MX

macromedia®

DREAMWEAVER® MX 2004

Using Dreamweaver

Trademarks

Add Life to the Web, Afterburner, Aftershock, Andromedia, Allaire, Animation PowerPack, Aria, Attain, Authorware, Authorware Star, Backstage, Bright Tiger, Clustercats, Cold Fusion, Contribute, Design in Motion, Director, Dream Templates, Dreamweaver, Drumbeat 2000, EDJE, EJIPT, Extreme 3D, Fireworks, Flash, Fontographer, FreeHand, Generator, HomeSite, JFusion, JRun, Kawa, Know Your Site, Knowledge Objects, Knowledge Stream, Knowledge Track, LikeMinds, Lingo, Live Effects, MacRecorder Logo and Design, Macromedia, Macromedia Action!, Macromedia Flash, Macromedia M Logo & Design, Macromedia Spectra, Macromedia xRes Logo and Design, MacroModel, Made with Macromedia, Made with Macromedia Logo and Design, MAGIC Logo and Design, Mediamaker, Movie Critic, Open Sesame!, Roundtrip HTML, Shockwave, Sitespring, SoundEdit, Titlemaker, UltraDev, Web Design 101, what the web can be, and Xtra are either registered or 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.

Third-Party Information

This guide contains links to third-party websites 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 website 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.

Third Party Software Notices and/or Additional Terms and Conditions can be found at www.macromedia.com/go/thirdparty/.

Opera ® browser Copyright © 1995-2002 Opera Software ASA and its suppliers. All rights reserved.

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 © 2003 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. Part Number ZDW70M200

Acknowledgments

Senior Management: Sheila McGinn

Project Management: Charles Nadeau

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

Editing: Lisa Stanziano, Mary Ferguson, Noreen Maher

Production Management: Patrice O'Neill

Media Design and Production: Adam Barnett, Aaron Begley, Chris Basmajian, John Francis, Jeff Harmon

Special thanks to Jay London, Alain Dumesny, Lori Hylan-Cho, Jack Herrington, Vic Mitnick, Rebecca Hyatt, Russ Helfand, Erik Bergman, Luciano Arruda, Sho Kuwamoto, Ken Karleskint, Scott Richards, Bonnie Loo, David Deming, Jennifer Taylor, Rosana Francescato, Randy Nielsen, the beta testers, and the entire Dreamweaver engineering and QA teams.

First Edition: September 2003

Macromedia, Inc.
600 Townsend St.
San Francisco, CA 94103

CONTENTS

PART I: Dreamweaver Basics

INTRODUCTION: Welcome to Dreamweaver 17

What's new in Dreamweaver MX 2004 18
Where to start 20
Dreamweaver workflow for creating websites 22
Using Dreamweaver with other applications 25
Dreamweaver and accessibility 26
Guide to Dreamweaver instructional media 27
Typographical conventions 29
HTML and web technologies resources 29

CHAPTER 1: Exploring the Workspace 31

About the Dreamweaver workspace 32
Working in the Document window 39
Using toolbars, inspectors, and context menus 41
Using panels and panel groups 46
Using Dreamweaver accessibility features 49
Optimizing the workspace for accessible page design 53
Using visual guides in Dreamweaver 54
Dreamweaver customizing basics 54

CHAPTER 2: Setting Up a Dreamweaver Site 59

About Dreamweaver sites 59
Setting up a new Dreamweaver site 62
Using the Advanced settings to set up a Dreamweaver site 63
Editing settings for a Dreamweaver site 67
Editing existing websites in Dreamweaver 67

CHAPTER 3: Creating and Opening Documents 69
Creating new documents 70
Saving a new document 72
Setting a default new document type 72
Opening existing documents 73
Importing a Microsoft Word HTML file 74

PART II: Working with Dreamweaver Sites

CHAPTER 4: Managing Your Files 77
About site management 78
Accessing sites, a server, and local drives 80
Viewing files and folders 83
Managing files and folders in the Files panel 86
Working with a visual map of your site 89
Importing and exporting sites 96
Removing a Dreamweaver site from your list of sites 97
Checking in and checking out files 97
Getting and putting files to and from your server 101
Synchronizing the files on your local and remote sites 103
Cloaking folders and files in your site 104
Storing file information in Design Notes 108
Testing your site 111

CHAPTER 5: Managing Site Assets and Libraries 115
About library items 115
Working with assets 116
Creating and managing a list of favorite assets 123
Working with library items 125

CHAPTER 6: Managing Contribute Sites with Dreamweaver 133
About managing Contribute sites 134
Preparing a site for use with Contribute 138
Administering a Contribute site using Dreamweaver 138
Troubleshooting a Contribute site 140
Managing Contribute files using Dreamweaver 141

PART III: Laying Out Pages

CHAPTER 7: Laying Out Pages with CSS 145

About layers in Dreamweaver 146
Inserting content blocks for layout 147
Changing highlight color for content blocks 148
Working with content blocks for layout. 149
Inserting a layer 150
Setting layer preferences and properties 152
Managing layers 153
Manipulating layers 157
Converting layers to tables. 159
Using rulers and grids to lay out pages. 160
Using a tracing image 161

CHAPTER 8: Presenting Content with Tables 163

About tables 164
Inserting a table and adding content 165
Importing and exporting tabular data 166
Selecting table elements 167
Using Expanded Tables mode for easier table editing. 170
Formatting tables and cells 170
Resizing tables, columns, and rows 173
Adding and removing rows and columns 176
Splitting and merging cells 178
Copying, pasting, and deleting cells. 179
Nesting tables 181
Sorting tables. 181

CHAPTER 9: Laying Out Pages in Layout Mode 183

About Layout mode 184
Switching from Standard to Layout mode 187
Drawing in Layout mode 187
Adding content to a layout cell 190
Clearing automatically set cell heights 192
Resizing and moving layout cells and tables. 192
Formatting layout cells and tables 194
Setting column width 195
Setting preferences for Layout mode 197

CHAPTER 10: Using Frames. 199

About frames and framesets 200
Working with framesets in the Document window 203
Creating frames and framesets 204
Selecting frames and framesets. 207
Opening a document in a frame 209
Saving frame and frameset files 209

Viewing and setting frame properties and attributes 210
Viewing and setting frameset properties 212
Controlling frame content with links. 213
Handling browsers that can't display frames. 214
Using JavaScript behaviors with frames 214

CHAPTER 11: Managing Templates 215

About Dreamweaver templates 216
Creating a Dreamweaver template 226
Creating templates for a Contribute site 229
Creating editable regions 232
Creating repeating regions 234
Using optional regions. 237
Defining editable tag attributes 239
Creating a nested template 241
Editing and updating templates. 242
Managing templates 245
Exporting and importing template XML content 246
Exporting a site without template markup. 247
Applying or removing a template from an existing document 247
Editing content in a template-based document 248

PART IV: Adding Content to Pages

CHAPTER 12: Working with Pages. 253

About working with pages 253
Saving web pages 257
Specifying HTML instead of CSS 258
Setting page properties 258
Working with colors 260
Selecting elements in the Document window 262
Using the History panel. 263
Automating tasks. 265
Using JavaScript behaviors to detect browsers and plug-ins 270
Previewing and testing pages in browsers 270
Setting download time and size preferences 272

CHAPTER 13: Inserting and Formatting Text 273

About text formatting in Dreamweaver 273
Inserting text 280
Formatting paragraphs and document structure 282
Formatting text 285
Using Cascading Style Sheets styles 290
Updating CSS style sheets in a Contribute site 300
Checking spelling 301
Searching for and replacing text. 301

CHAPTER 14: Inserting Images 303
About images 303
Inserting an image 305
Resizing an image 309
Cropping an image 310
Optimizing an image using Fireworks 310
Adjusting the brightness and contrast of an image 311
Sharpening an image 311
Creating a rollover image 312
Using an external image editor 313
Applying behaviors to images 313

CHAPTER 15: Linking and Navigation 315
Understanding document locations and paths 316
Jump menus 318
Navigation bars 319
About image maps 320
Creating links 320
Managing links 327
Inserting jump menus 331
Using navigation bars 333
Using image maps 335
Attaching JavaScript behaviors to links 336
Checking for broken, external, and orphaned links 336
Fixing broken links 338
Opening linked documents in Dreamweaver 339

CHAPTER 16: Working with Other Applications 341
About Fireworks and Flash integration 341
Optimizing your work environment for Fireworks and Flash 342
Working with Fireworks 342
Working with Flash 351

CHAPTER 17: Adding Audio, Video, and Interactive Elements 355
About media files 356
Inserting and editing media objects 358
Starting an external editor for media files 359
Using Design Notes with media objects 360
Inserting and modifying a Flash button object 361
Inserting a Flash text object 363
Inserting Flash content 363
Downloading and installing Flash elements 364
Inserting Flash elements 364
Editing Flash element attributes 365
Inserting Shockwave movies 365
Adding video 365
Adding sound to a page 366

Inserting Netscape Navigator plug-in content 367
Inserting an ActiveX control 368
Inserting a Java applet 368
Using behaviors to control media 369

CHAPTER 18: Using JavaScript Behaviors. 371

Using the Behaviors panel 372
About events 372
Applying a behavior. 373
Attaching a behavior to text. 374
Changing a behavior 375
Updating a behavior 375
Creating new actions 375
Downloading and installing third-party behaviors 376
Using the behavior actions that come with Dreamweaver 376

PART V: Working with Page Code

CHAPTER 19: Setting Up Your Coding Environment 401

Viewing your code. 401
Using the coder-oriented workspace (Windows only) 402
Setting viewing preferences 403
Setting coding preferences 403
Customizing keyboard shortcuts 405
Opening files in Code view by default 405
Setting Validator preferences 405
Managing tag libraries 406
Importing custom tags into Dreamweaver 410
Using an external HTML editor with Dreamweaver 412

CHAPTER 20: Coding in Dreamweaver 415

About coding in Dreamweaver 416
Writing and editing code. 423
Searching and replacing tags and attributes 428
Making quick changes to a code selection 430
Using language-reference material 431
Printing your code. 432

CHAPTER 21: Optimizing and Debugging Your Code 433

Cleaning up your code 433
Verifying that tags and braces are balanced 434
Checking for browser compatibility. 434
Validating your tags 437
Making pages XHTML-compliant 438
Using the ColdFusion debugger (Windows only) 439

CHAPTER 22: Editing Code in Design View 441
Editing code with the Property inspector 441
Changing attributes with the Tag inspector 442
Editing code with the Quick Tag Editor 443
Editing code with the tag selector 446
Editing scripts 446
Working with server-side includes 448
Using JavaScript behaviors 449
Viewing and editing head content 449

PART VI: Preparing to Build Dynamic Sites

CHAPTER 23: Setting Up a Web Application 453
What you need to build web applications 453
Setting up a web server 454
Setting up an application server 454
Creating a root folder for the application 458
Defining a Dreamweaver site 459
Connecting to a database 461

CHAPTER 24: Database Connections for ColdFusion Developers 463
Connecting to a database 463
Editing or deleting a database connection 464

CHAPTER 25: Database Connections for ASP.NET Developers 465
Connecting to a database 465
Editing or deleting a database connection 468

CHAPTER 26: Database Connections for ASP Developers 469
About database connections in ASP 469
Creating a DSN connection 471
Creating a DSN-less connection 474
Connecting to a database on an ISP 475
Editing or deleting a database connection 478

CHAPTER 27: Database Connections for JSP Developers 479
About database connections in JSP 479
Connecting to a database 481
Connecting through an ODBC driver 482
Editing or deleting a database connection 485

CHAPTER 28: Database Connections for PHP Developers 487
Connecting to a database. 487
Editing or deleting a database connection 488

CHAPTER 29: Troubleshooting Database Connections 489
Troubleshooting permissions problems 489
Troubleshooting Microsoft error messages 491

PART VII: Making Pages Dynamic

CHAPTER 30: Optimizing the Workspace for Visual Development. 497
Displaying web-application development panels 497
Viewing your database within Dreamweaver 499
Viewing live data in Design view 500
Working in Design view without live data 505
Previewing dynamic pages in a browser 505
Restricting database information displayed in Dreamweaver 506

CHAPTER 31: The Workflow for Dynamic Page Design 507
Designing the page 508
Creating a source of dynamic content 508
Adding dynamic content to a web page 510
Enhancing the functionality of a dynamic page 510
Testing and debugging the page. 511

CHAPTER 32: Obtaining Data for Your Page. 515
Using a database to store content. 515
Collecting data submitted by users 516
Accessing data stored in session variables 520

CHAPTER 33: Defining Sources of Dynamic Content 525
About dynamic content sources. 526
Defining a recordset 530
Defining URL parameters 534
Defining form parameters 535
Defining session variables 536
Defining application variables for ASP and ColdFusion 537
Using a variable as a data source for a ColdFusion recordset 538
Defining server variables 538
Caching content sources 542
Changing or deleting content sources 543
Copying a recordset from one page to another page. 543

CHAPTER 34: Adding Dynamic Content to Web Pages 545
About adding dynamic content 546
Making text dynamic 547
Making images dynamic 548
Making HTML attributes dynamic 549
Making ActiveX, Flash, and other object parameters dynamic 551
Editing dynamic content 552
Deleting dynamic content 552
Creating dynamic pages in a Contribute site 553

CHAPTER 35: Displaying Database Records 555
About displaying database records 555
Using predefined data formats 562
Creating recordset navigation links 563
Showing and hiding regions based on recordset results 566
Displaying multiple recordset results 566
Creating a table with a Repeat Region server behavior 567
Creating a record counter 569
Creating ASP.NET DataGrid and DataList web controls 570

CHAPTER 36: Using ColdFusion Components 573
About ColdFusion components 573
Visually creating a component in Dreamweaver 574
Viewing ColdFusion components in Dreamweaver 575
Editing ColdFusion components in Dreamweaver 576
Building web pages that use ColdFusion components 577

CHAPTER 37: Using Web Services 579
About web services 580
Configuring proxy generators for use with Dreamweaver 583
Adding a web service proxy using the WSDL description 585
Adding a web service to a page 586
Editing the UDDI web service site list 588

CHAPTER 38: Adding Custom Server Behaviors 589
About custom server behaviors 589
Installing third-party server behaviors 598
Using the Server Behavior Builder 598
Using parameters in server behaviors 601
Positioning code blocks 602
Creating a dialog box for a custom server behavior 603
Editing and modifying server behaviors 605

CHAPTER 39: Creating Forms 607
About forms 607
Creating HTML forms 611
Inserting HTML form objects 613
Inserting dynamic HTML form objects 617
Validating HTML form data 620
Attaching JavaScript behaviors to HTML form objects 620
Attaching custom scripts to HTML form buttons 621
Creating accessible HTML forms 622
Creating ASP.NET forms 623

PART VIII: Developing Applications Rapidly

CHAPTER 40: Building ColdFusion Applications Rapidly 629
About rapid application development (all servers) 629
Building master/detail pages (ColdFusion) 634
Building search/results pages (ColdFusion, ASP, JSP, PHP) 641
Building a record insert page (all servers) 646
Building pages to update a record (ColdFusion) 649
Building pages to delete a record (ColdFusion) 655
Using stored procedures to modify databases (ColdFusion) 665
Building pages that restrict access to your site (ColdFusion, ASP, JSP, PHP) 666

CHAPTER 41: Building ASP.NET Applications Rapidly 675
Building master/detail pages (ASP.NET) 675
Building a database search page (ASP.NET) 681
Building a record insert page (ASP.NET) 685
Building pages to update a record (ASP.NET) 686
Building pages to delete a record (ASP.NET) 692
Using stored procedures to modify databases (ASP.NET) 700
Building pages that restrict access to your site (ASP.NET) 701

CHAPTER 42: Building ASP and JSP Applications Rapidly 703
Building master/detail pages (ASP and JSP) 703
Building search/results pages (ASP and JSP) 706
Building a record insert page (ASP and JSP) 706
Building pages to update a record (ASP and JSP) 707
Building pages to delete a record (ASP and JSP) 711
Building pages with advanced data manipulation objects (ASP and JSP) 714
Building pages that restrict access to your site (ASP and JSP) 718

CHAPTER 43: Building PHP Applications Rapidly 719
Building master/detail pages (PHP) 719
Building search/results pages (PHP) 725
Building a record insert page (PHP) 725
Building pages to update a record (PHP) 726
Building pages to delete a record (PHP) 732
Building pages that restrict access to your site (PHP) 738

PART IX: Appendixes

APPENDIX A: Beginner's Guide to Databases 741
About databases 741
Database design basics 742
Understanding database connections 747

APPENDIX B: SQL Primer 753
Syntax basics 753
Defining the columns of a recordset 755
Limiting the records in a recordset 755
Sorting the records in a recordset 758
Joining tables 759

INDEX 761

PART I
Dreamweaver Basics

Learn how to use Macromedia Dreamweaver MX 2004 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:

Introduction, “Welcome to Dreamweaver,” on page 17

Chapter 1, “Exploring the Workspace,” on page 31

Chapter 2, “Setting Up a Dreamweaver Site,” on page 59

Chapter 3, “Creating and Opening Documents,” on page 69

INTRODUCTION
Welcome to Dreamweaver

Macromedia Dreamweaver MX 2004 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 or another graphics application, then importing them directly into Dreamweaver, or by adding Macromedia Flash objects.

