

- Check a record in the database table.
- If the value in the “date” column of the record falls between the two date values submitted by the user, then include that record in the results recordset.
- Check the next record in the table.

You can express this logic with the following WHERE clause:

```
WHERE ColumnName BETWEEN ParameterValue1 AND ParameterValue2
```

ParameterValue1 and ParameterValue2 are SQL variables containing search parameters. In a web application, the user typically supplies these parameters using an HTML form.

Here's how this type of database query can be expressed in SQL:

```
SELECT FIRSTNAME, LASTNAME, DEPARTMENT, STARTDATE
FROM EMPLOYEES
WHERE STARTDATE BETWEEN #varStartRange# AND #varEndRange#
```

For example, if the user enters “7/1/99” and “12/31/99” as the range parameters, all employees starting in the second half of 1999 are included in the recordset, as in the following example:

FIRSTNAME	LASTNAME	DEPARTMENT	STARTDATE
Charles	Nicholas	Trip Staff	10/1/99
David	Gallagher	Trip Staff	8/9/99
David	Grandel	Operations	7/7/99
Lars	Richie	Consultants	11/30/99

Filtering records based on a combination of search conditions

This section describe how to include records in the search results recordset based on a combination of search conditions. You combine search conditions in SQL using the AND, OR, and NOT logical operators.

If you want all the conditions to be true for a record to be included in the recordset, use the AND operator as follows:

```
...WHERE LASTNAME LIKE 'varLastName' AND DEPARTMENT LIKE 'varDept'
```

If you want any one of the conditions to be true for a record to be included in the recordset, use the OR operator as follows:

```
...WHERE LASTNAME LIKE 'varLastName' OR DEPARTMENT LIKE 'varDept'
```

If you want one condition to be true but not another, use the NOT operator as follows:

```
...WHERE DEPARTMENT LIKE 'varDept' AND NOT COUNTRY LIKE 'varCountry'
```

You can use parentheses to group search conditions:

```
...WHERE (DEPARTMENT LIKE 'varDept' AND STARTDATE < #varStart#)  
OR STARTDATE BETWEEN #varStartRange# AND #varEndRange#
```

Sorting the records in a recordset

Use the ORDER BY clause to sort the records in your recordset. For example, suppose you want to sort the records in the recordset by customer earnings, from the lowest to the highest. In SQL, order the records as follows:

```
SELECT LastName, FirstName, Earnings FROM Customers  
ORDER BY Earnings
```

By default, the ORDER BY clause sorts records in ascending order (1, 2, 3... or A, B, C...). If you want to sort them in descending order, from the highest earnings to the lowest, use the DESC keyword as follows:

```
ORDER BY Earnings DESC
```

Joining tables

You can use a single `SELECT` statement to retrieve data from more than one table in the database. The statement joins the tables and returns a single recordset containing selected data from each table.

For example, a company database might contain one table with personal data about employees and another table with data about the company's departments. If you want to build an employee directory that displays an employee's name, phone number, and department, you must retrieve information from the two tables simultaneously.

To do this, create a join specifying all the tables to include and how the tables are related to each other. Here's an example:

```
SELECT FIRSTNAME, LASTNAME, PHONE, DEPTNAME
FROM EMPLOYEES, DEPARTMENTS
WHERE EMPLOYEES.DEPT_ID = DEPARTMENTS.DEPT_ID
```

Note: Use dot notation to identify the columns more precisely. For example, `EMPLOYEES.DEPT_ID` refers to the `DEPT_ID` column in the `EMPLOYEES` table.

The first line specifies the columns to retrieve. The first three columns—`FIRSTNAME`, `LASTNAME`, `PHONE`—exist in the `EMPLOYEES` table, while the fourth column—`DEPTNAME`—exists only in the `DEPARTMENTS` table.

The second line specifies the two tables from which to retrieve data, `EMPLOYEES` and `DEPARTMENTS`.

The final line specifies the records to join and retrieve from the two tables. Each table has a column called `DEPT_ID`. (In the `DEPARTMENTS` table, the column is the primary key. For more information, see “Defining relationships between the tables” on page 656.) The `WHERE` clause compares the value of `DEPT_ID` in one table to the value of `DEPT_ID` in the other table. When a match is found, all the fields of the record in the `EMPLOYEES` table are joined with all the fields of the record in the `DEPARTMENTS` table. Next, the combined data is filtered to create a new record made up of a `FIRSTNAME`, `LASTNAME`, `PHONE`, and `DEPTNAME` column. Finally, the new record is added to the recordset.

Using slightly different join syntax may be preferable in some database systems. For example, the following SQL statement uses the SQL keywords `INNER JOIN . . . ON` to achieve the same results as the previous example:

```
SELECT FIRSTNAME, LASTNAME, PHONE, DEPTNAME
FROM EMPLOYEES INNER JOIN DEPARTMENTS
ON EMPLOYEES.DEPT_ID = DEPARTMENTS.DEPT_ID
```

Consult your database system's documentation to determine which join syntax you should use.

APPENDIX C

Setting Up a DSN in Windows

A DSN is a kind of Windows shortcut you use to establish a database connection (see “Understanding database connections” on page 657). Before you can use one in your web application, you must set it up on your computer or on a remote server as described in this appendix.

This appendix applies only if your database is located on a system that supports ODBC data source names (DSNs)—systems such as Microsoft Windows and Windows NT but not the Macintosh.

This appendix contains the following sections:

- “Understanding DSNs” on page 671
- “Creating a DSN” on page 671

Understanding DSNs

A DSN is a single-word identifier for a set of database connection parameters. The parameters can include the server name, the path to the database or the database name, the ODBC driver to use, and the user name and password, if any.

For example, suppose you have a Microsoft SQL Server database called *Precinct* located on a server called *Kojak*. To gain access to the database, you must enter the user name “*columbo*” and the password “*savalas7*”. After using these parameters to define a DSN called *ourcops*, you can create the connection by entering the single word *ourcops* in Dreamweaver MX instead of all the other parameters.

Creating a DSN

To create a DSN, an ODBC driver for your database must be installed on the Windows computer running your application server. ODBC (for Open Database Connectivity) is a software intermediary that lets an application communicate with a database. For more information, see “Interfacing with the database” on page 658.

Make sure your system has the proper driver for your database. For a list of ODBC drivers on a Windows 95, 98, or NT system, choose Start > Settings > Control Panel, then double-click the ODBC Data Sources icon. (Depending on your system, the icon could also be called ODBC or 32bit ODBC.) When you click the Drivers tab, you’ll see a list of drivers installed on the system. In Windows 2000, choose Start > Settings > Control Panel > Administrative Tools > Data Sources, then click the Drivers tab. In Windows XP, choose Start > Control Panel > Performance and Maintenance > Administrative Tools > Data Sources (ODBC), then click the Drivers tab.

If the required driver is not installed, then download and install the Microsoft Data Access Components (MDAC) 2.5 and 2.6 packages, which you can download for free from the Microsoft website at <http://www.microsoft.com/data/download.htm>. These packages contain the latest Microsoft ODBC drivers for the most common databases.

Note: Install MDAC 2.5 first, then install MDAC 2.6.

If MDAC does not have an ODBC driver for your database, see your database vendor.

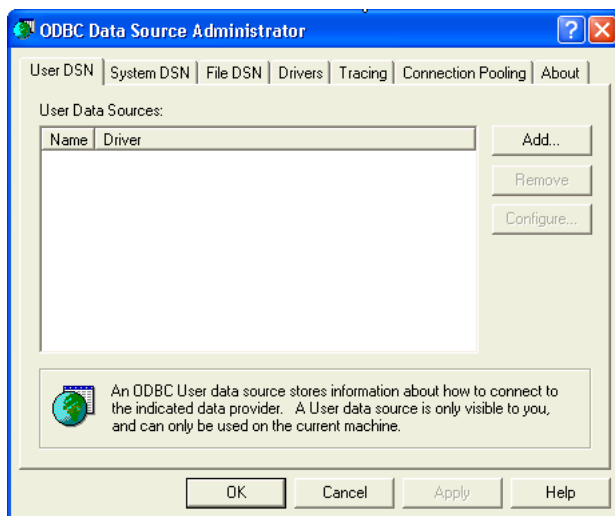
To create a DSN:

1 Open Windows' ODBC Data Source Administrator as follows:

- In Windows 95, 98, or NT, choose Start > Settings > Control Panel, then double-click the ODBC Data Sources icon. Depending on your system, the icon could also be called ODBC or 32bit ODBC.
- In Windows 2000, choose Start > Settings > Control Panel > Administrative Tools > Data Sources.
- In Windows XP, choose Start > Control Panel > Performance and Maintenance > Administrative Tools > Data Sources (ODBC).

2 In the Dreamweaver dialog box you use to create a DSN connection, click the Define button.

The ODBC Data Source Administrator appears:



3 Click the System DSN tab.

The tab displays the list of DSNs currently on your system.

4 Click Add to add a new DSN to the list.

The Create New Data Source dialog box appears, listing all the drivers currently loaded on your system.

- 5 Select a driver from the list, then click Finish.

For example, if your database is a Microsoft Access file, select Microsoft Access Driver (*.mdb). If a driver for your product does not appear in the list, you'll have to download the driver from a vendor's website and install it.

- 6 In the dialog box that appears, enter a name for the DSN and specify the connection parameters.

The dialog boxes for specifying parameters differ depending on the driver you selected. For the Microsoft Access Driver, you enter a name, click Select, locate the database file on the hard disk, and click OK.

- 7 Click OK to close the dialog box.

The new DSN is added to your list of system DSNs.

APPENDIX D

Quick Reference: Macromedia ASP.NET Tags

Macromedia Dreamweaver MX uses a set of custom tags to create server behaviors in your ASP.NET pages. You can also use these tags in your own ASP.NET pages.

The appendix describes the following Macromedia custom tags:

- “MM:DataSet” on page 675
- “MM:If” on page 677

Dreamweaver also uses the following tags to create server behaviors:

- “MM:Insert” on page 678
- “MM:Update” on page 678
- “MM>Delete” on page 679

Finally, if a tag contains a SQL statement or stored procedure that takes parameters, use the following two tags to specify the parameter values:

- “Parameters” on page 680
- “Parameter” on page 681

Note: Like the Microsoft ASP.NET tags, each Macromedia custom tag requires a `runat="server"` attribute.

For a more detailed description of the tags and their attributes, visit the Dreamweaver Support Center at http://www.macromedia.com/go/customtags_asp.

MM:DataSet

Use the MM:DataSet tag to create a DataSet. ASP.NET DataSets are very similar to ASP recordsets.

Example

```
<MM:DataSet  
id="fruitsDS"  
runat="server"  
CommandText="SELECT name, color FROM fruits"  
ConnectionString=< %# MM_NOTD_STRING %>  
>
```

Required directives

```
<%@ Register TagPrefix="MM" Namespace="MM" Assembly="MMtags" %>
```

If parameter tags will be used, then the following directives are often (but not always) needed:

```
<%@ Import Namespace="System.Data" %>  
<%@ Import Namespace="System.Data.SqlClient" %>  
<%@ Import Namespace="System.Data.OleDb" %>
```

Attributes

CommandText

Required. Specifies either a SQL statement or the name of a stored procedure. The SQL statement can be parameterized by using question marks as placeholders. See “Parameters” on page 680.

ConnectionString

Required. Specifies a connection string containing information to connect to a database.

CreateDataSet

Optional. True or False. When false, no DataSet is created. Default is true.

Expression

Optional. True or False. Lets you control when the CommandText statement is executed. If false, the CommandText statement is not executed. You can programmatically toggle the value.

IsStoredProcedure

Optional. True or False. Lets you indicate the nature of CommandText. The default value is false.

PageSize

Optional. Integer. When CreateDataSet is true, lets you divide the records into groups (pages) of a fixed size. See CurrentPage. Do not set both PageSize/CurrentPage and StartRecord/MaxRecords. If both sets of attributes are specified, the former (PageSize/CurrentPage) is used.

CurrentPage

Optional. Integer. When PageSize is used, lets you specify a particular page (group of records). The index of the first page is zero. See PageSize.

StartRecord

Optional. Integer. An offset from the first record in a DataSet (the index of the first record is zero). When CreateDataSet is true, lets you retrieve a subset of records in conjunction with the MaxRecords attribute. Do not set both PageSize/CurrentPage and StartRecord/MaxRecords. If both sets of attributes are specified, the former (PageSize/CurrentPage) is used. If PageSize is greater than zero, then StartRecord is calculated automatically as (PageSize * CurrentPage).

MaxRecords

Optional. Integer. Lets you specify how many records to retrieve starting at the StartRecord offset. See StartRecord.

GetRecordCount

Optional. True or False. When PageSize is used, lets you turn off record counting to improve performance. If the page never uses the RecordCount or LastPage attributes, then you don't need to count records. The default value is true.

RecordCountCommandText

Optional. To improve performance when `GetRecordCount` is true, lets you specify an explicit SQL statement to obtain a record count. The statement should be in the form `SELECT COUNT (*) FROM. . .` and can include `WHERE` clauses, but not parameters.

EndRecord

Optional. Read-only. Lets you obtain the minimum of $(\text{CurrentPage} + 1) * \text{PageSize}$ and `RecordCount`.

LastPage

Optional. Read-only. When `PageSize` is used and `GetRecordCount` is true, lets you obtain the zero-based index of the last page.

RecordCount

Optional. Read-only. When `CreateDataSet` and `GetRecordCount` are both true, lets you obtain the total number of records generated by the `CommandText` expression.

theDS

Optional. Read-only. Gives you access to the `DataSet` holding the records retrieved as a result of setting `CreateDataSet` to true.

DefaultView

Optional. Read-only. Gives you access to the `theDS.Tables[0].DefaultView`

Debug

Optional. True or False. Lets you see extra clues if the execution of this tag throws an exception. If the `Debug` attribute is true, then `FailureURL` is ignored (see `FailureURL`). When developing the page, set `Debug` to true to get robust diagnostics information about problems with the page. When about to deploy the page, set `Debug` to false so external visitors never see diagnostics and other internal information in case of problems with the page.

FailureURL

Optional. Lets you specify the URL of a page to open if an exception is thrown while executing the `CommandText` or if the given attributes conflict.

SuccessURL

Optional. Lets you specify the URL of a page to open if the `CommandText` statement executes with no exceptions thrown.

MM:If

Use the `MM:If` tag to control the inclusion of content or logic on a page.

Example

```
<MM:If  
runat="server"  
Expression="<%# (fruitsDS.CurrentPage == 0) %>">  
  <ContentsTemplate>  
    <font color='#CCCCCC'>Previous</font>  
  </ContentsTemplate>  
</MM:If>
```

Required directive

```
<%@ Register TagPrefix="MM" Namespace="MM" Assembly="MMtags" %>
```

Attribute

Expression

Required. Lets you specify an expression to resolve. If the expression is true, then the tag's inner content is used. If the expression is false, the inner content is ignored.

MM:Insert

Use the MM:Insert tag to insert records in a database.

Example

```
<MM:Insert
runat="server"
CommandText="INSERT INTO fruits (name, color) VALUES (?, ?)"
ConnectionString='<%# MM_NOTD_STRING %>'
Expression=<%# Request.Form["MM_insert"] == "true" %>
Debug="true"
CreateDataSet="false"
>
  <Parameters>
  <Parameter Value='<%# Request.Form["fruitName"] %>' />
  <Parameter Value='<%# Request.Form["fruitColor"] %>' />
  </Parameters>
</MM:Insert>
```

Required directive

```
<%@ Register TagPrefix="MM" Namespace="MM" Assembly="MMtags" %>
```

Attribute

The MM:Insert tag uses the logic used by the MM:DataSet tag, so it shares all of that tag's attributes. For more information, see "MM:DataSet" on page 675. Note the following attributes values:

Expression

By setting this attribute to `false`, the tag does nothing. That is, the SQL statement or stored procedure specified in the `CommandText` attribute is not executed. You can use the `Expression` attribute to control when the insert operation takes place. For example, you might want to perform server-side form validation before carrying out the insert operation. In another case, the code might set `Expression` to `true` only if the form containing the data to insert has been submitted.

CommandText

Typically, this is an INSERT SQL statement or a stored procedure that inserts records.

CreateDataSet

This attribute is normally set to `false` because a `DataSet` is not created after executing an INSERT statement. In the case of a stored procedure, however, a `DataSet` might be created after the insert operation takes place.

MM:Update

Use the MM:Update tag to update records in a database.

Example

```
<MM:Update
runat="server"
CommandText="UPDATE fruits SET color=? WHERE name=?"
ConnectionString='<# MM_NOTD_STRING %>'
Expression=<# Request.Form["MM_update"] == "true" %>
Debug="true"
CreateDataSet="false"
>
  <Parameters>
  <Parameter Value='<# Request.Form["fruitColor"] %>' />
  <Parameter Value='<# Request.Form["updateNames"] %>' />
  </Parameters>
</MM:Update>
```

Required directive

```
<%@ Register TagPrefix="MM" Namespace="MM" Assembly="MMtags" %>
```

Attribute

The MM:Update tag uses the logic used by the MM:DataSet tag, so it shares all of that tag's attributes. For more information, see "MM:DataSet" on page 675. Note the following attributes values:

Expression

By setting this attribute to `false`, the tag does nothing. That is, the SQL statement or stored procedure specified in the `CommandText` attribute is not executed. You can use the `Expression` attribute to control when the update operation takes place. For example, you might want to perform server-side form validation before carrying out the update operation. In another case, the code might set `Expression` to `true` only if the form containing the data to update has been submitted.

CommandText

Typically, this is an UPDATE SQL statement or a stored procedure that updates records.