Dreamweaver also provides a full-featured coding environment that includes code-editing tools (such as code coloring and tag completion) and reference material on HTML, Cascading Style Sheets (CSS), JavaScript, ColdFusion Markup Language (CFML), Microsoft Active Server Pages (ASP), and JavaServer Pages (JSP). Macromedia Roundtrip HTML technology imports your hand-coded HTML documents without reformatting the code; you can then reformat code with your preferred formatting style.

Dreamweaver also enables you to build dynamic database-backed web applications using server technologies such as CFML,ASP.NET, ASP, 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:

- "What's new in Dreamweaver MX 2004" on page 18
- "Where to start" on page 20
- "Dreamweaver workflow for creating websites" on page 22
- "Using Dreamweaver with other applications" on page 25
- "Dreamweaver and accessibility" on page 26
- "Guide to Dreamweaver instructional media" on page 27
- "Typographical conventions" on page 29
- "HTML and web technologies resources" on page 29

What's new in Dreamweaver MX 2004

Dreamweaver MX 2004 contains a new, streamlined interface, along with improved product performance. There's also a wide variety of new features that improve usability and help you build pages whether you are working in the design environment or the coding environment.

Streamlined design and development interface

The Dreamweaver interface is more approachable to help you improve your productivity and quality of work.

Insert bar improvements give the Insert bar a new streamlined look, which takes up less space in the workspace. There's also a new Favorites category, which you can use to customize an Insert bar with the objects you use most often. For more information, see "The Insert bar" on page 37.

Microsoft Word and Microsoft Excel copy and paste commands enable you to copy and paste a Microsoft Word or Excel document directly into Dreamweaver. When you paste a Word or Excel document, Dreamweaver preserves formatting and generates quality HTML. For more information, see "Copying and pasting text from MS Office documents" on page 281.

Table editing visual feedback enables you to see the effects column resize operations will have on your tables. Visual feedback also makes it easier for you to select table elements. For more information about resizing, see "Resizing tables, columns, and rows" on page 173. For more information about using Expanded Tables mode for selecting elements, see "Using Expanded Tables mode for easier table editing" on page 170.

User interface overhaul maximizes the usable workspace, showing context and focus more clearly, and makes the interface more approachable and logical. For more information about the Dreamweaver workspace, see "About the Dreamweaver workspace" on page 32.

The Start page enables you to access recently used files, create new files, and access Dreamweaver resources. The Start page appears when you start Dreamweaver or when you don't have any documents open. For information hiding or showing the Start page, see "Hiding and displaying the Start page" on page 55.

Saved desktop option gives you the choice to have Dreamweaver reopen documents you were last working on when you restart Dreamweaver. For information about setting this option, see "Setting General preferences for Dreamweaver" on page 56.

Full Unicode support means that Dreamweaver in Windows supports all text encodings supported by Internet Explorer. You can use almost any language font installed on your system in Dreamweaver, and Dreamweaver will render and save it properly. For information about setting font encodings, see "Setting Fonts preferences for Dreamweaver display" on page 56.

Secure FTP enables you to fully encrypt all file transfers and prevent unauthorized access to your data, files, user names, and passwords. For information about setting up a remote connection using FTP, see "Setting up a remote folder" on page 65.

Modernized page layout and design environment

Dreamweaver includes the following enhanced CSS features that provide a more sophisticated way to style and add interactivity to designs, and also includes feedback to improve visual editing.

Dynamic cross-browser validation automatically checks the current document for cross-browser compatibility issues when you save the document. You specify which browsers to target, and Dreamweaver checks to make sure pages are not using tags or CSS constructs that those browsers do not support. For more information, see "Checking for browser compatibility" on page 434.

Relevant CSS tab displays the CSS rules applied to the current Code view or Design view selection. Click any rule to see which attributes it specifies (border style, margin, padding, text size, and so on). Special markings let you know when a specific attribute has been overridden, and in-place editing allows for quick edits that are instantly reflected in Design view. For more information, see "The CSS Properties tab" on page 279.

CSS layout visualization makes laying out pages in CSS much easier. You can easily select `div` and other content blocks in Design view, then modify properties with the CSS Properties tab. For more information, see Chapter 7, "Laying Out Pages with CSS," on page 145.

Improved CSS rendering means Dreamweaver can render more complicated CSS-based layouts and designs more accurately. Improved CSS rendering enables you to design more sophisticated layouts and use Dreamweaver design tools for robust visual manipulations.

Improved CSS Styles panel provides more options for editing styles in the current document. You can also now directly apply styles from the text Property inspector. For more information, see "Using Cascading Style Sheets styles" on page 290.

CSS-based text Property inspector lists all of the styles available for use in text, and includes a preview that shows what text will look like after the style is applied. For more information, see "The Property inspector and text formatting" on page 277.

CSS-based page properties give you more control of page properties, such as the appearance of headings and links, using modern CSS coding constructs. For more information, see "About setting page properties" on page 254.

Integrated image-editing toolbar enables you to do basic image manipulation and editing from Dreamweaver using Macromedia Fireworks technology. You can crop, resize, resample, and so on, without leaving Dreamweaver. For more information, see "Editing images in Dreamweaver" on page 304.

Powerful and open coding environment

Dreamweaver offers the following new features for coders, including improvements to the Code view and the ability to edit files without creating a Dreamweaver site. Dreamweaver also includes current support for server technologies.

Improved Tag inspector displays a list of properties available for the current selection and enables you to make quick, comprehensive edits. For more information, see "Changing attributes with the Tag inspector" on page 442.

Siteless file editing enables you to work directly on an FTP or RDS server without creating a Dreamweaver site. For more information about setting up Dreamweaver to work on a server, see "Setting up Dreamweaver to work without defining a site" on page 82. For more information about setting up a Dreamweaver site, see "Setting up a new Dreamweaver site" on page 62.

PHP server behaviors enable you to create a master/detail page set and user authentication pages. For more information, see "Building master/detail pages (PHP)" on page 719 and "Building pages that restrict access to your site (PHP)" on page 738.

Improved support for ASP.NET form controls provides new and improved ways, including new Property inspectors, to build and manipulate ASP.NET web forms in Design view. For more information, see Chapter 42, "Building ASP and JSP Applications Rapidly," on page 703.

Code view context menu enables you to make quick formatting changes to selected code. For more information, see "Making quick changes to a code selection" on page 430.

Where to start

Dreamweaver documentation 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 information about Dreamweaver resources, see "Guide to Dreamweaver instructional media" on page 27.

Web-design novices

If you are relatively new to web design, this section will point you to sections of the Dreamweaver documentation that are most appropriate for your background.

For web-design novices:

1. Begin by reading the tutorials for creating a static site in Getting Started Help.
2. In Using Dreamweaver Help (Help > Using Dreamweaver), read Chapter 1, "Exploring the Workspace," on page 31, Chapter 2, "Setting Up a Dreamweaver Site," on page 59, Chapter 4, "Managing Your Files," on page 77, and Chapter 3, "Creating and Opening Documents," on page 69.
3. Learn about page layout by reading Chapter 9, "Laying Out Pages in Layout Mode," on page 183.
4. To learn about formatting text and including images in your pages, read Chapter 13, "Inserting and Formatting Text," on page 273 and Chapter 14, "Inserting Images," on page 303.
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 Using Dreamweaver Help. You might want to wait to read the dynamic-page chapters until you're more familiar with creating web pages.

Experienced web designers

If you are an experienced web designer, this section will point you to sections of the Dreamweaver documentation that are most appropriate for your background. There are two different approaches: one for designers who are new to Dreamweaver and one for designers who are familiar with Dreamweaver but would like to learn more about creating dynamic pages.

For experienced web designers who are new to Dreamweaver:

1. Begin by reading the tutorials for creating a static site in Getting Started Help.
2. In Using Dreamweaver Help, read Chapter 1, "Exploring the Workspace," on page 31 to learn more about the Dreamweaver user interface.
3. Although much of the material in Chapter 2, "Setting Up a Dreamweaver Site," on page 59 and Chapter 4, "Managing Your Files," on page 77 is probably familiar to you, skim those chapters to see how these familiar concepts are implemented in Dreamweaver. Pay particular attention to the sections about setting up a Dreamweaver site.

4 For useful information on the details of using Dreamweaver to create basic HTML pages, read Chapter 13, "Inserting and Formatting Text," on page 273 and Chapter 14, "Inserting Images," on page 303.

5 For information about coding in Dreamweaver, see "Setting Up Your Coding Environment" on page 401, "Coding in Dreamweaver" on page 415, "Optimizing and Debugging Your Code" on page 433, and "Editing Code in Design View" on page 441.

6 Read the overview at the beginning of each of the other chapters in Using Dreamweaver Help 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 "Understanding Web Applications" and "Tutorial: Developing a Web Application" in Getting Started Help.

2 In Using Dreamweaver Help, skim Chapter 1, "Exploring the Workspace," on page 31 to learn about new aspects of the Dreamweaver user interface, then read "Optimizing the Workspace for Visual Development" on page 497.

3 Become familiar with the Dreamweaver workflow for dynamic pages by reading "The Workflow for Dynamic Page Design" on page 507.

4 Set up a web server and application server. (See Chapter 23, "Setting Up a Web Application," on page 453.)

5 Connect to a database. (See "Connecting to a database" on page 461.)

6 Read the overview at the beginning of each chapter in Using Dreamweaver Help to determine whether its topics are of interest to you.

7 If you're interested in customizing Dreamweaver by hand, read Customizing Dreamweaver on the Macromedia Support Center at www.macromedia.com/go/customizing_dreamweaver. If you want to write extensions for Dreamweaver, read Extending Dreamweaver Help.

Experienced hand-coders

If you are an experienced hand-coder, this section will point you to sections of the Dreamweaver documentation that are most appropriate for your background.

For experienced hand-coders:

1 Begin by reading "Tutorial: Editing Code" in Getting Started Help.

2 In Using Dreamweaver Help, read Chapter 1, "Exploring the Workspace," on page 31 to learn more about the Dreamweaver user interface.

3 Although much of the material in Chapter 2, "Setting Up a Dreamweaver Site," on page 59 and Chapter 4, "Managing Your Files," on page 77 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 setting up a Dreamweaver site.

4 Read more about coding with Dreamweaver in Chapter 19, "Setting Up Your Coding Environment," on page 401, Chapter 20, "Coding in Dreamweaver," on page 415, Chapter 21, "Optimizing and Debugging Your Code," on page 433, and Chapter 22, "Editing Code in Design View," on page 441.

5 Read the overview at the beginning of each chapter in Using Dreamweaver Help to determine whether its topics are of interest to you.

Web application developers

If you are a web application developer, this section will point you to sections of the Dreamweaver documentation that are most appropriate for your background. There are two different approaches, depending on whether you've used Dreamweaver before.

For web application developers who have not used Dreamweaver:

1. Begin by quickly reading Getting Started Help to familiarize yourself with the basics of using Dreamweaver.
2. In Using Dreamweaver Help, read Chapter 1, "Exploring the Workspace," on page 31 to learn more about the Dreamweaver user interface.
3. Although much of the material in Chapter 2, "Setting Up a Dreamweaver Site," on page 59 and Chapter 4, "Managing Your Files," on page 77 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 setting up a Dreamweaver site.
4. Set up a web server and application server using Dreamweaver. (See Chapter 23, "Setting Up a Web Application," on page 453.)
5. Connect to a database. (See "Connecting to a database" on page 461.)
6. Read the overview at the beginning of each of the chapter in Using Dreamweaver Help to determine whether its topics are of interest to you.

For experienced web application developers who have used Dreamweaver:

1. Begin by reading "What's new in Dreamweaver MX 2004" on page 18.
2. In Using Dreamweaver Help, skim Chapter 1, "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 www.macromedia.com/go/customizing_dreamweaver. If you want to write extensions for Dreamweaver, read Extending Dreamweaver Help.

Dreamweaver workflow for creating websites

There are many possible approaches to creating a website. The workflow presented in this documentation starts 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.

Planning your site

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.

- Use the Dreamweaver Files panel to set up your site's organizational structure. In the Files panel, you can easily add, delete, and rename files and folders to change the organization as needed. (See Chapter 2, "Setting Up a Dreamweaver Site," on page 59 and Chapter 4, "Managing Your Files," on page 77.)
- 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. (See Chapter 3, "Creating and Opening Documents," on page 69.)

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 "Checking in and checking out files" on page 97.)
- Use Design Notes to communicate with web team members. (See "Storing file information in Design Notes" on page 108.)

Managing your site files

The Dreamweaver Files and Assets panels make it easy for you to manage your site files.

- In the Dreamweaver Files 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 "Managing Your Files" on page 77.)
- 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 Chapter 5, "Managing Site Assets and Libraries," on page 115.)
- You can use Dreamweaver to manage aspects of your Contribute sites. (See Chapter 6, "Managing Contribute Sites with Dreamweaver," on page 133.)

Laying out web pages

Dreamweaver provides you flexibility as you mock-up and work toward a final layout for your pages. Choose the layout technique that works for you, or use the Dreamweaver layout options in conjunction with one another to create your site's look.

- You can use Dreamweaver layers or CSS positioning styles to create your layout. (See Chapter 7, "Laying Out Pages with CSS," on page 145.)
- The table tools and Layout mode in Dreamweaver let you quickly design web pages by drawing and then rearranging the page structure. (See Chapter 8, "Presenting Content with Tables," on page 163 and Chapter 9, "Laying Out Pages in Layout Mode," on page 183.)
- If you want to display multiple documents at once in a web browser, you can lay out documents using frames. (See Chapter 10, "Using Frames," on page 199.)
- Dreamweaver templates enable you to 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 11, "Managing Templates," on page 215.)

Adding content to pages

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.

- Dreamweaver page creation features enable you to specify web page properties such, as page titles, background images and colors. In addition, Dreamweaver provides tools to help you maximize website performance, and to test pages to ensure compatibility with different web browsers. (See Chapter 12, "Working with Pages," on page 253.)
- Type directly in a Dreamweaver document, or import text from other documents, then format the text using the Dreamweaver Property inspector. You can also easily create your own Cascading Style Sheets. (See Chapter 13, "Inserting and Formatting Text," on page 273.)
- Insert images, including rollover images, image maps, and Fireworks sliced images, and use alignment tools to position images in a page. You can also resize images directly in Dreamweaver. (See Chapter 14, "Inserting Images," on page 303 and Chapter 16, "Working with Other Applications," on page 341.)
- 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 15, "Linking and Navigation," on page 315.)
- Insert other types of media in a web page, such as Flash, Shockwave, and QuickTime movies, sound, and applets. (See Chapter 17, "Adding Audio, Video, and Interactive Elements," on page 355.)
- 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 18, "Using JavaScript Behaviors," on page 371.)

Hand-coding

Coding web pages by hand is another approach to creating pages. Dreamweaver provides easy-to-use visual editing tools, but it also provides a sophisticated coding environment; 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, and make sure that your code adheres to standards. (See Chapter 19, "Setting Up Your Coding Environment," on page 401 and Chapter 20, "Coding in Dreamweaver," on page 415, and Chapter 21, "Optimizing and Debugging Your Code," on page 433.)
- You can also use some Dreamweaver coding tools in Design view, the visual design environment. (See Chapter 22, "Editing Code in Design View," on page 441.)

Setting up a web application

Many 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 23, "Setting Up a Web Application," on page 453.)
- Connect to a database. (See "Connecting to a database" on page 461).

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 30, "Optimizing the Workspace for Visual Development," on page 497 and Chapter 31, "The Workflow for Dynamic Page Design," on page 507.)
- Define and display dynamic content on your pages. (See Chapter 32, "Obtaining Data for Your Page," on page 515, Chapter 33, "Defining Sources of Dynamic Content," on page 525, Chapter 34, "Adding Dynamic Content to Web Pages," on page 545, and Chapter 35, "Displaying Database Records," on page 555.)
- Encapsulate application or business logic using leading-edge technologies such as Macromedia ColdFusion components and web services. (See Chapter 36, "Using ColdFusion Components," on page 573 and Chapter 37, "Using Web Services," on page 579.)
- If you need more flexibility, you can create your own server behaviors and interactive forms. (See Chapter 38, "Adding Custom Server Behaviors," on page 589 and Chapter 39, "Creating Forms," on page 607.)