CreateDataSet

This attribute is normally set to `false` because a DataSet is not created after executing an UPDATE statement. In the case of a stored procedure, however, a DataSet might be created after the update operation takes place.

MM:Delete

Use the MM:Delete tag to delete records in a database.

Example

```
<MM>Delete
runat="server"
CommandText="DELETE FROM fruits WHERE name=?"
ConnectionString='<# MM_NOTD_STRING %>'
Expression=<# Request.Form["MM_delete"] == "true" %>
Debug="true"
CreateDataSet="false"
>
  <Parameters>
  <Parameter Value='<# Request.Form["deleteNames"] %>' />
  </Parameters>
</MM>Delete>
```

Required directive

```
<%@ Register TagPrefix="MM" Namespace="MM" Assembly="MMtags" %>
```

Attribute

The MM:Delete tag uses the logic used by the MM:DataSet tag, so it shares all of that tag's attributes. For more information, see “MM:DataSet” on page 675. Note the following attributes values:

Expression

By setting the Expression to `false`, the tag does nothing. That is, the SQL statement or stored procedure specified in the CommandText attribute is not executed. You can use the Expression attribute to control when the delete operation takes place. For example, you might want to perform server-side validation before carrying out the delete operation.

CommandText

Typically, this is a DELETE SQL statement or a stored procedure that deletes records.

CreateDataSet

This attribute is normally set to `false` because a DataSet is not created after executing a DELETE statement. In the case of a stored procedure, however, a DataSet might be created after the delete operation takes place.

Parameters

Use the Parameters tag to specify parameters. There are two conditions under which parameters might be specified:

- The CommandText attribute of the main tag specifies a parameterized SQL statement. This is an SQL statement that has question marks (?) as placeholders for parameters.

Note: It is not legal to parameterize every part of every SQL statement. For example, it is possible to parameterize the WHERE clause of a SELECT statement but not its ORDER BY clause. Generally, you can parameterize any name/value pair in a SQL statement.

- The CommandText attribute specifies a stored procedure that takes parameters.

In both of these cases, parameter values are provided by Parameter tags that appear in the same order as the placeholders (see “Parameter” on page 681). The Parameter tags must be surrounded by a Parameters tag.

Example

```
<MM:Update
runat="server"
CommandText="UPDATE fruits SET color=? WHERE name=?"
ConnectionString='<%=# MM_NOTD_STRING %>'
Expression=<%=# Request.Form["MM_update"] == "true" %>
Debug="true"
CreateDataSet="false"
>

    <Parameters>
    <Parameter Value='<%=# Request.Form["fruitColor"] %>' />
    <Parameter Value='<%=# Request.Form["updateNames"] %>' />
    </Parameters>

</MM:Update>
```


Parameter

Use the Parameter tag to specify parameter values.

Example

```
<MM:Insert
runat="server"
CommandText="INSERT INTO fruits (name, color) VALUES (?, ?)"
ConnectionString='<# MM_NOTD_STRING %>'
Expression=<# Request.Form["MM_insert"] == "true" %>
Debug="true"
CreateDataSet="false"
>

    <Parameters>
    <Parameter Value='<# Request.Form["fruitName"] %>' />
    <Parameter Value='<# Request.Form["fruitColor"] %>' />
    </Parameters>

</MM:Insert>
```

Attributes

Value

Required. The value to be assigned to Input or InputOutput parameters (see Direction). The value will be cast to an appropriate data type before being passed to the database. The casting depends on the setting of the Type attribute. When the Direction attribute is Output, InputOutput, or ReturnValue, the Value will be set after the CommandText is executed.

Direction

Optional. Lets you specify one of the following values: Input, InputOutput, Output, ReturnValue. Default is Input.

Name

Optional. For parameterized SQL statements, the name can be anything. Typically, the name relates to the parameter's meaning, such as @deptName, and begins with the @ character. However, the only requirement is that the order in which the Parameter tags appear must correspond to the order of the question marks in the SQL statement. Since the name is optional, you can omit this attribute, in which case the name will be set to "parameter_" plus the one-based index of the parameter within the list of parameters provided.

For stored procedures, the name must match the parameter markers in the stored procedure (for example, @id). This means you must provide this attribute when using stored procedures. If a parameter of type ReturnValue is needed, its Parameter tag must occur before any other Parameter tag.

Size

Optional. The size of the parameter's value. This is also known as the width of the column. If not provided, the value of this attribute is inferred from the Type and Value attributes.

Type

Optional. Specifies the parameter type. You can specify the following data types, each of which can be expressed in one of two ways.

(int)OleDbType.Boolean	"Boolean"
(int)OleDbType.UnsignedTinyInt	"UnsignedTinyInt"
(int)OleDbType.Char	"Char"
(int)OleDbType.DBTimeStamp	"DBTimeStamp"
(int)OleDbType.Decimal	"Decimal"
(int)OleDbType.Double	"Double"
(int)OleDbType.Single	"Single"
(int)OleDbType.Integer	"Integer"
(int)OleDbType.SmallInt	"SmallInt"
(int)OleDbType.BigInt	"BigInt"
(int)OleDbType.VarChar	"VarChar"
(int)OleDbType.UnsignedSmallInt	"UnsignedSmallInt"
(int)OleDbType.UnsignedInt	"UnsignedInt"
(int)OleDbType.UnsignedBigInt	"UnsignedBigInt"
(int)OleDbType.Currency	"Currency"
(int)OleDbType.DBDate	"DBDate"
(int)OleDbType.TinyInt	"TinyInt"
(int)OleDbType.DBTime	"DBTime"
(int)OleDbType.VarNumeric	"VarNumeric"

The following OleDbType values are not supported:

(int)OleDbType.Guid	"Guid"
(int)OleDbType.Variant	"Variant"

INDEX

Symbols

@import 290

A

Absolute Bottom alignment (image Property inspector)
300

Absolute Middle alignment (image Property inspector)
300

Absolute paths 400

Access options 63

access privileges

adding to pages 646

example 645

storing in a database 647

accessibility

authoring for 337

dialog box, activating 337

Federal Rehabilitation Act 333

frames 340

images 338

keyboard-only navigation 334

media objects 341, 342

objects, inserting 339

operating system features 334

screen reader 334

testing 343

Web Accessibility Initiative (W3C) 333

actions

browser compatibility 354

changing in behaviors 353

choosing in Behaviors panel 351

controlling timelines 395

creating 353

defined 349

included with Dreamweaver 354–376

See also individual actions by name

Active Links color option (Page Properties) 115

ActiveX controls

inserting 330

overview 330

ActiveX objects, making dynamic 525

Add Frame command 390

Add Object to Library command 432

Add Object to Timeline command 390

adding

dynamic content 519

extensions to Dreamweaver 53

frames to a timeline 392

links 424, 425

objects to a timeline 390

rows and columns 234

Adjust Position command 118

Align with Selection command 118

aligning

images 276

layers 383

options 300

page elements 300

text 276

tracing images 118

Always Nest When Created Within Existing Layer

option 380

Anchor object (Insert bar) 406

anchors, named 406

animations

along complex path 392

applying to objects 394

copying and pasting 394

creating 390

improving 395

overview 21

timelines 388

anonymous file upload 582

Answers panel 49

applets. *See* Java applets

- application server
 - choosing 129
 - setting up 129
 - application servers
 - troubleshooting errors 136
 - application variables 510
 - applications, other, using with Dreamweaver 49
 - Apply Source Formatting command 178
 - Apply Template to Page command 456
 - applying
 - behaviors to images 303
 - colors to text 423
 - CSS styles 289
 - custom styles 289
 - HTML styles 283
 - ASP 131
 - database connections 151
 - ASP.NET 130
 - database connections 145
 - DataGrid 538
 - Delete Buttons 538
 - Edit, Update, Cancel Buttons 538
 - Free Form 538
 - Hyperlink 538
 - Simple Data Field 538
 - DataList 538
 - DataSet custom tag 675
 - Delete custom tag 679
 - dynamic form elements 592
 - If custom tag 677
 - importing tags 185
 - Insert custom tag 678
 - Macromedia custom tags 675
 - Parameter tag 681
 - Parameters tag 680
 - Update custom tag 678
 - writing SQL for 506
 - assets
 - applying colors to text 423
 - categories 421
 - colors, creating 429
 - editing 424
 - Favorites folders 430
 - Favorites lists 428
 - inserting 423
 - Library category of Assets panel 425
 - opening the Assets panel 420
 - overview 420
 - planning 59
 - refreshing the Site list 422
 - selecting multiple 424
 - Templates category of Assets panel 426
 - URLs, creating 429
 - viewing in Assets panel 421
 - Attach Style Sheet icon 290
 - attributes
 - making dynamic 523
 - searches 197
 - See also* tags
 - audience 56
 - audio. *See* sound
 - authorization levels 645
 - Auto Apply option (HTML Styles panel) 281
 - Auto Refresh 479
 - automating tasks 120
 - Autoplay option 389
 - Autoplay option (Timelines panel) 390
 - autostretch (Layout view) 251
- B**
- background
 - page image and color 113
 - transparency in 113
 - balancing braces 204
 - Baseline alignment (image Property inspector) 300
 - basics of Dreamweaver 23
 - BBEdit integration (Macintosh only) 187
 - behaviors 349–376
 - actions, creating 353
 - and forms 591
 - and images 303
 - and library items 434
 - and links 418
 - and media 331
 - and timelines 352
 - attaching 351
 - changing 353
 - channel in timelines 389
 - compatibility with browsers 354
 - defined 349
 - deleting 353
 - third-party 354
 - triggering 351
 - Behaviors panel 350
 - Bindings panel
 - adding dynamic text 520
 - creating a record counter 537
 - deleting data sources 516
 - Format column 528
 - making forms dynamic 592
 - making HTML attributes dynamic 523

- bitmaps, resizing 301
 - See also* images
 - blockquote
 - applying 276
 - body tag 119
 - bold text 274
 - Bottom alignment (image Property inspector) 300
 - braces, balancing 204
 - breakpoints in JavaScript debugger 212
 - broken links 465
 - Browser Default alignment (image Property inspector) 300
 - browsers
 - 3.0 compatible files 387
 - and CSS styles 293
 - colors, safe 115
 - compatibility, planning 56
 - compatibility, testing 462
 - previewing in 463
 - primary, defining 464
 - targeting 462
 - versions, checking 355
 - buttons
 - creating form buttons 589
 - Go buttons 362
 - Submit buttons, graphical 590
- C**
- caching data sources 516
 - Call JavaScript action 354
 - Callable (Stored Procedure) option 629
 - catalogs 481
 - categories
 - assets 421
 - preferences 50
 - Cell Padding option (Insert Table) 228
 - Cell Spacing option (Insert Table) 228
 - cells. *See* layout cells, table cells
 - CGI
 - reference material 29
 - CGI (Common Gateway Interface)
 - scripts 591
 - Change Link Sitewide command 412
 - Change Property action 355
 - Check Browser action 355
 - Check In File(s) 72
 - Check In/Check Out 88
 - Check In/Check Out reports 102
 - Check Links Sitewide command 466
 - Check Out File(s) 72
 - Check Plugin action 356
 - Check Spelling command 294
 - Check Target Browsers command 462
 - checkboxes 584
 - checkboxes, making dynamic 594
 - checking files in and out
 - overview 89
 - undoing a check-out 90
 - checking links 465
 - Clean Up HTML command 203
 - Clear Paragraph Style option 281
 - Clear Selection Style option 281
 - cloaking 94
 - code
 - editing with BBEdit 187
 - formatting preferences 177
 - code blocks
 - coding guidelines 566
 - parameter markers 568
 - positioning 567
 - writing 558
 - Code Coloring preferences 180
 - Code Hints 191
 - preferences 179
 - Code inspector 175
 - code snippets 192
 - Code view 175
 - opening non-HTML files 180
 - options 176
 - toggling to Design view 37
 - viewing templates 439
 - writing and editing code 191, 194
 - coding environment 189
 - coding guidelines 566
 - ColdFusion 130
 - client variables 512
 - ColdFusion variables 512
 - components 543–546
 - Components panel 546
 - creating data sources 140
 - database connections 139
 - debugger 215
 - pages that use components 546
 - UltraDev 4 127
 - UltraDev 4 connections 141
 - ColdFusion Components panel 546
 - color box option 114
 - Color Cubes color palette 114
 - color picker
 - Dreamweaver 114
 - system 114

- Color Wheel (System Color) button 114
 - colors
 - as assets. *See* assets
 - choosing 114
 - creating color assets 429
 - default for page text 115
 - eyedropper, using 114
 - frame background 264
 - page background 113
 - swatches 114
 - text, changing 277, 423
 - web-safe 115
 - column header menu 252
 - Columns option (Insert Table) 228
 - columns, rows, and cells
 - adding and removing 234
 - formatting 232
 - combining floating panels 44
 - Command (Stored Procedure) option 629
 - commands
 - accessing from context menu 39, 41
 - creating, from history steps 123
 - recording 124
 - comments, inserting 195
 - Common Gateway Interface (CGI). *See* CGI 591
 - connecting to databases 135
 - connection parameters 167
 - Connection Speed option 39
 - connection string, OLE DB 147
 - connection strings 156
 - Connections dialog box 163
 - content, adding to tables 228
 - context menus 39, 41
 - Continuous Tone color palette 114
 - Control Shockwave or Flash action 357
 - conventions 26
 - Convert Layers to Tables command 386
 - Convert Tables to Layers command 387
 - converting styles
 - to HTML
 - CSS styles to HTML tags 293
 - copying and pasting
 - Fireworks HTML 311
 - recordsets 517
 - copying history steps 122
 - Create Web Photo Album command 313
 - creating 408
 - a new CSS style 288
 - a rollover image 301
 - an HTML style 282
 - colors and URLs 429
 - links 402
 - CSS styles 110
 - applying custom 289
 - applying custom (class) 289
 - attributes, converting to HTML 293
 - browsers and 293
 - class attribute 288
 - creating 288
 - exporting 292
 - removing 287
 - removing style from a selection 289
 - using with XHTML 208
 - CSS Styles panel 286
 - Custom Connection String dialog box 162
 - custom styles. *See* style sheets
 - custom tags, importing 184
 - customizing
 - Dreamweaver
 - basics 50
 - Launcher bar 47
- D**
- data formats
 - applying 528
 - creating 529
 - editing 528
 - Data Link Properties 147
 - Data Source Name (DSN) dialog box 162
 - data source name, setting up 671
 - data sources
 - about 502
 - application variables 510
 - caching 516
 - ColdFusion variables 512
 - deleting 516
 - editing or deleting 516
 - form parameters, about 493
 - JavaBeans 514
 - JSP variables 513
 - recordsets, about 502
 - recordsets, creating (simple) 503
 - session variables 509
 - session variables, about 495
 - submitted by users 492
 - URL parameters, about 494
 - database
 - connections, about 657
 - content storage 491
 - design 652

- records 651
- tables, about 651
- database connections 135
 - ASP 151
 - ASP.NET 145
 - ColdFusion 139
 - connection strings 156
 - JSP 165
 - OLE DB 157
 - PHP 171
- database drivers
 - basics 658
 - viewing installed drivers 660
- Database Items tree 506
- databases
 - about 651
 - MySQL 171
 - schemas and catalogs 481
- DataSet custom tag (ASP.NET) 675
- dates
 - inserting 278
- debugging ColdFusion pages 215
- Default Color button 114
- Define HTML Style dialog box 281
- defining sites 132
- Delete custom tag (ASP.NET) 679
- delete page, building a 626
- Delete Record behavior 629
- deleting
 - a data source 516
 - a recordset 516
 - dynamic content 489, 526
 - rows and columns 234
- dependent files
 - getting 91
 - putting 92
 - showing and hiding 85
- design 55
 - files 109
 - frames 257
- Design Notes
 - adding status choices 100
 - for documents and objects 99
 - for Fireworks files 101
 - for media objects 321
 - overview 97
 - reports 102
 - saving file information 98, 99
 - setting up 98
- Design view
 - toggleing to Code view 37
 - viewing templates 438
- Detach from Original option 434
- Detach from Template command 457
- detail pages
 - creating link to 614
 - find specified record 603, 615, 616
- Director, creating Shockwave movies with 326
- directory structure, site. *See* sites
- Display Dependent Files option 85
- display options
 - floating panels 47
 - fonts 50
- displaying
 - layers 385
 - multiple records 533
 - tracing images 118
- docking panels and panel groups 44
- Document toolbar 36
- Document Type Definition 185
- Document window 36
 - basics 37
 - frames in 263
 - opening existing documents 111
 - opening new documents 109
 - page size and download time 38
 - playing Navigator plug-ins 329
 - resizing 38
 - searching for text 294
 - selecting elements 116
 - status bar 37
 - tag selector 37
 - title bar 37
 - viewing code 37
 - Window Size pop-up menu 37
- document-relative paths
 - about 400
 - setting 403
- documents
 - applying a template 456
 - checking links 465
 - creating 108
 - Design Notes, using with 97
 - download size, time 467
 - opening 111
 - opening new. *See* documents 109
 - page title 113
 - previewing in browsers 463
 - saving 111