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. Provide security by restricting access to your pages. (See "Building ColdFusion Applications Rapidly" on page 629, "Building ASP.NET Applications Rapidly" on page 675, "Building ASP and JSP Applications Rapidly" on page 703, and "Building PHP Applications Rapidly" on page 719.)

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 412.
- You can specify preferred browsers for previewing your site. (See "Previewing and testing pages in browsers" on page 270.)
- You can start an external image editor, such as Macromedia Fireworks or Adobe Photoshop, from within Dreamweaver. (See "Using an external image editor" on page 313.)
- You can configure Dreamweaver to start a different editor for each file type. (See "Starting an external editor for media files" on page 359.)

- For information about adding interactivity to your site using Macromedia Flash, see "Inserting and modifying a Flash button object" on page 361, "Inserting a Flash text object" on page 363, or "Downloading and installing Flash elements" on page 364.
- To learn how to add animation to your site using Macromedia Shockwave movies, see "Inserting Shockwave movies" on page 365.
- For information about using ColdFusion, see ColdFusion help in Dreamweaver (Help > Using ColdFusion).

Dreamweaver and accessibility

Accessibility refers to making websites and web products usable for people with visual, auditory, motor, and other disabilities. 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.

Note: For more information about two significant accessibility initiatives, see the World Wide Web Consortium Web Accessibility Initiative (www.w3.org/wai) and Section 508 of the Federal Rehabilitation Act (www.section508.gov).

Dreamweaver provides tools that make the product accessible and tools that help you author accessible content:

Using Dreamweaver accessibility features For Dreamweaver web designers who need to use accessibility features, Dreamweaver offers screen reader support, keyboard navigation, and operating system accessibility support.

For more information, see "Using Dreamweaver accessibility features" on page 49.

Authoring for accessibility For Dreamweaver web designers who need to create accessible content, Dreamweaver assists you in creating accessible pages that contain useful content for screen readers and comply with government guidelines.

Dreamweaver provides dialog boxes that prompt you to enter accessibility attributes when you insert page elements (see "Optimizing the workspace for accessible page design" on page 53). For example, the accessibility dialog box for images reminds you to add text equivalents for graphics. Then, when the image appears on a page for a user with visual disabilities, the screen reader reads the description.

Dreamweaver also provides sample web pages that were designed for accessibility (see "Creating a document based on a Dreamweaver design file" on page 70) and an accessibility report that you can run to test your page or site against the Section 508 accessibility guidelines (see "Testing your site" on page 111).

Note: Remember that no authoring tool can automate the development process. Designing accessible websites requires you to understand accessibility requirements and make many ongoing subjective decisions about how users with disabilities interact with web pages. The best way to ensure that a website is accessible is through deliberate planning, development, testing, and evaluation.

The Dreamweaver accessibility validation 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 www.usablenet.com.

Guide to Dreamweaver instructional media

Dreamweaver has a variety of resources to help you learn the program quickly and become proficient in creating your own websites and pages:

- "Dreamweaver Help systems" on page 27
- "Macromedia website resources" on page 28
- "Additional books and training" on page 28

To get the most out of the Dreamweaver resources, see "Where to start" on page 20.

Note: For a list of HTML and web resources, see "HTML and web technologies resources" on page 29.

Dreamweaver Help systems

The Dreamweaver Help systems are accessible through the Help menu. All Dreamweaver Help appears in the Help viewer provided by your operating system: Microsoft HTML Help (Windows) or Apple Help (Macintosh).

Dreamweaver help includes the following help systems:

Getting Started Help 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 contains tutorials 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.

Using Dreamweaver Help provides comprehensive information about all Dreamweaver features, optimized for online reading.

Extending Dreamweaver Help provides information on the Dreamweaver Document Object Model.

Dreamweaver API Reference Help provides information on the APIs (application programming interfaces) that enable JavaScript and C developers to create extensions for Dreamweaver.

Related topics

- "Additional books and training" on page 28

Macromedia website resources

The Macromedia website contains the following resources to support Dreamweaver users:

The Dreamweaver Support Center website helps you get the most out of Dreamweaver. The Dreamweaver Support Center website at www.macromedia.com/go/dreamweaver_support/ 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.

Macromedia Developer Center at www.macromedia.com/go/devnet/ provides tools, tutorials, and more for all Macromedia products.

The Dreamweaver online forums give you an opportunity to discuss technical issues and share helpful hints with other Dreamweaver users. You'll find information on accessing the forums on the Macromedia website at www.macromedia.com/go/dreamweaver_newsgroup.

Keyboard shortcut charts show all the keyboard shortcuts in the default Dreamweaver configuration are available on the Macromedia website at www.macromedia.com/go/dreamweaver_mx_shortcuts.

Related topics

- "Dreamweaver Help systems" on page 27

Additional books and training

You can use additional books and training to become more proficient with Dreamweaver; you can also find Dreamweaver documentation in PDF and printed formats available online:

Macromedia Press books enable you to improve your Dreamweaver skills. Check out the latest books written by the experts on the Macromedia website at www.macromedia.com/go/dw2004_help_mmp.

Macromedia Dreamweaver Training and Certification provides hands-on tasks and real-world scenarios. Choose between instructor-led and online training, or combine them to create a learning path that is most effective for you. Learn more on the Macromedia website at www.macromedia.com/go/dreamweaver_training.

Dreamweaver PDFs are available on your product CD for the complete set of Dreamweaver documentation. Certain reference topics are not included in the PDF version of Using Dreamweaver; for more information on those topics, see Using Dreamweaver Help.

Dreamweaver manuals, not provided with the product, are available for purchase online. To purchase printed copies of Dreamweaver documentation, see the Macromedia website at www.macromedia.com/go/books_and_training/.

Related topics

- "Dreamweaver Help systems" on page 27
- "Macromedia website resources" on page 28

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.

For a complete list of Dreamweaver resources, see "Guide to Dreamweaver instructional media" on page 27.

HTML and web technologies resources

The following are some useful resources available on the web:

Note: For a list of Dreamweaver resources, see "Guide to Dreamweaver instructional media" on page 27.

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

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

Index DOT HTML (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 (www.w3.org/TR/xhtml1/) is the official specification for Extensible HyperText Markup Language.

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

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

Microsoft ASP.NET page (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 website (www.php.net/) provide information about PHP.

The MySQL site (www.mysql.com/) provides information about MySQL.

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

Macromedia ColdFusion product page (www.macromedia.com/go/coldfusion/) provides information about ColdFusion.

Macromedia JRun Server product page (www.macromedia.com/go/jrun/) provides information about the JRun Java application server.

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

Sun ONE product page (www.Chilisoft.com/products/) provides information about the Sun ONE ASP application server.

Web Services Demystified (www.sitepoint.com/article/692) explains what web services are and how they work.

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.

The CGI Resource Index (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 1
Exploring the Workspace

To get the most out of your Macromedia Dreamweaver MX 2004 experience, you should understand the basic concepts behind the Dreamweaver workspace and how to select options, use inspectors and panels, and set preferences that fit your work style best.

This chapter contains the following topics:

- "About the Dreamweaver workspace" on page 32
- "Working in the Document window" on page 39
- "Using toolbars, inspectors, and context menus" on page 41
- "Using panels and panel groups" on page 46
- "Using Dreamweaver accessibility features" on page 49
- "Optimizing the workspace for accessible page design" on page 53
- "Using visual guides in Dreamweaver" on page 54
- "Dreamweaver customizing basics" on page 54

Related topics

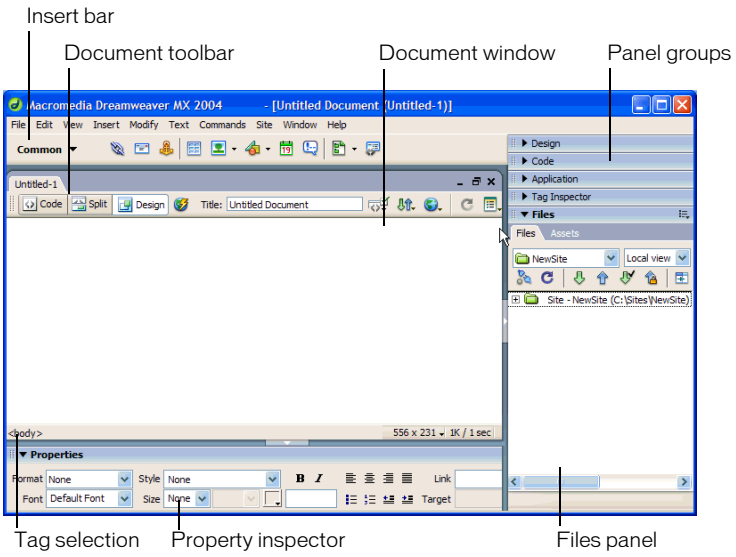
- "Optimizing the Workspace for Visual Development" on page 497

About the Dreamweaver workspace

The Dreamweaver workspace enables you to view documents and object properties. The workspace also places many of the most common operations in toolbars so that you can quickly make changes to your documents.

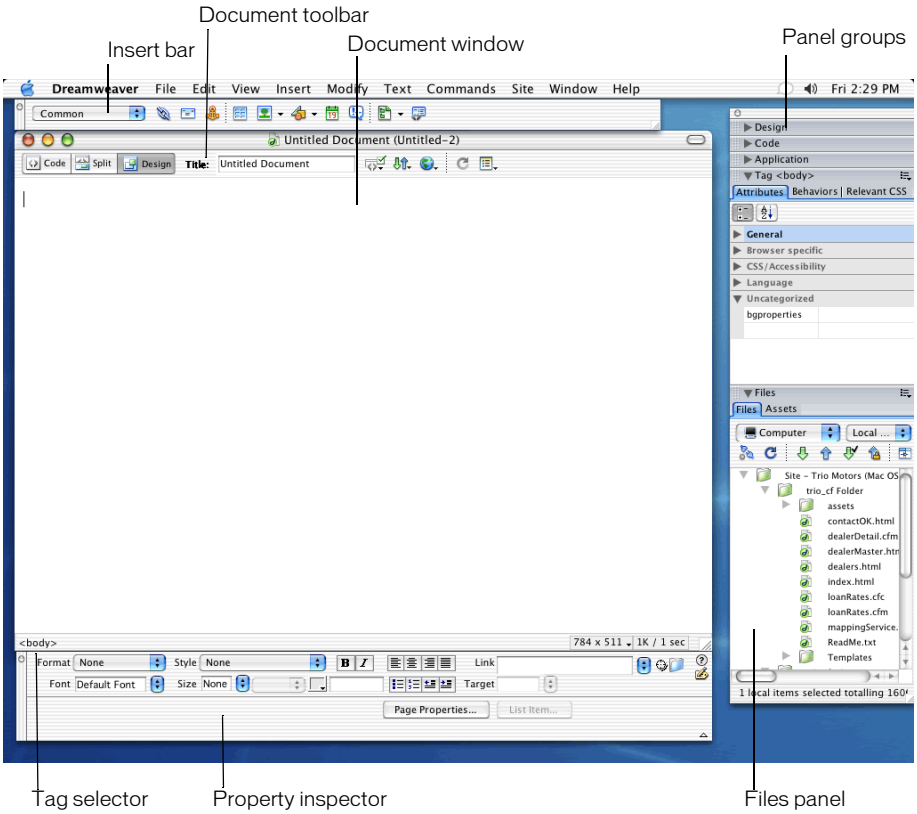
The workspace layout

In Windows, Dreamweaver provides an all-in-one-window integrated layout. In the integrated workspace, all windows and panels are integrated into a single larger application window.



Note: The Windows workspace also has a Coder option, which docks the panel groups on the left side and displays the Document window in Code view by default. For more information, see "Using the coder-oriented workspace (Windows only)" on page 402. To use this option, see "Choosing the workspace layout (Windows only)" on page 55.

On the Macintosh, Dreamweaver provides a floating workspace layout, in which each document is in its own individual window. Panel groups are initially docked together, but can be undocked into their own windows. Windows "snap" automatically to each other, to the sides of the screen, and to the Document window as you drag or resize them.



Related topics

- "Dreamweaver workspace elements" on page 33
- "Using toolbars, inspectors, and context menus" on page 41
- "Using panels and panel groups" on page 46.

Dreamweaver workspace elements

This section briefly describes some elements of the Dreamweaver workspace.

Note: Dreamweaver provides many other panels, inspectors, and windows. To open Dreamweaver panels, inspectors, and windows, use the Window menu. If you can't find a panel, inspector, or window that's marked as open, select Window › Arrange Panels to neatly lay out all open panels.

The Start page enables you to open a recent document or create a new document. From the Start page you can also learn more about Dreamweaver by taking a product tour or a tutorial.

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 enables you to set various attributes as you insert it. For example, you can insert a table by clicking the Table button 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 that provide options for 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, select 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. The Property inspector is not expanded by default in the Coder 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. Click any tag in the hierarchy to select that tag and all its contents.

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 Files panel enables you to manage your files and folders, whether they are part of a Dreamweaver site or on a remote server. The Files panel also enables you to access all the files on your local disk, much like Windows Explorer (Windows) or the Finder (Macintosh).

Related topics

- "Working in the Document window" on page 39
- "Using toolbars, inspectors, and context menus" on page 41
- "Using panels and panel groups" on page 46
- "Managing files and folders in the Files panel" on page 86

The Document window

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

Design view is 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 500).

Code view is a hand-coding environment for writing and editing HTML, JavaScript, server-language code—such PHP or ColdFusion Markup Language (CFML)—and any other kind of code. For more information, see Chapter 20, "Coding in Dreamweaver," on page 415.

Code and Design view enables you to see both Code view and Design view for the same document in a single window.

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.

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

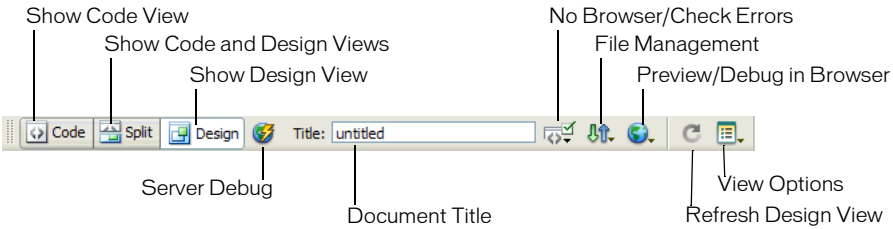
Related topics

- "Working in the Document window" on page 39

The Document toolbar

The Document toolbar contains buttons that let you toggle between different views of your document quickly: Code, Design, and a split view that shows both Code and Design view.

The toolbar also contains some common commands and options related to viewing the document and transferring it between the local and remote sites.



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.

Server Debug displays a report to help you debug the current ColdFusion page. The report includes errors, if any, in your page.

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.

No Browser/Check Errors enables you to check cross-browser compatibility.

File Management displays the File Management pop-up menu.

Preview/Debug in Browser allows you to preview or debug your document in a browser. Select a browser from the pop-up menu.

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.

View Options allows you to set options for Code view and Design view, including which view should appear above the other. Options in the menu are for the current view: Design view, Code view, or both. For information about Code view options, see "Setting viewing preferences" on page 403. For information about Design view options, see "Selecting elements in the Document window" on page 262, "Viewing and editing head content" on page 449, and "Using visual guides in Dreamweaver" on page 54.

Related topics

- "Displaying toolbars" on page 41

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.

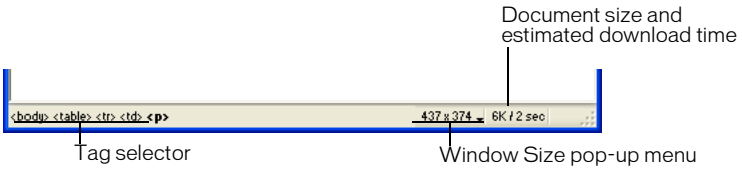
For more information about operations such as Open and Save, see Chapter 3, "Creating and Opening Documents," on page 69.

Related topics

- "Displaying toolbars" on page 41

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 select a class or ID from the context menu.

The *Window Size* pop-up menu (visible in Design view only) lets you resize the Document window to predetermined or custom dimensions. For more information, see "Resizing the Document window" on page 40.

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. For more information, see "Setting download time and size preferences" on page 272.

Related topics

- "Setting Status Bar preferences" on page 41

The Insert bar

The Insert bar contains buttons for creating and inserting objects such as tables, layers, and images. When you roll the pointer over a button, a tooltip appears with the name of the button.

The buttons are organized into several categories, which you can switch on the left side of the Insert bar. Additional categories appear when the current document contains server code, such as ASP or CFML documents. When you start Dreamweaver, the category you were last working in opens.

Note: If you prefer to view the categories as tabs across the top of the Insert bar, you can change the layout of the Insert bar (see "Setting Fonts preferences for Dreamweaver display" on page 56).

Some categories have buttons with pop-up menus. When you select an option from a pop-up menu, it becomes the default action for the button. For example, if you select Image Placeholder from the Image button's pop-up menu, the next time you click the Image button, Dreamweaver inserts an image placeholder. Anytime you select a new option from the pop-up menu, the default action for the button changes.

The Insert bar is organized in the following categories:

The Common category enables you to create and insert the most commonly used objects, such as images and tables.

The Layout category enables you to insert tables, `div` tags, layers, and frames. You can also you choose among three views of tables: Standard (default), Expanded Tables, and Layout. When Layout mode is selected, you can use the Dreamweaver layout tools: Draw Layout Cell and Draw Layout Table.

The Forms category contains buttons for creating forms and inserting form elements.

The Text category enables you to insert a variety of text- and list-formatting tags, such as `b`, `em`, `p`, `h1`, and `ul`.

The HTML category enables you to insert HTML tags for horizontal rules, head content, tables, frames, and scripts.

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

The Application category enables you to insert dynamic elements such as recordsets, repeated regions, and record insertion and update forms.

The Flash elements category enables you to insert Flash elements.

The Favorites category enables you to group and organize the Insert bar buttons you use the most in one common place.

You can modify any object in the Insert bar or create your own objects (see "Customizing Dreamweaver" on the Macromedia Support Center at www.macromedia.com/go/customizing_dreamweaver).

Related topics

- "Using the Insert bar" on page 42

Reports in Dreamweaver

You can run reports in Dreamweaver to find content, troubleshoot, or test content. You can generate the following types of reports:

Search enables you to search for tags, attributes, and specific text within tags. To search your code, see "Searching and replacing tags and attributes" on page 428.

Validation enables you to check for code or syntax errors. To validate your code, see "Validating your tags" on page 437.

Target Browser Check enables you to test the HTML in your documents to see if any tags or attributes are unsupported by your target browsers. To run a target browser check, see "Checking for browser compatibility" on page 434.

Link Checker enables you to find and fix broken, external, and orphaned links. To run the link checker, see "Checking for broken, external, and orphaned links" on page 336.

Site Reports enable you to improve workflow and test HTML attributes in your site. Workflow reports include checked out by, recently modified, and design notes; HTML reports include combinable nested font tags, accessibility, missing Alt text, redundant nested tags, removable empty tags, and untitled documents. To run site reports, see "Testing your site" on page 111.

FTP Log enables you view all FTP file transfer activity. To view the FTP log, see "Getting files from a remote server" on page 101 or "Putting files on a remote server" on page 102.

Server Debug enables you to view information to debug a ColdFusion application. To view debugging information, see "Using the ColdFusion debugger (Windows only)" on page 439.

About customizing Dreamweaver in multiuser systems

You can customize Dreamweaver to suit your needs even in a multiuser operating system such as 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).

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.

Related topics

- "Dreamweaver customizing basics"

Working in the Document window

The Document window shows the current document. You can view a document in Design view, Code view, or Code and Design views. The Document window's status bar provides information about the current document.

Related topics

- "The Document window" on page 34
- "The status bar" on page 36

Switching between views in the Document window

You can view a document in the Document window in Code view, Design view, or Code and Design views.

To switch views in the Document window, do one of the following:

- Use the View menu:
  - Select View > Code.
  - Select View > Design.
  - Select View > Code and Design.
- Use the Document toolbar:
  - Click the Show Code View button.

  - Click the Show Code and Design Views button.

  - Click the Show Design View button.

To toggle between Code view and Design view:

- Press Control+tilde (~) (Windows) or Command+backquote (‘) (Macintosh).

  Note: If both views are showing in the Document window, this keyboard shortcut changes keyboard focus from one view to the other.

Related topics

- "The Document window" on page 34

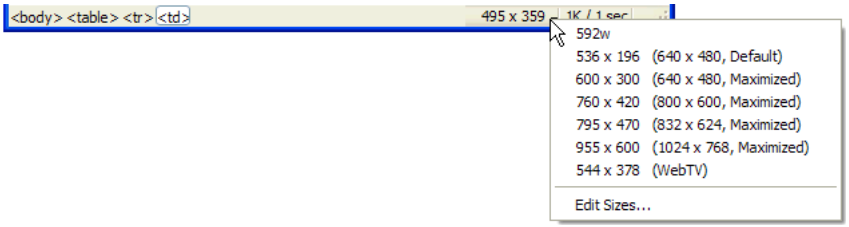
Resizing the Document window

The status bar displays the Document window's current dimensions (in pixels). 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.

To resize the Document window to a predetermined size:

- Select one of the sizes from the Window Size pop-up menu at the bottom of the Document window.

  Note: In Windows, you can maximize a Document window so that it fills the entire document area of the integrated window. You can't resize a Document window when it is maximized.



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.

Tip: For less precise resizing, use your operating system's standard methods of resizing windows, such as dragging the lower right corner of a window.

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

1. Select Edit Sizes from the Window Size pop-up menu.
2. Click any of the width or height values in the Window Sizes list, then 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. Select 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.

Setting Status Bar preferences

Set preferences for the status bar using the Preferences dialog box.

To set preferences for the status bar:

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).
The Preferences dialog box appears.
2 Select Status Bar from the Category list on the left.
3 Set preference options.
For more information, click the Help button in the dialog box.
4 Click OK.

Related topics

- "The status bar" on page 36

Using toolbars, inspectors, and context menus

Dreamweaver contains several tools that enable you to quickly make changes as you create or edit a document.

The Document and Standard toolbar are for editing and working with the current document. The Insert bar contains buttons for creating and inserting objects such as tables, layers, and images, while the Property inspector enables you to edit properties for those objects.

Note: You can also use the Tag inspector to view and edit properties (see "Changing attributes with the Tag inspector" on page 442). The Tag inspector enables you to view and edit every attribute associated with a given tag, while the Property inspector shows only the most common attributes.

Context menus give you an alternative to the Insert bar and Property inspector for creating and editing objects.

Displaying toolbars

Use the Document and Standard toolbars to perform document-related and standard editing operations. You can choose to display the toolbars as necessary. The setting for displaying or hiding toolbars is document specific; the setting applies to the current document only.

To display or hide the Document toolbar, do either of the following:

- Select Window > Toolbars > Document.
- Right-click (Windows) or Control-click (Macintosh) in the Insert bar or the Document or Standard toolbar, then select Document Toolbar.

To display or hide the Standard toolbar, do either of the following:

- Select Window > Toolbars > Standard.
- Right-click (Windows) or Control-click (Macintosh) in the Insert bar or the Document or Standard toolbar, then select Standard Toolbar.

Related topics

- "The Document toolbar" on page 35
- "The Standard toolbar" on page 36

Using the Insert bar

The Insert bar contains buttons for creating and inserting objects such as tables and images. The buttons are organized into categories. When you roll the pointer over a button, a tooltip appears with the name of the button.

Related topics

- "The Insert bar" on page 37

Displaying the Insert bar and its categories and menus

You can hide or show, and collapse or expand the Insert bar as necessary. You can also display the different categories of the Insert bar. Some categories of the Insert bar have buttons that contain menus with common commands.

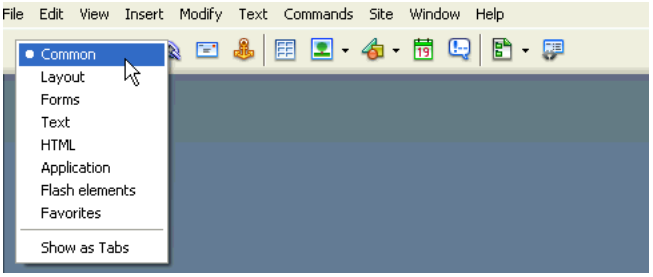
If you prefer to view the categories as tabs across the top of the Insert bar, you can change the layout of the Insert bar.

To hide or show the Insert bar, do either of the following:

- Select Window > Insert.
- Right-click (Windows) or Control-click (Macintosh) in the Insert bar or the Document or Standard toolbar, then select Insert Bar.

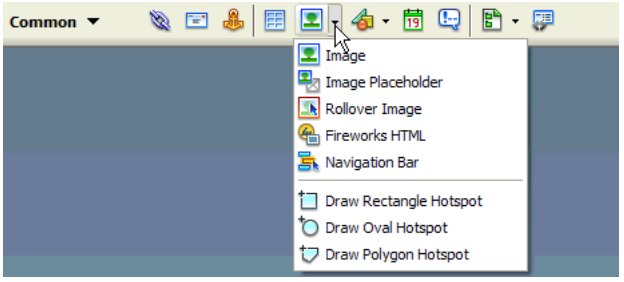
To show the buttons in a particular category:

- Click the arrow beside the category name on the left end of the Insert bar, then select another category from the pop-up menu.



To display the pop-up menu for a button:

- Click the down arrow beside the button's icon.



To show Insert bar categories as tabs:

- Click the arrow beside the category name on the left end of the Insert bar, then select Show as Tabs.

The Insert bar displays the categories as tabs across the top of the Insert bar.

Note: You might need to click the Insert bar title bar to reopen the Insert bar.

To show Insert bar categories as a menu:

- Right-click (Windows) or Control-click (Macintosh) in the Insert bar, then select Show as Menus.

The Insert bar displays the categories in a menu instead of tabs.

Related topics

- "Customizing and using the Favorites category in the Insert bar" on page 44

Using the Insert bar to insert objects

The Insert bar is a convenient method for creating and inserting objects.

To insert an object:

1. Select the appropriate category from the left side of the Insert bar.
2. Do one of the following:
   - Click an object button or drag the button's icon into the Document window.
   - Click the arrow on a button, then select an option from the menu.

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. Or, Dreamweaver may insert code into the document, or open a tag editor (see "Editing tags with Tag editors" on page 426) or panel for you to specify 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.

Note: 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, see "Showing and hiding invisible elements" on page 263.

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.

To modify preferences for the Insert bar:

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

The Preferences dialog box displays the General preferences category.

2 Deselect Show Dialog When Inserting Objects to suppress dialog boxes when you insert objects such as images, tables, scripts, and head elements or by holding down the Control key (Windows) or the Option key (Macintosh) while creating the object.

Tip: 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.

3 Click OK.

Related topics

- "Displaying the Insert bar and its categories and menus" on page 42

Customizing and using the Favorites category in the Insert bar

The Favorites category of the Insert bar enables you to group and organize commonly used Insert bar buttons. You can add, manage, and delete buttons from the Favorites category.

To add, delete, or manage items in the Favorites category:

1 Select any category in the Insert bar.

2 Right-click (Windows) or Control-click (Macintosh) in the area where the buttons appear (do not right-click in the category name), then select Customize Objects.

The Customize Favorite Objects dialog box appears.

3 Make changes as necessary.

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

4 Click OK.

The Favorites category reflects the changes you made.

Tip: If you're not in the Favorites category, select that category to see your changes.

To insert objects using buttons in the Favorites category:

- Select the Favorites category from the left side of the Insert bar, then click the button for any Favorites object you've added.

Note: The Favorites category does not contain any buttons until you customize Favorites to add objects.

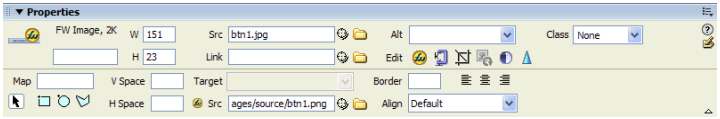
Related topics

- "Displaying the Insert bar and its categories and menus" on page 42
- "Using the Insert bar to insert objects" on page 43

Using the Property inspector

The Property inspector lets you examine and edit the most common properties for the currently selected page element, such as text or an inserted object. The contents of the Property inspector vary depending on the element selected.

Note: Use the Tag inspector to view and edit every attribute associated with a given tag properties (see "Changing attributes with the Tag inspector" on page 442).



Tip: The Property inspector is at the bottom of the workspace by default, but you can dock it at the top of the workspace if you want. Or, you can make it a floating panel in the workspace. For more information about moving the Property inspector, see "Docking and undocking panels and panel groups" on page 47.

To show or hide the Property inspector:

- Select Window > Properties.

To expand or collapse the Property inspector:

- Click the expander arrow in the lower right corner of the Property inspector.

To view properties for a page element:

- Select the page element in the Document window.

Note: You might need to expand the Property inspector to view all the properties for the selected element.

To change properties for a page element:

1. Select the page element in the Document window.
2. Change any of the properties in the Property inspector.

   Note: 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.

   Most changes you make to properties are immediately applied in the Document window.
3. If your changes are not applied immediately, do one of the following:
   - Click outside the property-editing text fields.
   - Press Enter (Windows) or Return (Macintosh).
   - Press Tab to switch to another property.

Using context menus

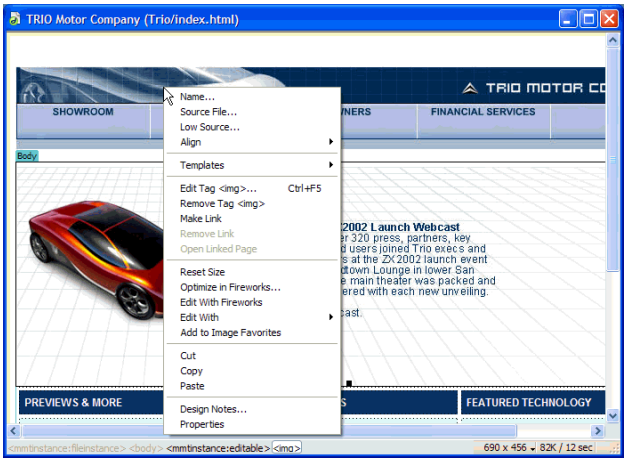
Dreamweaver makes extensive use of context menus, which provide convenient access to the most useful commands and properties related to the object or window you're working with. Context menus list only those commands that pertain to the current selection.

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 Select a command from the context menu.



Using panels and panel groups

Panels in Dreamweaver are grouped together into panel groups. The selected panel within a panel group appears as a tab. 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 only). This makes it easy to access the panels you need without cluttering your workspace.

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.

Viewing panels and panel groups

You can display or hide panel groups and panels in the workspace as necessary.

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 close a panel group so that it isn't visible on your screen:

- Select 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 or panel that isn't visible on your screen:

- Select the Window menu, then select a panel name from the 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).

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

To select a panel within an expanded panel group:

- Click the panel's name.

To see a panel group's Options menu if it isn't showing:

- Expand the panel group by clicking its name or 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.

Docking and undocking panels and panel groups

You can move panels and panel groups as necessary, and arrange them so that they are floating or docked in the workspace.

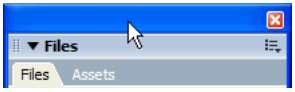
Most panels can be docked only to either the left or the right of the Document window area in the integrated workspace, while others (such as the Property inspector and the Insert bar) can be docked only to the top or bottom of the integrated window.

To 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.



To undock a panel from a panel group:

- Select Group With > New Panel Group from the Options menu in the panel group's title bar. (The Group With command's name changes depending on the name of the active panel.)

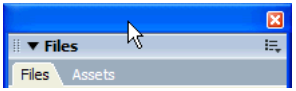
  The panel appears in a new panel group of its own.

To dock a panel in a panel group:

- Select the name of a panel group from the Group With submenu of the panel group's Options menu. (The Group With command's name changes depending on the name of the active panel.)

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.



Resizing and renaming panel groups

You can change the size and name of panel groups depending on your needs.

To change the size of the panel groups:

- For floating panels, 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.
- For docked panels, drag the splitter bar between the panels and the Document window.

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

- Select 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 rename a panel group:

1. Select Rename Panel Group from the Options menu in the panel group's title bar.
2. Enter a new name, then click OK.

Setting Panels preferences

You can set 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.

To set preferences for panels:

1. Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

   The Preferences dialog box appears.
2. Select Panels from the Category list on the left.
3. Select options.

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

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.

Related topics

- "Dreamweaver and accessibility" on page 26

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.

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.

Dreamweaver supports JAWS for Windows, from Freedom Scientific (www.freedomscientific.com), and Window Eyes screen readers, from GW Micro (www.gwmicro.com).

Related topics

- "Using the keyboard to navigate Dreamweaver" on page 50

Support for operating system accessibility features (Windows only)

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.

Related topics

- "Using screen readers with Dreamweaver" on page 49

Using the keyboard to navigate Dreamweaver

You can use the keyboard to navigate Dreamweaver panels, inspectors, dialog boxes, frames, and tables without a mouse.

Related topics

- "Using screen readers with Dreamweaver" on page 49
- "Support for operating system accessibility features (Windows only)" on page 49

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 Spacebar. Press the Spacebar 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 Spacebar 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.

Related topics

- "Navigating dialog boxes" on page 51
- "Navigating frames" on page 52
- "Navigating tables" on page 52

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 Spacebar.

Related topics

- "Navigating panels" on page 50
- "Navigating frames" on page 52
- "Navigating tables" on page 52

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.