- saving as a template 441
 - searching 294
 - setting properties 112
 - templates, detaching from 457
 - download time 38
 - downloading
 - behaviors 354
 - size, time estimates 467
 - downloading files 90
 - Drag Layer action 358
 - drawing
 - layers 378
 - layout cells and tables 243
 - Dreamweaver and Fireworks integration
 - creating web photo albums 313
 - inserting Fireworks files 307
 - Launch and Edit preferences 307
 - launching and editing Fireworks images 309, 315
 - launching and optimizing Fireworks images 310
 - Optimize Image in Fireworks command 310
 - updating Fireworks HTML 312
 - Dreamweaver discussion group 19
 - Dreamweaver Help 18
 - Dreamweaver site 132
 - Dreamweaver Support Center 19
 - DSN 152
 - DSN, connections without 155
 - DTD files 185
 - Dynamic Checkbox dialog box 595
 - dynamic content 504
 - about 502
 - adding to pages 519
 - attributes 523
 - deleting from a page 489
 - forms 592
 - images 521
 - objects 525
 - recordset, creating 503
 - recordset, defining 502
 - removing 526
 - replacing 526
 - text 520
 - Dynamic Data dialog box 524
 - dynamic form elements
 - ASP.NET 592
 - Dynamic List/Menu dialog box 593
 - Dynamic Radio Buttons dialog box 595
- E**
- Edit Font List command 275
 - Edit Format List dialog box 529
 - Edit NoFrames Content command 266
 - Edit Style Sheet dialog box 291
 - Edit with BBEdit command 187
 - Edit with HomeSite command 187
 - editable regions
 - creating 442
 - removing 443
 - editable regions (templates) 436
 - editable tag attributes (templates) 437, 446
 - editing
 - a CSS Style Sheet 291
 - an HTML style 284
 - assets 424
 - code (Code view or Code inspector) 191, 194
 - CSS style sheet 291
 - data sources 516
 - external style sheets 290
 - Flash button objects 324
 - keyboard shortcuts 51
 - recordsets 516
 - server behaviors 571
 - editors. *See* external editors
 - elements
 - aligning 300
 - e-mail files 111
 - e-mail links
 - changing 412
 - creating 407
 - encodings 50
 - errors in JavaScript code 212
 - event handlers. *See* events
 - events
 - available for different browsers and objects 350
 - changing in behaviors 353
 - defined 349
 - triggering actions 351
 - Excel. *See* Microsoft Excel files, importing.
 - Export Table command 240
 - exporting
 - sites 86
 - XML, tag notations for 460
 - exporting styles 292
 - to create a CSS style sheet 292
 - exporting table data 239
 - Extending Dreamweaver MX* documentation 19
 - extensibility
 - third-party behaviors 354

- Extension Manager 53
- extensions 53
 - creating 558
 - installing 557
- external editors
 - BBEdit (Macintosh only), integration with 187
 - HTML, overview 187
 - images 302
 - media 320
 - text, overview 187
- external links 465
- external style sheets
 - creating 289
 - editing 290
 - linking to 289
- eyedropper 114
- F**
- Favorites folders 430
- features, new in Dreamweaver 26
- fields
 - creating hidden fields 583
 - creating text fields 578
 - file fields 582
 - uploading files to a server 582
- file browser 80
- file formats, image 297
- file types
 - Flash files 322
- File Types/Editors preferences 188, 321
- files
 - browsing 80
 - cloaking 94
 - creating 108
 - downloading and uploading 90
 - organizing 57
 - putting 91
 - searching 294
 - supported types 190
 - synchronizing local and remote sites 93
 - text 111
 - uploading 91
 - viewing in Site panel 78
- Find In options 294
- finding a specific record 615
- finding and replacing. *See* searching
- Fireworks
 - Design Notes in 101
- Fireworks and Dreamweaver integration. *See* Dreamweaver and Fireworks integration
- Flash button dialog box 322
- Flash button objects 322
 - modifying 324
 - previewing 324
- Flash content
 - overview 322
- Flash movies
 - as assets. *See* assets
 - controlling 357
 - inserting 326
 - overview 322
- Flash objects, making dynamic 525
- Flash Text dialog box 324
- Flash text objects
 - inserting 324
 - previewing 324
- folders
 - Favorites 430
 - searching 294
 - searching text 294
- foms
 - list/menu objects, making dynamic 593
- font characteristics
 - changing 274
- fonts
 - changing characteristics 274
 - changing combinations 275
 - encodings, setting fonts for 50
- Fonts/Encoding preferences 50
- Format column 528
- Format Table command 232
- formatting HTML source code
 - preferences 177
- forms
 - adding to a document 575
 - behaviors, using with 591
 - buttons 589
 - client-side scripting 590
 - creating 575
 - creating file fields 582
 - fields, validating 376
 - inserting tables 590
 - jump menus, creating 413
 - making checkboxes dynamic 594
 - making image fields dynamic 594
 - making radio buttons dynamic 595
 - making text fields dynamic 594
 - overview 573
 - password field 578
 - pop-up menu 588
 - radio button groups 585

- scrolling lists 586
- server-side scripting 591
- Submit buttons 590
- text fields, creating 578
- using to gather data 608

fps (frames per second) 389

frames 255–267

- behaviors, using with 267
- browser compatibility 266
- changing background color 264
- changing content 265
- creating 259
- defined 256
- deleting 260
- nested 260
- overview 255
- panel 261
- properties 264
- saving 263
- selecting 261
- steps in creating 258
- targeting 265
- using links 265

Frames panel 261

frames per second (fps) 389

framesets 259–??

- nested 260
- predefined, inserting 259
- properties 264
- saving 263
- selecting 261
- targeting links in 403

See also frames

FTP 64

- get and put 90
- log 91
- troubleshooting 65

G

gathering data from users 608

General preferences 50

Generator objects, making dynamic 525

Get command 90

Get More Behaviors command 354

getting and putting files 90

Getting Started

- overview 18

GIF images

- as tracing image 118
- uses for 297

Go button, associating with a jump menu 362

Go to Detail Page behavior 603, 614

Go to Related Page behavior 618

Go To Timeline Frame action 374

Go To URL action 361

going

- to a detail page 614
- to a related page 617

going to a detail page 614

graphics. *See* images

Grayscale color palette 114

grid

- as guide 117
- showing 246, 383
- snap to 246
- snapping layers to 383
- spacing 246

H

head section, editing content in 119

Help 18

hidden files, showing and hiding 85

Hide Pop-up Menu 373

hiding

- links 532

hiding invisible elements 116

History panel

- automating tasks with 120
- clearing the history list 49
- commands, creating from history steps 123
- maximum steps, setting 49
- overview 48
- steps, applying to other objects 121
- steps, copying and pasting 122
- steps, repeating 120

home directory 135

home page, setting 84

HomeSite 187

horizontal rules, inserting and modifying 279

hotspots

- applying behaviors to 303
- in image maps 417
- resizing 418
- selecting multiple in an image map 417

HTML

- formatting 271
- inserting 271
- non-breaking space 273
- searching 294
- tag styles 285

HTML attributes, making dynamic 523

- HTML files
 - importing 112
 - HTML forms. *See* forms
 - HTML source code
 - searching 294
 - tag styles
 - HTML style
 - editing an existing style 284
 - HTML styles
 - applying 281, 283
 - clearing 281
 - creating 282, 283
 - deleting 281
 - modifying 284
 - New Style icon 282
 - removing from panel 281
 - using in other sites 284
 - viewing 281
 - HTTP server 129
 - Hyperlink dialog box 407
 - hypertext links 402
- I**
- IBM WebSphere 131
 - If custom tag (ASP.NET) 677
 - Ignore Whitespace Differences option 295
 - image inspector 416
 - image maps
 - creating client-side 417
 - hotspots 417
 - overview 416
 - selecting multiple hotspots 417
 - images
 - about 297
 - aligning 276
 - applying behaviors to 303
 - as assets. *See* assets
 - changing source file with timelines 393
 - editing 302
 - external image editors 302
 - formats, supported 297
 - image maps 416
 - inserting 298
 - inserting in Layout view 246
 - making dynamic 521
 - preloading (behavior) 364
 - restoring swapped (behavior) 374
 - scalability 301
 - swapping (behavior) 373
 - Import Table Data command 229
 - Import Word HTML command 112
 - importing
 - external CSS style sheet 290
 - sites 86
 - tabular data 272
 - text from other documents 272
 - Word HTML files 112
 - importing custom tags 184
 - importing JSP tags 185
 - Index, Help 18
 - Insert bar 36, 193
 - categories 42
 - docking 44
 - overview 42
 - preferences 44
 - Insert custom tag (ASP.NET) 678
 - Insert E-Mail Link dialog box 407
 - Insert Image Field command 590
 - Insert Jump Menu dialog box 413
 - Insert Named Anchor dialog box 406
 - Insert Navigation Bar dialog box 415
 - insert page, building 619
 - Insert Record behavior 622
 - Insert Tabular Data command 229
 - inserting
 - ActiveX controls 330
 - assets 423
 - dates 278
 - Fireworks images into Dreamweaver 307
 - Flash button objects 322
 - Flash movies 326
 - Flash text objects 324
 - images 298
 - Java applets 331
 - media elements 320
 - rollover images 302
 - server-side includes 224
 - Shockwave movies 327
 - special characters 278
 - Inserting an image placeholder 299
 - inserting assets. *See* assets
 - inspectors
 - image inspector 416
 - opening and closing with Launcher bar 39
 - Property inspector 44
 - See also* panels
 - integrating Dreamweaver with other applications 49
 - interactivity 21
 - invalid tags
 - displaying 190