Related topics

- "Navigating panels" on page 50
- "Navigating the Property inspector" on page 51
- "Navigating tables" on page 52

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.

Related topics

- "Navigating panels" on page 50
- "Navigating the Property inspector" on page 51
- "Navigating dialog boxes" on page 51

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 insertion point 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.

Related topics

- "Navigating panels" on page 50
- "Navigating the Property inspector" on page 51
- "Navigating dialog boxes" on page 51
- "Navigating frames" on page 52

Optimizing the workspace for accessible page design

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 box for each object, so that Dreamweaver prompts you for accessibility information when you insert objects. You can activate a dialog box for any of the objects in the Accessibility category in Preferences.

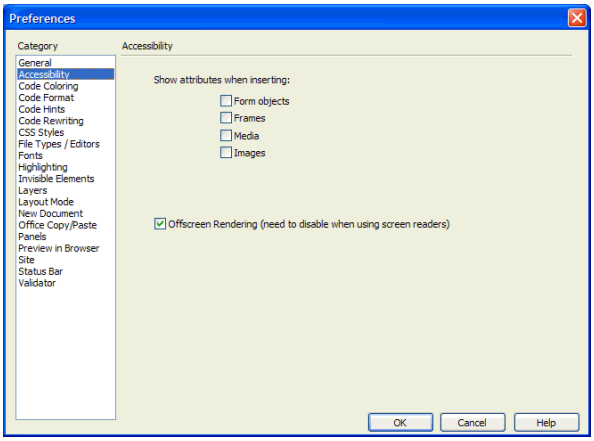
To activate the Accessibility dialog boxes:

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

The Preferences dialog box appears.

2 Select Accessibility from the Category list on the left.

The Preferences dialog box displays accessibility options.



3 Select the objects you want to activate Accessibility dialog boxes for.

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

Note: Accessibility attributes automatically appear in the Insert Table dialog box when you insert a new table.

4 Click OK.

For each object you select, an Accessibility dialog box prompts you to enter accessibility tags and attributes when you insert that object in a document.

Related topics

- “Dreamweaver and accessibility” on page 26
- “Inserting an image” on page 305
- “Validating HTML form data” on page 620
- “Creating frames and framesets” on page 204
- “Inserting and editing media objects” on page 358
- “Inserting a table and adding content” on page 165

Using visual guides in Dreamweaver

Dreamweaver provides several kinds of visual guides to help you design documents and predict (approximately) how they will appear in browsers. You can do any 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 40.)
- Use rulers to provide a visual cue for positioning and resizing layers or tables. (See "Using rulers and grids to lay out pages" on page 160.)
- 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 161.)
- Use the grid for precise positioning and resizing of layers. (See "Using rulers and grids to lay out pages" on page 160.)

  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.

Dreamweaver customizing 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 change the workspace layout, hide or display the Start page, set preferences, create your own keyboard shortcuts, and add extensions to Dreamweaver.

For information about customizing panel layout in the workspace, see "Using panels and panel groups" on page 46.

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

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

Related topics

- "About customizing Dreamweaver in multiuser systems"
- Extending Dreamweaver Help

Choosing the workspace layout (Windows only)

In Windows, you can choose between the Designer and Coder workspace layouts. The first time you start Dreamweaver, a dialog box appears that enables you to choose a workspace layout. You can switch to a different workspace at anytime.

To choose a workspace layout the first time you start Dreamweaver:

1 Select one of the following layouts:

Designer 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.

Coder 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 the Document window showing Code view by default.

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

2 Click OK.

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

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

The Preferences dialog box appears.

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, then 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.

Related topics

- "The workspace layout" on page 32

Hiding and displaying the Start page

The Dreamweaver Start page appears when you start Dreamweaver and anytime that you do not have any documents open. You can choose to hide the Start page, then later display it again. When the Start page is hidden and there are no documents open, the Document window is blank.

Note: You can choose to have Dreamweaver open the last document you were working each time you start. For more information, see "Setting General preferences for Dreamweaver" on page 56.

To hide the Start page:

- Select the Don't Show Again checkbox on the Start page.

The Start page will not appear when you start Dreamweaver, or after you open and close a document.

To display the Start page:

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

The Preferences dialog box appears displaying the General preferences category.

2 Select the Show Start Page checkbox.

The Start page appears when you start Dreamweaver, or after you open and close a document.

Setting General preferences for Dreamweaver

The General preferences control the general appearance of Dreamweaver.

To change the General preferences:

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

The Preferences dialog box appears displaying the General preferences category.

2 Set preference options.

For more information, click Help.

3 Click OK.

Setting Fonts preferences for Dreamweaver display

A document's encoding determines how the document appears in a browser. Dreamweaver font preferences 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.

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

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

The Preferences dialog box appears.

2 Select Fonts from the Category list on the left.

3 Select an encoding type (such as Western (Latin1) or Japanese) from the Font Settings list, then select fonts to use within Dreamweaver for that encoding, using the font pop-up menus below the Font Settings list.

For more information, click Help.

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

4 Click OK.

Related topics

- "Understanding document encoding" on page 256

Customizing keyboard shortcuts

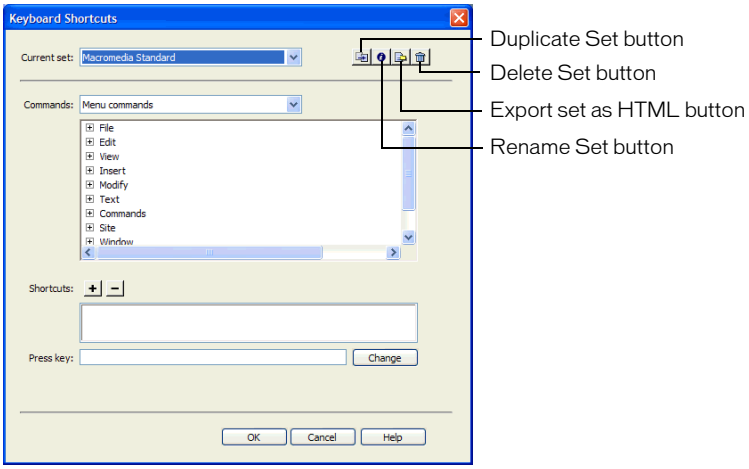
Use the Keyboard Shortcut Editor to create your own shortcut keys, including keyboard shortcuts for code snippets. You can also remove shortcuts, edit existing shortcuts, and select a predetermined set of shortcuts in the Keyboard Shortcut Editor.

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

To customize keyboard shortcuts:

1 Select Edit > Keyboard Shortcuts (Windows) or Dreamweaver > Keyboard Shortcuts (Macintosh).

The Keyboard Shortcuts dialog box appears.



2 Add, delete, or edit keyboard shortcuts.

For more information, click Help.

3 Click OK.

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 have access to 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 "About customizing Dreamweaver in multiuser systems" on page 38.

To find the latest extensions for Dreamweaver, use the Macromedia Exchange website at www.macromedia.com/go/dreamweaver_exchange/. 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. Start 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 let you 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 (.mxp) 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 select 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.

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

CHAPTER 2
Setting Up a Dreamweaver Site

A website is a set of linked documents and assets with shared attributes, such as related topics, a similar design, or a shared purpose. Macromedia Dreamweaver MX 2004 is a site creation and management tool, so you can use it to create complete websites, in addition to individual documents.

The first step in creating a website is planning. For best results, design and plan the structure of your website before you create any of the pages that the site will contain.

The next step is to set up Dreamweaver so that you can work on the basic structure of the site. If you already have a site on a web server, you can use Dreamweaver to edit that site.

Tip: As you develop site content, consider basing your pages on Dreamweaver templates. See "About Dreamweaver templates" on page 216 and "Creating a Dreamweaver template" on page 226.

This chapter includes the following sections:

- "About Dreamweaver sites" on page 59
- "Setting up a new Dreamweaver site" on page 62
- "Using the Advanced settings to set up a Dreamweaver site" on page 63
- "Editing settings for a Dreamweaver site" on page 67
- "Editing existing websites in Dreamweaver" on page 67

About Dreamweaver sites

A Dreamweaver site provides a way to organize all of the documents associated with a website. Organizing your files in a site enables you to use Dreamweaver to upload your site to the web server, automatically track and maintain your links, manage files, and share files. To take full advantage of Dreamweaver features, you should define a site.

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

The local folder is your working directory. Dreamweaver refers to this folder as your "local site." This folder can be on your local machine or it can be on a network server. It is where you store the files you are working on for a Dreamweaver site.

All you need to do to define a Dreamweaver site is set up a local folder. To transfer files to a web server or to develop web applications, you need to also add information for a remote site and testing server.

The remote folder is where you store your files, depending on your development environment, for testing, production, collaboration, and so on. Dreamweaver refers to this folder as your "remote site" in the Files panel. Typically, your remote folder is on the machine where your web server is running.

Together, 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.

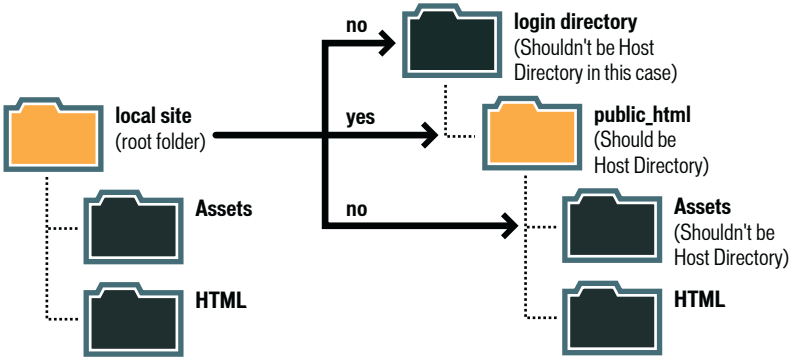
The testing server folder is the folder where Dreamweaver processes dynamic pages. For more information, see "Specifying where dynamic pages can be processed" on page 461.

Related topics

- "Setting up a new Dreamweaver site" on page 62

Understanding local and remote folder structure

When you set up access to the remote folder for your Dreamweaver site (see "Setting up a remote folder" on page 65), 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.

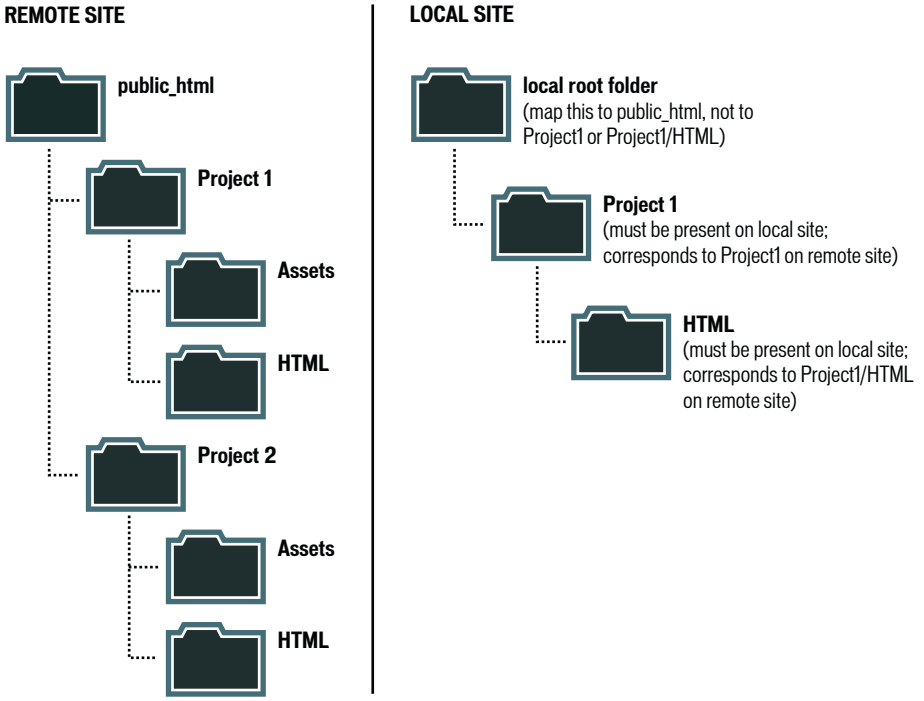


If the structure of your remote folder doesn't match the structure of your local folder, Dreamweaver uploads your files to the wrong place and the files might not be visible to site visitors. Also, your image and link paths might be broken.

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 one for you.

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.



Related topics

- "About Dreamweaver sites" on page 59
- "Setting up a remote folder" on page 65

Setting up a new Dreamweaver site

After you plan your site structure, or if you already have an existing site, you should define a site in Dreamweaver before you start developing. Setting up a Dreamweaver site is a way to organize all of the documents associated with a website. For more information, see "About Dreamweaver sites" on page 59.

Note: You can also edit files without setting up a Dreamweaver site. For more information, see "Accessing sites, a server, and local drives" on page 80.

After you set up a Dreamweaver site, it's a good idea to export the site so that you have a local backup copy. For more information, see "Importing and exporting sites" on page 96.

To set up a Dreamweaver site:

1 Select Site > Manage Sites.

The Manage Sites dialog box appears.

Note: If you don't have any Dreamweaver sites defined, then the Site Definition dialog box appears and you can skip the next step.

2 Click the New button.

The Site Definition dialog box appears.

3 Do one of the following:

- Click the Basic tab to use the Site Definition Wizard, which steps you through the setup process.

  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.

- Click the Advanced tab to use the Advanced settings, which enable you to set up local, remote, and testing folders individually, as necessary.

  Note: 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.

4 Complete the Dreamweaver site setup process:

- For the Site Definition Wizard, answer the questions on each screen, then click Next to advance through the setup process or click Back to return to a previous screen.
- For Advanced settings, complete the Local Info category, the RemoteInfo category, and the Testing Server category, as necessary (see "Using the Advanced settings to set up a Dreamweaver site" on page 63).

Tip: After you set up a Dreamweaver site, it's a good idea to export the site so that you have a local backup copy. For more information, see "Importing and exporting sites" on page 96.

If you are setting up a Dreamweaver site for a web application, see Chapter 23, "Setting Up a Web Application," on page 453.

Using the Advanced settings to set up a Dreamweaver site

You can use the Advanced settings of the Site Definition dialog box to set up a Dreamweaver site. The Advanced settings enable you to set up local, remote, and testing (for processing dynamic pages) folders individually, as necessary. This method is recommended for users who have experience with using Dreamweaver.

Note: To get started quickly, you can just start with the first step of setting up a Dreamweaver site, setting up your local folder. Then, you can add remote and testing information later. You should at least set up a local folder before you start using Dreamweaver.

If you are new to Dreamweaver, you can use the Site Definition Wizard, instead of the Advanced settings, to guide you through the setup process. (See "Setting up a new Dreamweaver site" on page 62.)

If you are setting up a Dreamweaver site for a web application, skip this section and see Chapter 23, "Setting Up a Web Application," on page 453.

Tip: After you set up a Dreamweaver site, it's a good idea to export the site so that you have a local backup copy. For more information, see "Importing and exporting sites" on page 96.

Related topics

- "About Dreamweaver sites" on page 59
- "Specifying where dynamic pages can be processed" on page 461

Setting up a local folder

The local folder is your working directory for your Dreamweaver site. This folder can be on your local machine or it can be on a network server.

To set up a local folder:

1 Select Site > Manage Sites.

The Manage Sites dialog box appears.

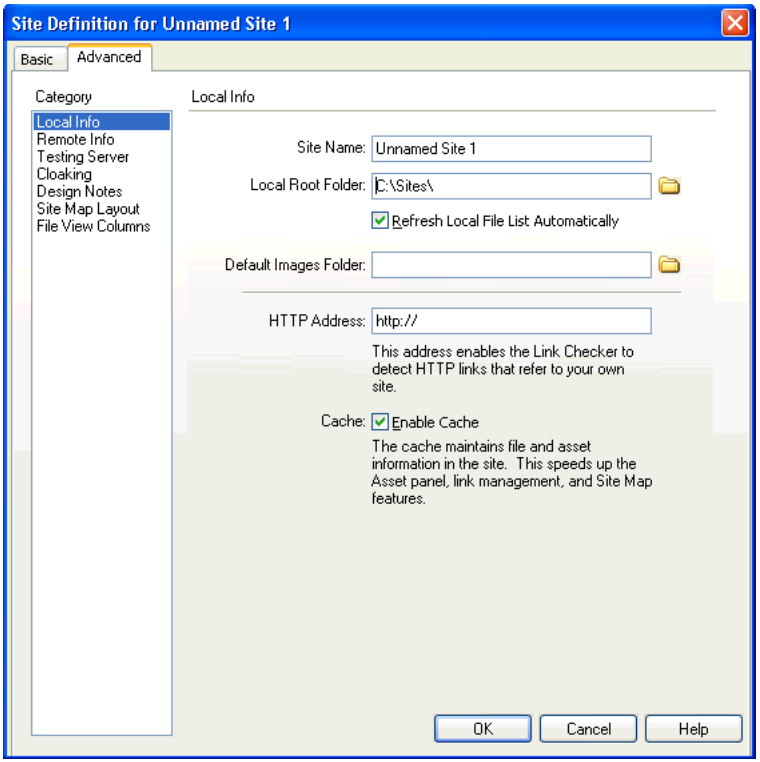
Note: If you don't have any Dreamweaver sites defined, then the Site Definition dialog box appears and you can skip the next step.

2 Click New.

The Site Definition dialog box appears.

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

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



4 Enter the Local Info options.

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

5 (Optional) If you are ready to set up your remote server now, skip the remaining step; select the Remote Info category on the left, then complete the dialog box.

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

6 Click OK.

Dreamweaver creates the initial site cache, and the new Dreamweaver site appears in the Files panel.

After you set up a local folder, you might also add remote and testing folders (see "Setting up a remote folder" on page 65 and "Specifying where dynamic pages can be processed" on page 461).

Related topics

- "About Dreamweaver sites" on page 59
- "Managing Your Files" on page 77

Setting up a remote folder

After you set up a local folder for a Dreamweaver site (see "Setting up a local folder" on page 63), you can set up a remote folder. Depending on your development environment, the remote folder is where you store files for testing, collaboration, production, deployment, or so on.

Note: You don't need to specify a remote folder if the folder you specified as your local folder 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. This section describes how to set up a remote folder and connect to it.

To set up a remote folder:

1 Select Site > Manage Sites.

The Manage 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 63).

3 Click Edit.

The Site Definition dialog box appears.

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

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

6 Select an Access option.

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

7 Click OK.

Dreamweaver creates a connection to the remote folder. If you are developing a dynamic site, you can add a Testing Server folder for processing dynamic pages (see "Specifying where dynamic pages can be processed" on page 461).

To connect to a remote folder with FTP access:

- In the Files panel, click the Connects to Remote Host button in the toolbar.

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

  Note: If you're using network access for your remote folder, you do not need to connect to the remote folder; you are always connected. Click the Refresh button to see your remote files.

To disconnect from a remote folder:

- In the Files panel, click the Disconnect button in the toolbar.

Related topics

- "About Dreamweaver sites" on page 59
- "Understanding local and remote folder structure" on page 60
- "Troubleshooting the remote folder setup" on page 66

Troubleshooting the 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 (see "Setting up a remote folder" on page 65), 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 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; however, 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. For more detailed information, look at the FTP log.

  Note: In general, when you encounter a problem with an FTP transfer, examine the FTP log by selecting Window › Results (Windows) or Site › FTP Log (Macintosh), then clicking the FTP Log tag.

Editing settings for a Dreamweaver site

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

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

- Select Site > Manage Sites, select a site in the Manage Sites dialog box, then click Edit.
- In the Files panel, select Manage Sites from the pop-up menu where the current site, server, or drive appears; select a site in the Manage Sites dialog box, then click Edit.

Related topics

- "Setting up a new Dreamweaver site" on page 62

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 edit 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.

Note: This section sets up a Dreamweaver site to edit an existing local website. You can also edit the existing site without creating a Dreamweaver site. For more information, see "Accessing sites, a server, and local drives" on page 80.

To edit an existing local website:

1 Select Site > Manage Sites.

The Manage Sites dialog box appears.

2 Click New.

The Site Definition dialog box appears.

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

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

4 Complete the dialog box.

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

5 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.

Note: This section sets up a Dreamweaver site to edit an existing remote website. You can also edit the existing site without creating a Dreamweaver site. For more information, see "Accessing sites, a server, and local drives" on page 80.

To edit an existing remote site:

1 Create a local folder to contain the existing site, then set it up as the local folder for the site (see "Setting up a local folder" on page 63).

Note: You must locally duplicate the entire structure of the relevant branch of the existing remote site. For more information, see "Understanding local and remote folder structure" on page 60.

2 Set up a remote folder (see "Setting up a remote folder" on page 65), using information about your existing site.

Make sure to choose the correct root folder for the remote site.

3 In the Files panel (Window > Files), click the Connects to Remote Host button (for FTP access) or the Refresh button (for network access) in the toolbar to view 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 in the Files panel, then click Get in the toolbar 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 Files panel, then click Get in the toolbar 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 Use Dreamweaver to work on your site.

Related topics

- "Editing an existing local website in Dreamweaver" on page 67

CHAPTER 3
Creating and Opening Documents

Macromedia Dreamweaver MX 2004 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 a variety of text-based documents, including CFML, ASP, JavaScript, and CSS. Dreamweaver also supports source code files, such as Visual Basic, .NET, C#, and Java.

Dreamweaver provides several options for creating a new document. You can create any of the following:

- A new blank document or template
- A document based on one of the predesigned page layouts included with Dreamweaver
- A document based on one of your existing templates

Other document options are also available. For example, 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 contains the following sections:

- "Creating new documents" on page 70
- "Saving a new document" on page 72
- "Setting a default new document type" on page 72
- "Opening existing documents" on page 73
- "Importing a Microsoft Word HTML file" on page 74

Creating new documents

Dreamweaver provides you with several choices for selecting a new document to work in. You can create a new document in the following ways:

- Start with a blank document (see "Creating a new blank document" on page 70).
- Create a blank document or template based on a Dreamweaver design file (see "Creating a document based on a Dreamweaver design file" on page 70).
- Use a template that defines how a document looks, and sets which parts of a document can be edited (see "Creating a document based on an existing template" on page 71).

Tip: 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 a default new document type" on page 72.

Creating a new blank document

You can select the type of blank document you want to create.

To create a new blank document:

1 Select File > New.

The New Document dialog box appears. The General tab is already selected.

2 From the Category list, select Basic Page, Dynamic Page, Template Page, Other, or Framesets; then, from the list on the right, select the type 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.

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

3 Click the Create button.

The new document opens in the Document window.

4 Save the document (see "Saving a new document" on page 72).

Related topics

- "Creating a document based on an existing template" on page 71

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. When you create a document based on a design file, Dreamweaver creates a copy of the file.

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 209.

To create a new document from a Dreamweaver design file:

1 Select File > New.

The New Document dialog box appears. The General tab is already selected.

2 In the Category list, select CSS Style Sheets, Table Based Layouts, Page Designs (CSS), Page Designs, or Page Designs (Accessible); then select a design file from the list on the right.

You can preview a design file and read a brief description of a document's design elements.

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

3 Click the Create button.

The new document opens in the Document window. If you selected a CSS style sheet, the CSS document appears in the Document window and the CSS Style Sheet opens in Code view.

4 Save the document (see "Saving a new document" on page 72).

If the file contains links to asset files, the Copy Dependent Files dialog box appears for you to save a copy of the dependent files.

5 If the Copy Dependent Files dialog box appears, set options, then click Copy to copy the assets to the selected folder.

You can choose your own location for the dependent files or use the default folder location Dreamweaver generates (based on the design file's source name).

Related topics

- "Creating a new blank document" on page 70

Creating a document based on an existing template

You can select, preview, and create a new document from an existing template. You can use the New Document dialog box to select a template from any of your Dreamweaver-defined sites or you can use the Assets panel to create a new document from an existing template.

For information about creating templates, see "Creating a Dreamweaver template" on page 226.

Tip: If you don't have any templates in your site, you can save a document in one of the design file categories of the New Document dialog box as a template, then create pages based on that template. For information about saving a design file as a template, see "Creating a document based on a Dreamweaver design file" on page 70.

To create a new document based on a template:

1 Select File > New.

The New Document dialog box opens.

2 Click the Templates tab.

3 In the Templates For list, select the Dreamweaver site that contains the template you want use, then select a template from the list on the right.

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

4 Click Create.

The new document opens in the Document window.

5 Save the document (see "Saving a new document" on page 72).

To create a new document from a template in the Assets panel:

1 Open the Assets panel (Window > Assets), if it is not already open.

2 In the Assets panel, click the Templates icon on the left to view the list of templates in your current site.

Tip: If you just created the template you want to apply, you might need to click the Refresh button to see it.

3 Right-click (Windows) or Control-click (Macintosh) the template you want to apply, then select New From Template.

The document opens in the Document window.

Related topics

- "Creating a new blank document" on page 70
- "Creating a document based on a Dreamweaver design file" on page 70

Saving a new document

When you create a new document, you need to save it.

To save a new document:

1 Select File > Save.

2 In the dialog box that appears, navigate to the folder where you want to save the file.

Tip: It's a good idea to save your file in a Dreamweaver site. For more information, see "Setting up a new Dreamweaver site" on page 62.

3 In the File Name text box, type a name for the file.

Avoid using spaces and special characters in file and folder names and do not begin a filename with a numeral. 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.

4 Click Save.

Setting a default new document type

You can define which document type Dreamweaver uses as a default document for a site.

For example, if most pages in your site will be a specific file type (such as Cold Fusion, HTML, or ASP documents), you can set document preferences that automatically create new documents of the specified file type.

To set a default new document type and preferences:

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

The Preferences dialog box appears.

Tip: You can also click the Preferences button in the New Document dialog box to set new document preferences when you create a new document (see "Creating a new blank document" on page 70).

2 Click New Document from the category list on the left.

3 Set or change preferences as necessary.

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

4 Click OK.

Dreamweaver saves your preferences.

Opening existing documents

In Dreamweaver, you can open an existing web page or text-based document, even if it wasn't created in Dreamweaver. You can open the document and use Dreamweaver to edit it in either Design view or Code view.

If the document you open is a Microsoft Word file that was saved as HTML, you should use the Cleanup Word HTML command to remove the extraneous markup tags Word inserts into HTML files.

Tip: Instead of using the procedure described in this section to open a Microsoft Word files which was saved as HTML, you may want to import the document into Dreamweaver (see "Importing a Microsoft Word HTML file" on page 74) so that Dreamweaver automatically prompts you to clean up the file.

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.

To open an existing file:

1 Select File > Open.

The Open dialog box appears.

Tip: You can also use the Files panel to open files. For more information, see "Working with files in the Files panel" on page 87.

2 Navigate to and select the file you want to open.

Note: If you haven't already done so, it's a good idea to organize files you plan to open and edit in a Dreamweaver site, instead of opening them from another location. For information about setting up a Dreamweaver site, see "Setting up a new Dreamweaver site" on page 62.

3 Click Open.

The document opens in the Document window.

JavaScript, text, and CSS Style Sheets open in Code view by default. You can update the document while working in Dreamweaver, then save the changes in the file. For information about setting an external text editor for viewing these types of files, see "Starting an external editor for media files" on page 359.

4 If you opened a Microsoft Word document, select Commands > Clean Up Word HTML, deselect options in the dialog box if you want, and then click OK.

There might be a slight delay while Dreamweaver attempts to determine which version of Word was used to save the file. If Dreamweaver is unable to determine this, select the correct version using the pop-up menu.

For more information about options in the Clean Up Word HTML dialog box, click the Help button.

Dreamweaver applies the cleanup settings to the new document and a log of the changes appears (unless you deselected that option in the dialog box).

Importing a Microsoft Word HTML file

In Dreamweaver, you can import documents saved by Microsoft Word as HTML files, then use the Clean Up Word HTML command to remove the extraneous HTML code generated by Word. The Clean Up Word HTML command is available for documents saved as HTML files by Word 97 or later.

The code that Dreamweaver removes is primarily used by Word to format and display documents in Word and is not needed to display the HTML file. Retain a copy of your original Word (.doc) file as a backup, because you may not be able to reopen the HTML document in Word once you've applied the Clean Up Word HTML feature.

To clean up HTML or XHTML that was not generated by Microsoft Word, see "Cleaning up your code" on page 433.

To import and clean up a Microsoft Word HTML file:

1. If you haven't already, in Microsoft Word, save your document as an HTML file.
2. In Dreamweaver, select File > Import > Word Document.

   The Open dialog box opens.
3. Navigate to and select the Word HTML file you want to open.

   The document opens and the Clean Up Word HTML dialog box appears.

   There may be a slight delay while Dreamweaver attempts to determine which version of Word was used to save the file. If Dreamweaver is unable to determine this, select the correct version using the pop-up menu.
4. Deselect options in the dialog box if you want.

   For more information about options in the dialog box, click the Help button.
5. Click OK.

   Dreamweaver applies the clean up settings to the new document and a log of the changes appears (unless you deselected that option in the dialog box).

Related topics

- "Copying and pasting text from MS Office documents" on page 281

PART II
Working with Dreamweaver Sites

Learn how to manage Macromedia Dreamweaver MX 2004 sites and site-specific assets.

This part contains the following chapters:

Chapter 4, “Managing Your Files,” on page 77

Chapter 5, “Managing Site Assets and Libraries,” on page 115

Chapter 6, “Managing Contribute Sites with Dreamweaver,” on page 133

CHAPTER 4
Managing Your Files

Macromedia Dreamweaver MX 2004 helps you organize and manage your files.

Dreamweaver includes a number of features for managing files 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.

This chapter contains the following sections:

- "About site management" on page 78
- "Accessing sites, a server, and local drives" on page 80
- "Viewing files and folders" on page 83
- "Managing files and folders in the Files panel" on page 86
- "Working with a visual map of your site" on page 89
- "Importing and exporting sites" on page 96
- "Removing a Dreamweaver site from your list of sites" on page 97
- "Checking in and checking out files" on page 97
- "Getting and putting files to and from your server" on page 101
- "Synchronizing the files on your local and remote sites" on page 103
- "Cloaking folders and files in your site" on page 104
- "Storing file information in Design Notes" on page 108
- "Testing your site" on page 111

About site management

Dreamweaver includes a number of features for managing a site and transferring files to and from a remote server. Dreamweaver also contains features to make collaborative work on a website easier, such as Check In/Check Out and Design Notes.

The Check In/Check Out system

If you're working in a collaborative environment, you can check files in and out from local and remote servers.

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. For more information about using Get and Put, see "Getting and putting files to and from your server" on page 101.

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 Files 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 for 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 Files 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.

Related topics

- "Setting up the Check In/Check Out system" on page 98
- "Checking files into and out of a remote folder" on page 99

Site cloaking

Site cloaking enables you to exclude folders and file types 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:

- Performing Put, Get, Check In, and Check Out operations
- Generating reports
- Finding newer local and newer remote files
- Performing sitewide operations, such as checking and changing links
- Synchronizing

- Working with Asset panel contents
- Updating templates and libraries

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 become out of sync with their instances.

Related topics

- "Cloaking folders and files in your site" on page 104

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 see which files have Design Notes attached in the expanded Files panel: a Design Notes icon appears in the Notes column.

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.

You can also use Design Notes to keep track of sensitive information that you can't put inside a document for security reasons, such as notes on how a particular price or configuration was chosen, or what marketing factors influenced a design decision.

If you open a file in Macromedia Fireworks or 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 the following line:

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

A similar Flash Design Note might contain the following 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 "Starting an external editor for media files" on page 359), Fireworks opens the original file for editing.

Related topics

- "Enabling and disabling Design Notes for a site" on page 108
- "Associating Design Notes with files" on page 109

Accessing sites, a server, and local drives

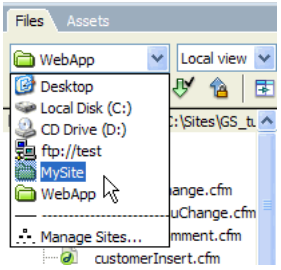
You can access, modify, and save files and folders in your Dreamweaver sites, as well as files and folders that are not part of a Dreamweaver site. In addition to Dreamweaver sites, you can access a server, a local drive, or your desktop.

Before you can access a remote server, you must set up Dreamweaver to work with that server (see "Setting up Dreamweaver to work without defining a site" on page 82).

Note: The best way to manage your files is to create a Dreamweaver site (see "Setting up a new Dreamweaver site" on page 62).

To open an existing Dreamweaver site:

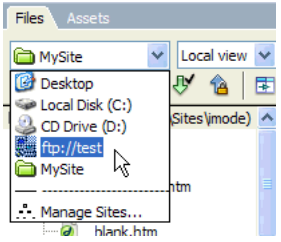
- In the Files panel (Window > Files), select a site from the pop-up menu (where the current site, server, or drive appears).



For information about changing your site view (for example, to view the remote folder), see "Viewing files and folders" on page 83. For information about opening files in your site, see "Working with files in the Files panel" on page 87.

To open a folder on a remote FTP or RDS server:

1 In the Files panel (Window > Files), select a server name from the pop-up menu (where the current site, server, or drive appears).