- invisible elements
 - comments 195
 - scripts 221
 - selecting 116
 - showing and hiding 116

ISP 158

italics 274

J

Jakarta Tomcat 131

jar files 185

Java applets

- inserting 320, 331
- making dynamic 525
- overview 330

JavaScript

- actions 350
- alerts 364
- behaviors 349
- executing 354
- files 111
- inserting scripts 221

JavaScript debugger 209

- logical errors 212
- setting breakpoints 212
- step in, step out, step over 213
- syntax errors 211
- variable list 214
- warning box 210
- watching variables 214
- window 212

JavaServer Pages

- JavaBeans 514
- JSP variables 513
- resultset, defined 484

JDBC

- connection parameters 167
- drivers 166

JPEG images

- as tracing image 118
- uses for 297

JRun 131, 186

JSP 131

- database connections 165
- importing tags 185

Jump Menu action 361

Jump Menu Go action 362

jump menus

- adding menu items 413
- changing menu items 413
- creating a selection prompt for 413

editing 361

Go buttons 362

Go buttons, adding automatically 413

K

Keyboard Shortcut Editor 51

keyboard shortcuts, editing 51

keyframes

- creating 390
- overview 389

L

Launch and Edit preferences 307

Launcher bar 36, 39

- customizing 47

Launching an external image editor 303

layers 396

- aligning 383
- changing stacking order of 385
- changing visibility with behaviors 368
- changing visibility with Layers panel 385
- converting for 3.0 browsers 387
- converting to tables 386
- creating 378
- draggable 359
- drawing multiple 379
- in table design 386
- inserting 378
- manipulating 381
- markers, displaying 379
- moving 382
- nesting 380
- positioning 384
- preferences 380
- preventing overlap 387
- properties 384
- properties for multiple 384
- resizing 382
- selecting 381
- selecting multiple 384
- snapping to grid 383
- visibility 385

Layers panel 379

layout cells 242

- alignment 250
- bg color 250
- clearing heights 247
- drawing 243
- formatting 250
- No Wrap option 250
- preferences 254

- layout tables 242
 - alignment 250
 - bg color 250
 - cell padding 250
 - cell spacing 250
 - clearing heights 250
 - drawing 243
 - formatting 250
 - Make Widths Consistent option 250
 - nested 245
 - preferences 254
 - Remove all Spacers option 250
 - Remove Nesting option 250
- Layout view 241
 - about layout cells and tables 242
 - adding content to 246
 - autostretch 251
 - cell padding 250
 - cell spacing 250
 - Clear Height attributes 250
 - clearing cell heights 247
 - column header menu 252
 - drawing layout cells and tables 243
 - fixed width 251
 - formatting layout cells 250
 - formatting layout tables 250
 - grid 246
 - Make Widths Consistent option 250
 - moving layout cells and tables 248
 - nested layout tables 245
 - overview 241
 - preferences 254
 - Remove All Spacers option 250
 - Remove Nesting option 250
 - resizing layout cells and tables 248
 - selecting layout cells and tables 248
 - setting width 251
 - spacer image preferences 253
 - spacer images, preferences 252
 - switching into 241
- layout, planning. *See* Layout view, templates
- learning Dreamweaver 23
- Left alignment (image Property inspector) 300
- library items 431
 - adding to pages 425, 432
 - as assets. *See* assets
 - creating 431
 - deleting 434
 - editing 425, 432
 - editing behaviors in 434
 - making editable in documents 434
 - overview 431
- link 402
- Link Checker dialog box 466
- Link Color option (Page Properties) 115
- Link External Style Sheet dialog box 290
- link href 290
- Link to Existing File command 83
- Link to New File command 84
- linked documents, opening 467
- linking
 - anchors 406
 - documents 403
 - to document using point-to-file icon 404
 - to external CSS style sheet 290
 - to named anchor using point-to-file icon 406
- linking and navigation 399
- links
 - applying to selection 424, 425
 - cache file 409
 - changing frames with 265
 - changing sitewide 412
 - checking 465
 - fixing 466
 - hiding 532
 - in templates 437
 - opening source 410
 - Relative to Document option 403
 - Relative to Site Root option 403
 - removing 410
 - site map 410
 - targeting 402
 - to anchors 406
 - to document 402
 - to style sheets 289
 - updating 409
- List view in Property inspector 524
- list/menu objects, making dynamic 593
- lists
 - creating 277
- Live Data Settings dialog box 479
- Live Data window
 - about 488
 - Auto Refresh 479
 - described 476
 - missing files 478
 - providing expected parameters 479
 - URL parameters on toolbar 479, 488

- live objects
 - Master/Detail Page Set 600
 - Record Insertion Form 620
 - Record Update Form 623
 - Recordset Navigation Bar 530
 - Recordset Navigation Status 536
 - Local Files option 76
 - local folder 133
 - local sites
 - See* sites 59
 - localhost 135
 - locked regions
 - clicking in 439
 - logging out users 647
 - logical errors in JavaScript code 212
 - login pages 643
 - Loop option 389
 - looping timelines 392
- M**
- Mac OS color palette 114
 - Macromedia Director, creating Shockwave movies with 326
 - Macromedia Exchange 53, 557
 - Macromedia HomeSite 187
 - Macromedia JRun 131, 186
 - macros (creating commands) 123
 - maintaining state information 617
 - managing
 - assets. *See* assets
 - extensions 53
 - links 409
 - panel groups 44
 - sites 69
 - map tag 417
 - markers for invisible elements 116
 - Master/Detail Page Set live object 600
 - master/detail pages 613
 - Match Case option 295
 - media elements
 - inserting 320
 - Merge Cells command 235
 - Microsoft Excel files, importing 229
 - Microsoft Word
 - files, opening 112
 - Middle alignment (image Property inspector) 300
 - mock-ups 58
 - modifying
 - databases, using stored procedures 629
 - page properties 113
 - monitor size, resizing pages to fit 38
 - Move to Record behavior 532
 - Move to Specific Record behavior 615
 - movies, inserting 320
 - MPEG movies, as assets. *See* assets
 - multimedia. *See* media
 - multiuser systems 54
 - MySQL 171
- N**
- Named Anchor command 406
 - named anchors
 - creating 406
 - navigation bar
 - adding images to 415
 - creating 414
 - image states 414
 - modifying elements 416
 - navigation links for records 529
 - navigation planning for a site 58
 - nested templates 454
 - nesting 380
 - frames 260
 - layers 380
 - .NET Framework 130
 - Netscape Navigator plug-ins
 - playing in Document window 329
 - troubleshooting 330
 - Netscape Navigator 4 layer compatibility 380
 - New Document dialog box 108
 - new features in Dreamweaver 26
 - New From Template command 111
 - New from Template command 450
 - non-breaking space, inserting 273
 - null links
 - changing 412
 - creating 408
- O**
- objects
 - adding Design Notes 100
 - checkboxes and radio buttons 584
 - inserting with Insert bar 42
 - making dynamic 525
 - ODBC
 - viewing installed drivers 660
 - OLE DB 157
 - OLE DB Connection 146
 - OLE DB connection parameters 147
 - OLE DB providers 146
 - onBlur event 376
 - online forums 19

- Open Attached Template command 457
- Open Browser Window action 362
- Open command 112
- Open Linked Page command 467
- opening
 - existing documents. *See* documents 111
 - text files 111
- Opening linked documents 467
- opening non-HTML files 180
 - preferences 180
- opening Word HTML files 112
- operating systems, multiuser 54
- Optimize Image in Fireworks command 310
- optional region (templates) 436, 447
- Oracle Thin Driver 167
- orphaned files 465
- overlapping tags 190
- overview
 - of Dreamweaver 17
- P**
- page layout. *See* Layout view, templates
- page properties
 - titles, changing 113
- pages
 - background image 113
 - changing title 113
 - color 113
 - default text colors 115
 - delete 626
 - Design Notes, using with 97
 - detail 613
 - download time, estimated 467
 - insert 619
 - login 643
 - mock-ups 58
 - previewing in browsers 463
 - related 617
 - resizing to fit monitor 38
 - restricting access to 645
 - results 609
 - search 608
 - size 467
 - update 622
 - user registration 639
- palettes, color 114
- Panel groups 36
- panels
 - Answers 49
 - Assets panel 425
 - Assets panel, Templates category 426
 - Behaviors panel 350
 - docking 44
 - History panel 48
 - Launcher bar 39
 - opening and closing with Launcher bar 39
 - setting floating preferences 47
- paragraph 273
 - line break 273
- paragraphs
 - formatting 276
- parameter markers 568
- Parameter tag (ASP.NET) 681
- parameters (templates) 437
- Parameters dialog box 525
- Parameters tag (ASP.NET) 680
- passwords
 - checking during log-in 644
 - letting users choose 640
 - storing 640
- pasting
 - history steps 122
- paths
 - absolute 400
 - document-relative 400
 - root-relative 401
- personal spelling dictionary 294
- photographs 297
- PHP 131
 - database connections 171
 - Mac OS X 128
- physical path 158
- pixels, transparent, in background 113
- planning 55
 - assets 59
 - frames 257
- Play Sound action 363
- Play Timeline action 375
- playback head 389
- playing Flash objects 324
- plug-ins
 - checking 356
 - making dynamic 525
 - troubleshooting 330
- plugins
 - playing in Document window 329
- PNG images
 - as tracing image 118
 - uses for 297
- pop-up menu
 - creating 588

- Popup Message action 364
- positioning
 - code blocks 567
- preferences
 - Code Coloring 180
 - Code Format 177
 - Code Hints 179
 - Code Rewriting 179
 - dictionary for spell checking 294
 - External Editors 320
 - File Types/Editors 320, 321
 - floating panels 47
 - Fonts/Encoding 50
 - General 50
 - Insert bar 44
 - Launch and Edit 307
 - Layers 380
 - Layout View 254
 - overview 50
 - Panels 47
 - Preview Using Local Server (root-relative links) 402
 - Site 73
 - Status Bar 39
 - updating links 409
 - Validator 181
- Preload Images action 364
- Prevent Layer Overlaps command 387
- Preview in Browser command 463
- Preview Using Local Server command 402
- previewing in browsers
 - overview 463
- properties
 - changing with behaviors 355
 - column, row, and cell 232
 - displaying 44
 - document, setting 112
 - frame 264
 - frameset 264
 - layer 384
 - layout cell 250
 - layout tables 250
 - multiple layer 384
 - table 231
- Property inspector 36, 217
 - displaying 44
 - editing a recordset 489, 526
 - expanding 44
 - fixing broken links 467
 - List view 524
 - making HTML attributes dynamic 523
 - Standard view 523
 - proportion, maintaining 301
 - Put command 91
 - putting and getting files 90
 - putting files on a remote server 91
- Q**
 - Quick Tag Editor 218
 - QuickTime movies
 - as assets. *See* assets
 - inserting 329
- R**
 - radio buttons 584
 - radio buttons, making dynamic 595
 - Radio Group dialog box 585
 - record counter, building a 535
 - record editing behaviors 619
 - Record Insertion Form live object 620
 - record navigation bar
 - creating 529
 - hiding 532
 - Record Path of Layer command 392
 - Record Update Form live object 623
 - recording commands 124
 - records
 - building a counter 535
 - deleting 626
 - displaying more than one 533
 - inserting 619
 - navigation links 529
 - updating 622
 - recordset
 - about 502
 - caching 516
 - columns, defining (SQL) 665
 - copying and pasting 517
 - Database Items tree 506
 - defining without SQL 503
 - editing or deleting 516
 - filtering records (SQL) 665, 668
 - in a detail page 604
 - in a results page 609
 - joining tables (SQL) 669
 - limiting records returned (SQL) 665
 - Recordset dialog box (simple) 504
 - writing SQL 663
 - Recordset dialog box
 - advanced 504
 - simple 504

- Recordset Navigation Bar live object 530
- Recordset Navigation Status live object 536
- recordsets
 - simple, creating 503
 - SQL, writing custom statements 504
- reference material 29
- Reference panel 200
- Refresh Local command 84
- refreshing
 - Site list (Assets panel) 422
- regions
 - hiding 532
- regions, locked
 - clicking in 439
- registration page 639
- regular expressions 199
- related pages 617
- Remote Files option 76
- remote folder 63, 133
 - troubleshooting 65
- Remove Frame command 390
- Remove Timeline command 393
- removing frames from a timeline 392
- removing rows and columns 234
- Rename Panel Groups command 46
- Repeat Region behavior 533
- repeating region
 - adding entry 453
 - deleting entry 453
 - editing region 453
- repeating region (templates) 436, 443
 - alternating colors 445
 - creating 444
- repeating steps 120
- repeating table (templates) 444
- Replacing an image placeholder 299
- Replay button 120
- reports 101
 - Checked Out By 102
 - Workflow 102
- Reports command 468
- requirements
 - web applications 128
- Reset Origin command 117
- Reset Position command 118
- resizing
 - handles 301
 - layers 382
 - layout cells and tables 248
 - table cells 233
- Resizing an image 301
- resources
 - for information on web technologies 29
- restricting site access 639
- restricting tables 481
- results pages
 - going to a detail page 614
 - using a simple recordset 609
 - using an advanced recordset 611
- resultset, JSP 484
- return 273
- reusing
 - library items 432
 - searches 198
- Right alignment (image Property inspector) 301
- roadmap (where to start) 23
- Rollover Image text field 302
- rollovers 301
 - creating 302
 - making a link 302
- root-relative paths
 - about 401
 - Preview Using Local Server preference 402
 - setting 403
- Roundtrip HTML 190
- Rows option (Insert Table) 228
- rulers 117
- running the Tag Validator 205

S

- Save All Frames command 263
- Save command 111
- Save Frame As command 263
- Save Frame command 263
- Save Frameset As command 263
- Save Frameset command 263
- saving
 - documents 111
 - files in frames and framesets 263
 - searches 198
- schemas 481
- screen reader
 - Window Eyes 334
- screen readers
 - JAWS for Windows 334
- script links
 - changing 412
 - creating 408
- scripts
 - as assets. *See* assets
 - balanced braces 204

- editing external 221
 - entering 221
 - viewing functions 196
- scrolling lists 586
- Search For options 295
- search pages 608
- searching
 - finding and replacing 294
 - for files 294
 - Help 18
 - regular expressions 199
 - saving search patterns 198
 - tags and attributes 197
 - text between specific tags 197
 - text in HTML source 196
 - text within files 294
- Secure Shell 64
- security 639
- Select Newer Remote command 93
- Select ODBC DSN dialog box 162
- selecting
 - frames and framesets 261
 - layers 381
 - layout cells and tables 248
 - objects in the Document window 116
- Server Behavior Builder 558
- server behaviors
 - coding 190
 - coding guidelines 566
 - creating 558
 - creating dialog boxes for 568
 - deleting records 629
 - editing custom behaviors 571
 - editing records 619
 - going to a detail page 603, 614
 - going to a related page 618
 - hiding regions 532
 - inserting records 622
 - installing more 557
 - moving to a specific record 615
 - moving to records 532
 - repeating regions 533
 - testing 570
 - updating records 626
- server objects
 - application objects 510
 - ColdFusion variables 512
 - session objects 509
- servers
 - access options 63
 - defining remote sites 63
 - setup troubleshooting 65
- server-side image maps 416
- server-side includes 223
 - editing 224
 - inserting 224
- session variables 509
 - about 495
 - and form parameters 498
 - and URL parameters 497
 - data, retrieving 500
 - data, storing 498
- Set as Home Page command 84
- Set Color Scheme command 115
- Set Nav Bar Image action 365
- Set Text of Frame action 366
- Set Text of Layer action 366
- Set Text of Status Bar action 367
- Set Text of Text Field action 368
- setting
 - breakpoints 212
 - document properties 112
 - fonts and changing characteristics 274
- Sheets
- Shockwave movies
 - as assets. *See* assets
 - controlling 357
 - inserting 327
 - overview 326
- Shockwave objects, making dynamic 525
- shortcut menus. *See* context menus
- Show Launcher bar in Status Bar option 47
- Show Layout Table Tabs option 243
- Show Region behavior 532
- Show-Hide Layers action 368
- showing invisible elements 116
- site design 55
- Site Files view 78
- site list, refreshing 422
- site maps 80
 - adding files to a site 83
 - changing links in 411
 - Link to New File command 84
 - links 410
 - removing links in 411
 - saving as an image file 86
 - showing dependent files in 85
 - viewing a branch of 85

- Site panel 36, 70
 - browsing files 80
 - changing the display 76
 - columns 74
 - Get command 90
 - Put command 91
 - searching for files 79
 - searching for text and/or HTML within documents 294
 - Site Files view 71
 - Site Map view 71
 - site maps 80
 - toolbar options 71
 - viewing sites 76
- Site panel, *See* Site panel 70
- site planning 55
- Site preferences 73
- site root-relative paths. *See* root-relative paths
- Site window
 - searching for text 294
- sites
 - adding files and folders 78
 - browser compatibility 56, 462
 - cache 409
 - cache file 410
 - changing display 76
 - changing links sitewide 412
 - checking files in and out 89
 - checking links 465
 - defining for a dynamic site 132
 - design and planning 55
 - Design Notes, using with 97
 - editing 66
 - file cloaking 94
 - fixing broken links 466
 - getting reports 468
 - importing and exporting 86
 - large, assets in 428
 - links 399
 - local versus remote 59
 - local, creating, setting up 59
 - locating files in 79
 - planning navigation 58
 - planning structure 57
 - previewing in browsers 463
 - reference material 29
 - remote 63
 - remote, access options 63
 - remote, locating files in 79
 - remote, setup troubleshooting 65
 - removing from site list 87
 - reports, running 102
 - searching for files in 294
 - security 639
 - selecting updated files 78
 - structure and navigation 69
 - testing 461
- Sitespring 103
- slider controls, creating 358
- Snap to Web-Safe color palette 114
- Snippets panel 192
- SOAP and web services 549
- sound 327–329
 - adding to a page 327
 - playing 363
- source code
 - cleaning up 203
 - coloring preferences 180
 - copying and pasting from Fireworks to Dreamweaver 311
 - copying and pasting, general 195
 - CSS attributes conversion 293
 - editing with BBEdit 187
 - external editors 187
 - formatting in existing documents 178
 - formatting preferences overview 177
 - formatting preferences, setting 177
 - options 176
 - reference 200
 - reference material 29
 - rewriting preferences 179
 - searching 196
 - selecting in the Document window 116
 - tag searches 197
 - tag styles 285
 - updating Fireworks HTML placed in Dreamweaver 312
 - writing and editing 191, 194
- Source Code option 196
- source control 88
- spacer images
 - preferences 253
- spacing, inserting non-breaking 273
- special characters
 - inserting 278
 - line break 273
- spell check 294
- spelling
 - checking 294
 - dictionaries 294