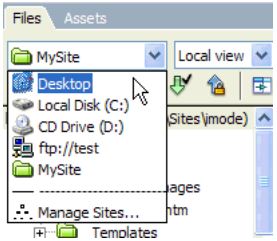
Note: Server names appear for servers you've configured Dreamweaver to work with. If you have not configured a server yet, see "Setting up Dreamweaver to work without defining a site" on page 82.

2 Navigate to and edit files as you normally do.

For more information, see "Working with files in the Files panel" on page 87.

To access a local drive or your desktop:

1 In the Files panel (Window > Files), select Desktop, Local Disk, or CD Drive from the pop-up menu (where the current site, server, or drive appears).



2 Navigate to a file, then do any of the following:

- Open files in Dreamweaver or another application
- Rename files
- Copy files
- Delete files
- Drag files

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

  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.

Related topics

- "Managing files and folders in the Files panel" on page 86

Setting up Dreamweaver to work without defining a site

Dreamweaver enables you to connect to an FTP or RDS server to work on your documents without creating a Dreamweaver site.

Note: If you work on files without creating a Dreamweaver site, you will not be able to perform sitewide operations, such as link checking. To set up a Dreamweaver site, see "Setting up a new Dreamweaver site" on page 62.

To set up Dreamweaver to work with a server from the Document window:

1 Select Site > Manage Sites.

The Manage Sites dialog box appears.

2 Click New, then select FTP & RDS Servers.

The Configure Server dialog box appears.

3 Complete the dialog box.

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

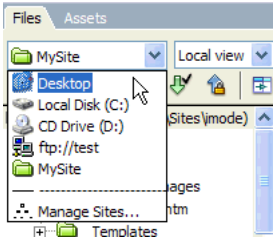
Note: You only need to complete this dialog box once for each server you want to connect to.

4 Click OK.

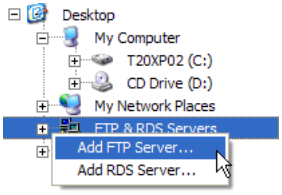
The Files panel displays the contents of the remote server folder you connected to, and the server name appears in the pop-up menu at the top of the panel.

To set up Dreamweaver to work with a server using the Files panel:

1 In the Files panel (Window > Files), select Desktop from the pop-up menu (where the current site, server, or drive appears).



2 Right-click (Windows) or Control-click (Macintosh) the FTP and RDS Servers node in the Files panel, then select Add FTP Server or Add RDS Server.



The Configure Server dialog box appears.

3 Complete the dialog box.

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

Note: You only need to complete this dialog box once for each server you want to connect to.

4 Click OK.

The Files panel displays the contents of the remote server folder you connected to, and the server name appears in the pop-up menu at the top of the panel.

Note: For large folders, it might take a few moments for the Files panel to display the folder.

Related topics

- "Accessing sites, a server, and local drives" on page 80

Viewing files and folders

You can view files and folders in the Files panel, whether they are associated with a Dreamweaver site or not. When you view sites, files, or folders in the Files panel, you can change the size of the viewing area, and, for Dreamweaver sites, you can expand or collapse the Files panel.

For Dreamweaver sites, you can also customize the Files panel by changing the view—either your local or remote site—that appears by default in the collapsed panel. Or, you can switch the content views in the expanded Files panel using the Always Show option (see "Setting preferences for the Files panel" on page 86).

To open or close the Files panel:

- Select Window > Files.

To change the size of the view area in the expanded Files panel:

- In the Files panel (Window > Files), with the panel expanded, 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 Files panel to scroll through the views' contents.
  - In the site map, drag the arrow above a file to change the space between files.

To expand or collapse the Files panel (Dreamweaver sites only):

- In the Files panel (Window > Files), click the Expand/Collapse button in the toolbar.

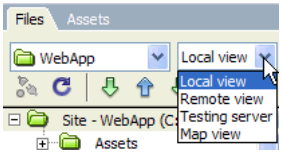
  Note: If you click the Expand/Collapse button to expand the panel while it is docked, the panel maximizes so that you cannot work in the Document window. To return to the Document window, click the Expand/Collapse button again to collapse the panel. If you click the Expand/Collapse button to expand the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

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

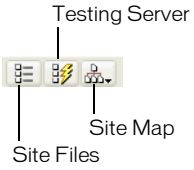
To change the site view in Files panel, do one of the following (Dreamweaver sites only):

- In the *collapsed* Files panel (Window > Files), select Local View, Remote View, Testing Server, or Map View from the Site View pop-up menu.

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



- In the *expanded* Files panel (Window > Files), click the Site Files button (for the remote site), Testing Server button, or Site Map button in the toolbar.



  If you click 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 "Viewing a site map" on page 90.

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 65 or "Specifying where dynamic pages can be processed" on page 461). Before you can view a site map, you must set up a home page (see "Viewing a site map" on page 90).

Related topics

- "Accessing sites, a server, and local drives" on page 80
- "Managing files and folders in the Files panel" on page 86

Customizing the file and folder details displayed in the expanded Files panel

When you view a Dreamweaver site in the expanded Files panel (see "Viewing files and folders" on page 83), Dreamweaver displays information about the files and folders in columns. For example, you can see the file type or the date a file was modified.

You can customize the columns by doing any of the following (some operations are only available for columns you add, not default columns):

- Reorder or realign columns
- Add new columns (for a maximum of 10 columns)
- Hide columns (except the filename column)
- Designate columns to be shared with all users connected to a site
- Delete columns (custom columns only)
- Rename columns (custom columns only)
- Associate with a Design Note (custom columns only)

To sort by any detail column in the Files panel:

- Click the heading for the column you want to sort.

  Tip: Click the heading again to reverse the order (ascending or descending) by which Dreamweaver sorts the column.

To add, delete, or change detail columns:

1 Select Site > Manage Sites.

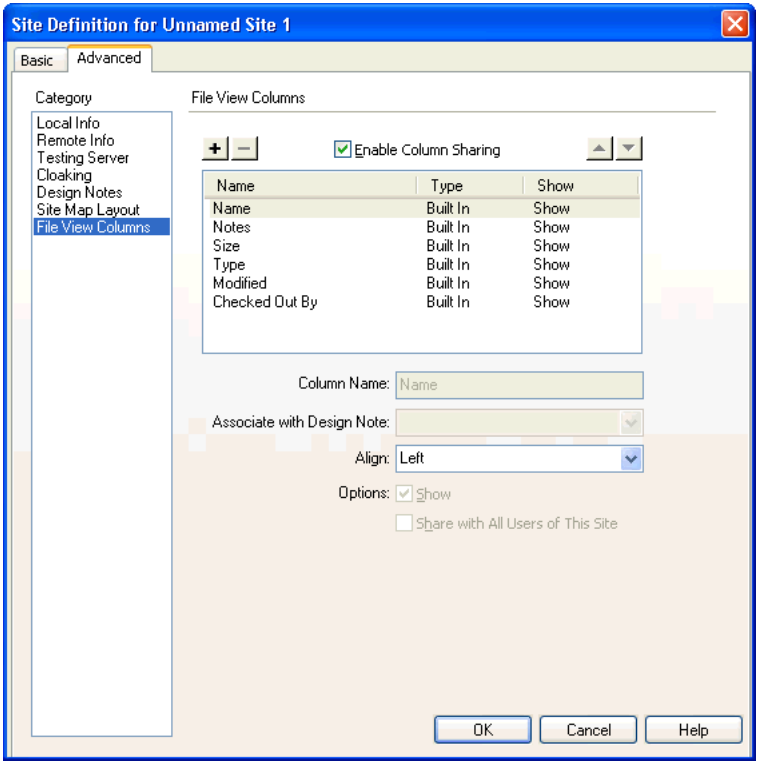
The Manage Sites dialog box appears.

2 Select a site, then click Edit.

The Site Definition dialog box appears.

3 Select File View Columns from the category list on the left.

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



4 Complete the dialog box.

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

5 Click OK.

Managing files and folders in the Files panel

You can organize and manage your site files and folders whether they are part of a Dreamweaver site, on a server you've connected to, or on your local drive or desktop.

Note: In previous versions of Dreamweaver, the Files panel was called the Site panel.

Related topics

- "Accessing sites, a server, and local drives" on page 80
- "Viewing files and folders" on page 83

Setting preferences for the Files panel

You select preferences to control file-transfer features in the Files panel.

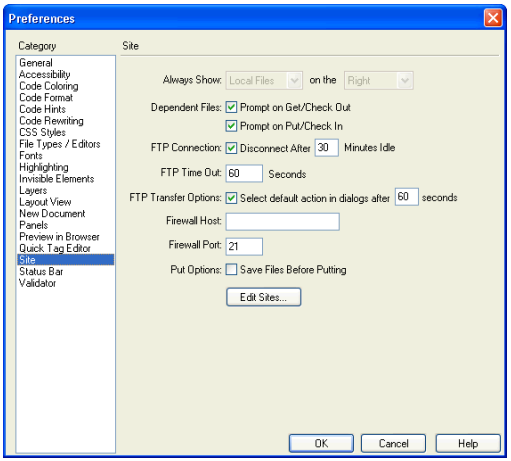
To edit Files panel preferences:

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

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.

Tip: 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 the Macintosh, FTPExtensionMapMac.txt). For more information see, Extending Dreamweaver Help.

Working with files in the Files panel

You can open or rename files; add, move, or delete files; or refresh the Files panel after you make changes.

For Dreamweaver sites, you can also 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 103.

To open a file:

1. In the Files panel (Window > Files), select a site, server, or drive from the pop-up menu (where the current site, server, or drive appears).
2. Navigate to the file you want to open.
3. Do one of the following:
   - Double-click the file's icon.
   - Right-click (Windows) or Control-click (Macintosh) the file's icon, then select Open.

Dreamweaver opens the file in the Document window.

To create a new file or folder:

1. In the Files panel (Window > Files), select a file or folder.

   Dreamweaver will create the new file or folder inside the currently selected folder, or in the same folder as the currently selected file.
2. Right-click (Windows) or Control-click (Macintosh), then select New File or New Folder.
3. Enter a name for the new file or folder.
4. Press Enter (Windows) or Return (Macintosh).

To delete a file or folder:

1. In the Files panel (Window > Files), select the file or folder you want to delete.
2. Right-click (Windows) or Control-click (Macintosh), then select Delete.

To rename a file or folder:

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

To move a file or folder:

1. In the Files panel (Window > Files), select the file or folder you want to move.
2. Do one of the following:
   - Copy the file or folder, then paste it in a new location.
   - Drag the file or folder to a new location.
3. Refresh the Files panel to see the file or folder in its new location.

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

- Right-click (Windows) or Control-click (Macintosh) any file or folder, then select Refresh.
- (Dreamweaver sites only) Click the Refresh button in the Files panel toolbar (this option refreshes both panes).

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

Related topics

- "Viewing files and folders" on page 83
- "Accessing sites, a server, and local drives" on page 80
- "Setting preferences for the Files panel" on page 86

Finding files in your Dreamweaver site

Dreamweaver makes it easy to find selected, open, checked out, or recently modified files in your site. You can also find files that are newer in your local or remote site.

To find an open file in your site:

1 Open the file in the Document window.

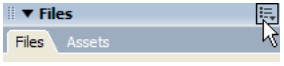
2 Select Site > Locate in Site.

Dreamweaver selects the file in the Files panel.

Note: If the open file in the Document window is not part of the current site in the Files panel, Dreamweaver attempts to determine which of your Dreamweaver sites the file belongs to; if the current file belongs to only one local site, Dreamweaver opens that site in the Files panel, then highlights the file.

To locate and select checked out files in a Dreamweaver site:

- In the *collapsed* Files panel (Window > Files), click the Options menu in the upper right corner of the Files panel, then select Edit > Select Checked Out Files.



Dreamweaver selects the files in the Files panel.

To find a selected file in your local or remote site:

1 Select the file in the Local or Remote view of the Files panel (Window > Files).

2 Right-click (Windows) or Control-click (Macintosh), then select Locate in Local Site or Locate in Local Site (depending on where you selected the file).

Dreamweaver selects the file in the Files panel.

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

- In the *collapsed* Files panel (Window > Files), click the Options menu in the upper right corner of the Files panel, then select Edit > Select Newer Local.

Dreamweaver selects the files in the Files panel.

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

- In the *collapsed* Files panel (Window > Files), click the Options menu in the upper right corner of the Files panel, then select Edit > Select Newer Remote.

Dreamweaver selects the files in the Files panel.

To find recently modified files in your site:

1. In the *collapsed* Files panel (Window > Files), click the Options menu in the upper right corner of the Files panel, then select Edit > Select Recently Modified.

   The Select Recently Modified dialog box appears.
2. Complete the dialog box.

   For more information, click the Help button in the dialog box.
3. Click OK to save your settings.

Dreamweaver highlights the files that were modified within the selected time frame in the Files panel.

Related topics

- "Accessing sites, a server, and local drives" on page 80
- "Viewing files and folders" on page 83
- "Working with files in the Files panel" on page 87

Working with a visual map of your site

You can view a local folder for a Dreamweaver site as a visual map of linked icons, called a *site map*. Use the site map to add new files to a Dreamweaver site, or to add, modify, or remove links.

The site map shows the site structure two levels deep, starting from the home page. It shows pages as icons and displays links in the order in which they are encountered in the source code.

The site map is ideal for laying out a site structure. You can quickly set up the entire structure of the site, 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, then use the Manage Sites command to define the site as a local site (see "Setting up a local folder" on page 63).

Related topics

- "Linking documents using the site map" on page 322

Viewing a 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 a site:

1 In the Files panel (Window > Files), select a site from the pop-up menu where the current site, server, or drive appears.

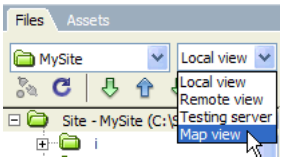
2 Right-click (Windows) or Control-click (Macintosh) the file you want to be the home page, then select Set as Home Page.

Note: You can also set the home page in the Site Map Layout options in the Site Definition dialog box (see "Modifying the site map layout" on page 91).

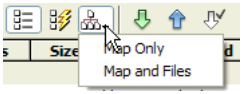
To view a site map:

1 To display the site map, in the Files panel (Window > Files), do one of the following:

- In the *collapsed* Files panel, select Map View from the Site View pop-up.



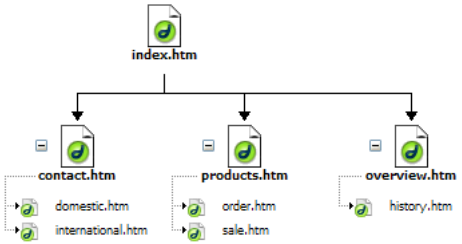
- In the *expanded* Files panel, click the Site Map button in the toolbar, then select Map Only or Map and Files.



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, Dreamweaver prompts you to select a home page.

The Files panel displays a site map showing the site structure two levels deep.



Note: By default, the site map does not display hidden files and dependent files. For more information, see "Showing and hiding site map files" on page 94.

2 Click the Plus (+) and Minus (–) next to a filename to show or hide pages linked below the second level.

3 Notice the colors in the site map:

- 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).

Related topics

- "Working with pages in the site map" on page 92
- "Showing and hiding site map files" on page 94
- "Viewing the site from a branch" on page 95

Modifying the site map layout

Use the Site Map Layout options to customize the appearance of your site map. You can change 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:

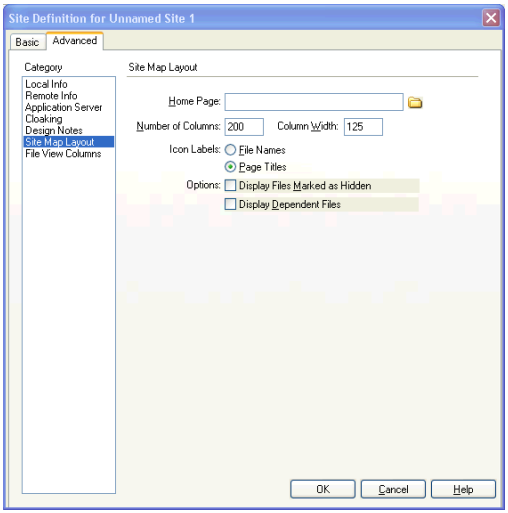
- Select Site > Manage Sites.
- In the Files panel, select Manage Sites from the pop-up menu where the current site, server, or drive appears.

The Manage Sites dialog box appears.

2 Click Edit.

The Site Definition dialog box appears.

3 Select Site Map Layout from the Category list on the left.
The Site Definition dialog box displays Site Map Layout options.



4 Make changes as necessary.
For more information, click the Help button in the dialog box.
5 Click OK to close the Site Definition dialog box.
6 Click Done to close the Manage Sites dialog box, if it is open.

Related topics

- "Viewing a site map" on page 90
- "Showing and hiding site map files" on page 94
- "Saving the site map" on page 96

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.
- Right-click (Windows) or Control-click (Macintosh) the file, then select Open.