- Split Cells command 236
 - Split Frame commands 259
 - SQL 663
 - ASP.NET 506
 - columns, defining 665
 - Database Items tree 506
 - DELETE 663
 - filtering records 665, 668
 - FROM 664
 - GROUP BY 664
 - INSERT 663
 - joining tables 669
 - limiting records 665
 - operators 664
 - ORDER BY 664
 - recordset, defining with SQL 504
 - SELECT statement 663
 - UPDATE 663
 - WHERE 664
 - SQL Server Connection 146
 - SSH 64
 - stacking order
 - changing with timelines 393
 - layers 385
 - Standard toolbar 36, 41
 - Standard view in Property inspector 523
 - starting out 23
 - status bar 37
 - preferences 39
 - setting text (behavior) 367
 - Window Size pop-up menu 37
 - stepping through code 213
 - Stop Timeline action 375
 - Stored Procedure (ColdFusion) option 629
 - stored procedures
 - creating a stored procedure object 629
 - modifying databases 629
 - Strikethrough (Default Color) button 114
 - style sheets
 - Edit Style Sheet dialog box 291
 - editing 290
 - editing external style sheets 290
 - external 289
 - overview 285
 - See also* styles
 - Style styles
 - Style submenu 274
 - About Cascading styles
 - See also* style sheets
 - styles 285
 - applying custom styles 289
 - conflicting 292
 - conflicting styles 292
 - converting to HTML 293
 - CSS to HTML markup conversion table 293
 - HTML 279
 - See also* style sheets
 - Submit buttons 590
 - Sun JDBC-ODBC Bridge driver 168
 - support file types 190
 - Swap Image action 373
 - Swap Image Restore action 374
 - swatches, color 114
 - synchronizing local and remote sites 93
 - syntax errors 211
 - System Color button 114
 - system color picker 114
- T**
- table cells
 - cutting, copying, and pasting 237
 - formatting 232
 - merging 235
 - splitting 235
 - See also* layout cells, tables
 - tables 590
 - adding and removing rows and columns 234
 - adding content to 228
 - adjusting rows and columns 233
 - cells, splitting 236
 - changing column width 234
 - clearing cell width and height 234
 - columns, rows, and cells 232
 - converting from layers 386
 - creating 228
 - exporting data 239
 - formatting 231
 - importing 229
 - importing tabular data 272
 - merging cells 236
 - nesting 237
 - overview 227
 - preset designs for 232
 - properties 231
 - resizing 233
 - restricting 481
 - selecting elements 229
 - sorting 239
 - See also* columns, rows, and cells
 - tabular data, importing 229

- tag
 - heading, applying 276
 - paragraph, applying 276
- Tag Chooser 193
- tag editors 194, 218
- Tag inspector 194
- tag libraries 181
- Tag Library Editor 181
- tag selector 36, 221
- tags
 - body 119
 - invalid 190
 - overlapping 190
 - searches 197
 - selecting 116
- tags, importing 184
- target audience 56
- targeting browsers
 - converting file to 3.0 compatible 387
- targeting frames to
 - _blank 266
 - _parent 266
 - _self 266
 - _top 266
- targeting links
 - in documents 403
 - opening document in a new window 403
- template parameters 437
- Template Properties dialog box 452
- templates 110, ??-460
 - applying to document 426
 - applying to existing document 456
 - as assets. *See* assets
 - Assets panel 426
 - checking syntax 458
 - clicking in locked regions 439
 - creating 440
 - creating editable regions 442
 - creating new documents with 111, 449
 - detaching document from 457
 - editable regions 436
 - editable tag attributes 437, 446
 - editable versus locked regions 436
 - editing 426
 - editing code outside of HTML tags 437
 - editing server scripts in documents 437
 - finding editable regions 450
 - links 437
 - making region noneditable 443
 - modifying properties 450
 - nesting 454
 - optional region 436, 447
 - overview 435
 - renaming 427
 - repeating region 436, 443
 - updating documents 457
 - updating Dreamweaver 4 457
 - XML 459
- testing for accessibility 343
- testing server 134
 - setting up 129
- testing server behaviors 570
- text
 - adding to a document 272
 - alignment 276
 - changing color of 277, 423
 - changing font combinations 275
 - default color in pages 115
 - external editors. *See* external editors
 - formatting 271, 273
 - formatting with HTML styles 279
 - importing from other documents 272
 - indenting 276
 - inserting 271
 - making dynamic 520
 - non-breaking space 273
 - outdenting 276
 - searching within documents 294
 - Text Color option (Page Properties) 115
- text editors
 - files created by 111
- text editors. *See* external editors
- text fields, setting text with behaviors 368
- text files
 - opening 111
- text-based HTML editors. *See* external editors
- TextTop alignment (image Property inspector) 300
- timelines
 - adding and removing frames 392
 - adding objects to 390
 - animation tips 395
 - attaching a behavior 352
 - changing image source file 393
 - changing layer properties 393
 - complex paths 392
 - controlling with behaviors 395
 - creating 390
 - keyframes 390
 - looping 392
 - modifying 392

- multiple 393
- playback head 389
- playing and stopping using behaviors 375
- playing automatically 392
- renaming 395
- Timelines panel 389
- titles
 - changing 113
- tld files 185
- toolbar
 - document titles, changing 113
- Top alignment (image Property inspector) 300
- tracing images 118
- transferring files, troubleshooting 91
- transparent pixels in background 113
- troubleshooting
 - clicking in locked regions 439
 - Navigator plug-ins 330
 - server errors 136
 - transferring files 91
- typographical conventions 26

U

- UDDI
 - public directories 548
 - site list, editing 556
- UltraDev 4 127
- UltraDev 4 ColdFusion connection 141
- underline 274
- undocking panels and panel groups 44
- Unmark Editable Region command 443
- Update Current Page command 433, 457
- Update custom tag (ASP.NET) 678
- Update HTML command 312
- update page, building 622
- Update Pages command 458
- Update Record behavior 626
- updating links 409
- updating records 622
- uploading files 90, 91
- URL prefix 134
- URLs
 - applying to selection 424, 425
 - as assets. *See* assets
 - creating URL assets 429
- URLs. *See* paths
- Use Regular Expressions option 295
- usemap attribute 417
- user names
 - checking during log-in 644
 - checking for uniqueness 642

- letting users choose 640
- storing 640
- Using an external image editor 302
- Using Design Time style sheets 291
- Using Reports to test a site 468
- Using the CSS Styles panel 286
- Using your HTML styles in other sites 284

V

- Validate Form action 376
- Validator 205
 - preferences 181
- variable list 214
- VBScript 221
- viewing
 - head content 119
 - invisible elements 116
 - template regions in Code view 440
 - templates in Code view 439
 - templates in Design view 438
- virtual directory 135
- virtual path 158
- Visited Links color option (Page Properties) 115
- visual guides
 - overview 117
 - rulers 117
 - tracing images 118

W

- web applications
 - requirements 128
- web design, levels of experience in 23
- web hosting services 158
- web photo albums, creating 313
- web server
 - setting up 129
- web services
 - about 547
 - adding to a page 554
 - AXIS proxy generator 550
 - proxy generators, additional 550
 - proxy generators, configuring 551
 - proxy generators, installing 550
 - SOAP 549
 - UDDI directories 548
 - UDDI site list, editing 556
 - workflow 549
- web.xml 186
- Welcome window 36
- Window Size pop-up menu 37
- Window Sizes option 39

windows

See also inspectors, panels

Windows OS color palette 114

word processors, files created by 111

workflow

for dynamic pages 483–489

Workflow reports 102

working environments

Live Data window 476

workspace 31

about 33

floating layout 35

layout 32

writing code (Code view or Code inspector) 191, 194

writing code blocks 558

X

XHTML, making pages compliant 205

XML 205, 458

in templates 459

overview 458

tag notations when exporting 460

XML DTD files 185

Z

Z-Index option (for layers)

changing stacking order 385