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

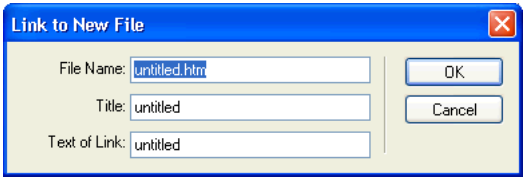
- Drag the file from the Windows Explorer or the Macintosh Finder to a file in the site map. The page is added to the site, and a link is created between it and the file you dragged it to.
- Right-click (Windows) or Control-click (Macintosh) the file in the site map, select Link to Existing File, then browse to the file.

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

1 Select a file in the site map.

2 Right-click (Windows) or Control-click (Macintosh), then select Link to New File.

The Link to New File dialog box appears.



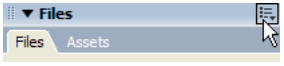
3 Enter a name, title, and text for the link.

4 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 94).

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

1 Make sure page titles are showing by clicking the Options menu in the upper right corner of the *collapsed* Files panel, then selecting File > Rename View > Show Page Titles.



2 Select a file in the site map, then do one of the following:

- Click the title. When the title becomes editable, enter a new title.
- Click the Options menu in the upper right corner, then select File > Rename.

3 Press Enter (Windows) or Return (Macintosh) after you type the new name.

Note: When you work in the Files 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 Files panel, select a file; right-click (Windows) or Control-click (Macintosh), then select Set as Home Page.
- In the Files panel, select Manage Sites from the pop-up menu where the current site, server, or drive appears, then click Edit. Select Site Map Layout in the Site Definition dialog box category list. Browse to a new home page, then click OK.

To update the site map display after making changes:

1 Click anywhere in the site map to deselect any files.

2 Click the Refresh button in the Files panel toolbar.

Related topics

- "Viewing a site map" on page 90
- "Modifying the site map layout" on page 91
- "Saving the site map" on page 96

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 Right-click (Windows) or Control-click (Macintosh), then select Show/Hide Link.

To unmark files marked as hidden in the site map:

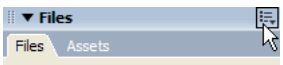
1 In the site map, select one or more files.

2 Do one of the following:

- Right-click (Windows) or Control-click (Macintosh) in the site map, then select Show/Hide Link.
- In the *collapsed* Files panel (Window > Files), click the Options menu in the upper right corner, then select View > Show Files Marked as Hidden.
- In the *collapsed* Files panel (Window > Files), click the Options menu in the upper right corner, then select View > Show/Hide Link.

To show or hide files marked as hidden in the site map:

1 In the *collapsed* Files panel (Window > Files), click the Options menu in the upper right corner.



2 Do one of the following:

- Select View > Show Files Marked as Hidden.
- Select View > Layout to open the Site Definition dialog box, then select the Display Files Marked as Hidden option.

To show dependent files in the site map:

1 In the *collapsed* Files panel (Window > Files), click the Options menu in the upper right corner.

2 Do one of the following:

- Select View > Show Dependent Files.
- Select View > Layout to open the Site Definition dialog box, then select the Display Dependent Files option.

Related topics

- "Viewing a site map" on page 90
- "Modifying the site map layout" on page 91
- "Working with pages in the site map" on page 92
- "Saving the site map" on page 96

Viewing the site from a branch

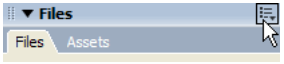
You can view the details of a specific section of a site by making a branch the focus of the site map.

To view a different branch in the site map:

1 Select the page you want to view.

2 Do one of the following:

- Right-click (Windows) or Control-click (Macintosh) in the site map, then select View as Root.
- In the *collapsed* Files panel (Window > Files), click the Options menu in the upper right corner, then select View > View as Root.



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 (–) button.

Related topics

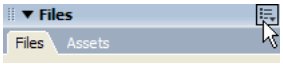
- "Viewing a site map" on page 90
- "Modifying the site map layout" on page 91
- "Working with pages in the site map" on page 92

Saving the site map

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

To create an image file of the current site map:

1 In the *collapsed* Files panel (Window > Files), click the Options menu in the upper right corner, then select File > Save Site Map.



The Save Site Map dialog box appears.

2 Enter a name in the File Name text box.

3 In the File Type pop-up menu, select .bmp or .png.

4 Select a location to save the file, then enter a name for the image.

5 Click Save.

Related topics

- "Viewing a site map" on page 90
- "Modifying the site map layout" on page 91
- "Working with pages in the site map" on page 92

Importing and exporting sites

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

Tip: It's a good idea to export your sites regularly so that you have a backup copy if anything happens to the site.

To export a site:

1 Select Site > Manage sites.

The Manage Sites dialog box appears.

2 Click the Export button.

The Export Site dialog box appears.

3 Browse to and select 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 Manage Sites dialog box.

To import a site:

1 Select Site > Manage sites.

The Manage Sites dialog box appears.

2 Click the Import button.

The Import Site dialog box 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, then the site name appears in the Manage Sites dialog box.

5 Click Done to close the Manage Sites dialog box.

Removing a Dreamweaver 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 Select Site > Manage Sites.

The Manage 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.

If you click Yes, the site name disappears from the list.

5 Click Done to close the Manage Sites dialog box.

Checking in and checking out files

If you're working in a collaborative environment, you can use the Check In/Check Out system to 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 to and from your server" on page 101).

Related topics

- "The Check In/Check Out system" on page 78

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 65).

To set up the Check In/Check Out system:

1 Select Site > Manage Sites.

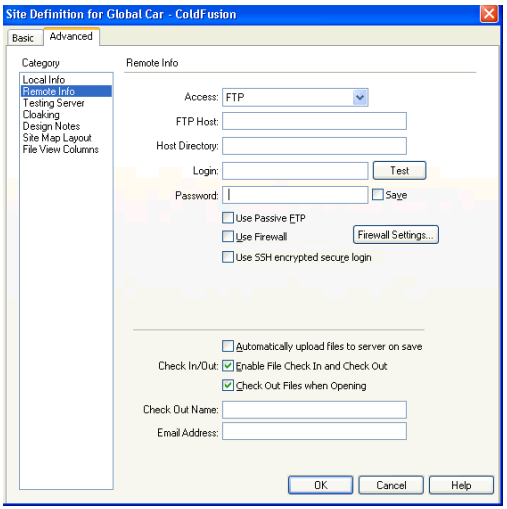
The Manage Sites dialog box appears.

2 Select a site, then click Edit.

The Site Definition dialog box appears.

3 Select Remote Info from the category list on the left.

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 65).

4 Complete the Check In/Out section.

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

5 Click OK.

Related topics

- "The Check In/Check Out system" on page 78
- "Undoing a file check-out" on page 100

Checking files into and out of a remote folder

After you set up the Check In/Check Out system (see "Setting up the Check In/Check Out system" on page 98), you can check in and check out files on a remote server using the Files panel or from the Document window.

To check out files using the Files panel:

1 In the Files panel (Window > Files), select files to check out from the remote server.

Note: You can select files in the Local or Remote view, not the Testing Server view.

A red check mark indicates that another team member has the file checked out and a lock symbol indicates that the file is read-only (Windows) or locked (Macintosh).

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

- Click the Check Out button in the Files panel toolbar.
- Right-click (Windows) or Control-click (Macintosh), then select Check Out from the context menu.

The Dependent Files dialog box appears.

3 Click Yes 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 local file's icon indicating that you have checked it out.

To check in files using the Files panel:

1 In the Files panel (Window > Files), select checked-out or new file(s).

Note: You can select files in the Local or Remote view, but not the Testing Server view.

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

- Click the Check In button in the Files panel toolbar.
- Right-click (Windows) or Control-click (Macintosh), then select Check In from the context menu.

The Dependent Files dialog box appears.

3 Click Yes 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 local file's icon indicating that the file is now read-only.

To check in or check out an open file from the Document window:

1 Make sure the file you want to check in or out is active in the Document window.

Note: You can check in only one open file at a time.

2 Do one of the following:

- Select Site > Check In or Site > Check Out.
- Click the File Management icon in the Document window toolbar, then select Check In or Check Out from the pop-up menu.

Note: If the current file is not part of the current site in the Files panel, 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, 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 preferences for the Files panel" on page 86).

Related topics

- "The Check In/Check Out system" on page 78
- "Setting up the Check In/Check Out system" on page 98

Undoing a file check-out

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

To undo a file check-out, do one of the following:

- Open the file in the Document window, then select Site > Undo Check Out.
- In the Files panel (Window > Files), right-click (Windows) or Control-click (Macintosh), then select Undo Check Out.

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

Related topics

- "The Check In/Check Out system" on page 78
- "Setting up the Check In/Check Out system" on page 98
- "Checking files into and out of a remote folder" on page 99

Getting and putting files to and from your server

If you're working in a collaborative environment, use the Check In/Check Out system to transfer files between local and remote sites (see "Checking in and checking out files" on page 97). 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.

Getting files from a remote server

Use the Get command to copy files from the remote site to your local site. You can use the Files panel or the Document window to get files.

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 get files from a remote server using the Files panel:

1 In the Files panel (Window > Files), select the desired files to download.

Usually you select these files in the Remote view, but you can select the corresponding files in the Local view if you prefer. If the Remote view is active, then Dreamweaver copies the selected 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.

Note: 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 103).

2 Do one of the following to get the file:

- Click the Get button in the Files panel toolbar.
- Right-click (Windows) or Control-click (Macintosh) the file in the Files panel, then select Get from the context menu.

The Dependent files dialog box appears.

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

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

Dreamweaver downloads the selected files, as follows:

- If you're using the Check In/Check Out system, 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 "Checking in and checking out files" on page 97).
- If you're not using the Check In/Check Out system, 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.

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

To get a file from a remote server using the Document window:

1 Make sure the document is active in the Document window.

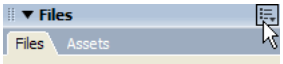
2 Do one of the following to get the file:

- Select Site > Get.
- Click the File Management icon in the Document window toolbar, then select Get from the pop-up menu.

Note: If the current file is not part of the current site in the Files panel, 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, then performs the Get operation.

To display the FTP log:

- Click the Options menu in the upper right corner of the Files panel, then select View > Site FTP Log.



Related topics

- "The Check In/Check Out system" on page 78

Putting files on a remote server

You can put 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 the Put command instead of Check In:

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

You can use the Files panel or the Document window to Put files. 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 put files on a remote or testing server using the Files panel:

1 In the Files panel (Window > Files), select the 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.

Note: 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" on page 103.

2 Do one of the following to put the file:

- Click the Put button in the Files panel toolbar.
- Right-click (Windows) or Control-click (Macintosh) the file in the Files panel, then select Put from the context menu.

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

3 If a dialog box appears, click Yes to save the file or No to put the previously saved version on the remote server.

Note: If you do not save the file, any changes you've made since the 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.

The Dependent Files dialog box appears.

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

Note: If the remote site already contains copies of the dependent files, click No.

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

To put files on a remote server using the Document window:

1 Make sure the document is active in the Document window.

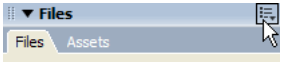
2 Do one of the following to put the file:

- Select Site > Put.
- Click the File Management icon in the Document window toolbar, then select Put from the pop-up menu.

Note: If the current file is not part of the current site in the Files panel, 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, then performs the Put operation.

To display the FTP log:

- Click the Options menu in the upper right corner of the Files panel, then select View > Site FTP Log.



Related topics

- "The Check In/Check Out system" on page 78
- "Getting files from a remote server" on page 101

Synchronizing the files on your local and remote sites

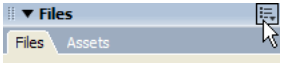
Once you've created files in your local and remote sites, you can synchronize the files between the two sites.

Note: If your remote site is an FTP server (rather than a networked server), then synchronizing your files uses FTP.

Before you synchronize your sites, you can 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:

- In the Files panel, click the Options menu in the upper right corner of the Files panel, then select Select > Select Newer Local or Select > Select Newer Remote.



- In the Files panel, right-click (Windows) or Control-click (Macintosh), then select Select > Select Newer Local or Select > Select Newer Remote.

To synchronize your files:

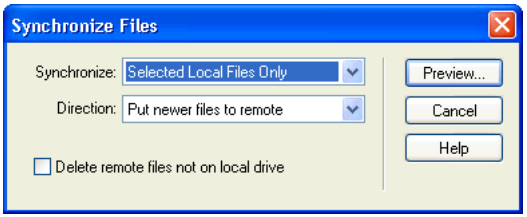
1 In the Files panel (Window > Files), select a select a site from the pop-up menu where the current site, server, or drive appears.

2 (Optional) Select specific files or folders.

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

3 Right-click (Windows) or Control-click (Macintosh), 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 you that no synchronization is necessary.

Related topics

- "Checking files into and out of a remote folder" on page 99
- "Getting files from a remote server" on page 101
- "Putting files on a remote server" on page 102

Cloaking folders and files in your site

Site cloaking enables you to exclude folders and file types in a site from sitewide operations such as Get or Put.

Related topics

- "Site cloaking" on page 78

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 Files panel (Window > Files), select a select a site from the pop-up menu where the current site, server, or drive appears.
2. Select a file or folder.
3. Right-click (Windows) or Control-click (Macintosh), then do one of the following:
   - Select Enable/Disable Cloaking.
   - Select Settings, then select Cloaking from the category list on the left in the Advanced Site Definition dialog box. Select or deselect Enable Cloaking, then click OK.

Cloaking is enabled or disabled for the site.

Related topics

- "Site cloaking" on page 78
- "Cloaking and uncloaking specific file types" on page 106
- "Uncloaking all folders and files" on page 107

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 Files panel (Window > Files), select a select a site that has site cloaking enabled from the pop-up menu where the current site, server, or drive appears.
2. Select the folder(s) you want to cloak or uncloak.
3. Right-click (Windows) or Control-click (Macintosh), then select 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.

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

Related topics

- "Site cloaking" on page 78
- "Enabling and disabling site cloaking" on page 105
- "Uncloaking all folders and files" on page 107

Cloaking and uncloaking specific file types

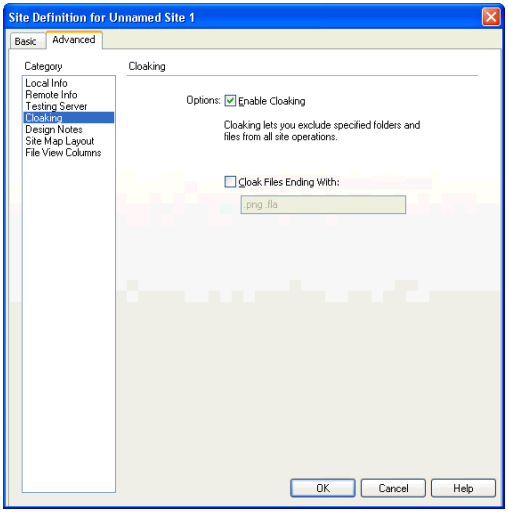
You can indicate specific file types to cloak, so that Dreamweaver cloaks all files ending with a specified pattern. For example, you can cloak all files ending with the .txt extension. The file types that you enter do not have to be file extensions; they can be any pattern at the end of a filename.

To cloak specific file types within a site:

1 In the Files panel (Window > Files), select a select a site that has site cloaking enabled from the pop-up menu where the current site, server, or drive appears.

2 Right-click (Windows) or Control-click (Macintosh), then select Settings.

The Site Definition dialog box displays the Cloaking options.



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.

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

To uncloak specific file types within a site:

1 In the Files panel (Window > Files), select a select a site that has site cloaking enabled from the pop-up menu where the current site, server, or drive appears.

2 Right-click (Windows) or Control-click (Macintosh), 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 uncloaked.

Related topics

- "Site cloaking" on page 78
- "Enabling and disabling site cloaking" on page 105
- "Cloaking and uncloaking site folders" on page 105

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.

Tip: If you want to temporarily uncloak all folders and files, then re-cloak those items, disable site cloaking (see "Enabling and disabling site cloaking" on page 105).

To uncloak all folders and files within a site:

1 In the Files panel (Window > Files), select a select a site that has site cloaking enabled from the pop-up menu where the current site, server, or drive appears.

2 Select any file or folder in that site.

3 Right-click (Windows) or Control-click (Macintosh), then select 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 uncloaked.

Related topics

- "Site cloaking" on page 78
- "Enabling and disabling site cloaking" on page 105
- "Cloaking and uncloaking site folders" on page 105
- "Cloaking and uncloaking specific file types" on page 106

Storing file information in Design Notes

Design Notes are notes associated with a file, but stored in a separate file. Use Design Notes to keep track of extra file information associated with your documents, such as image source-filenames and comments on file status.

Related topics

- "Design Notes" on page 79

Enabling and disabling Design Notes for a site

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 Select Site > Manage Sites.

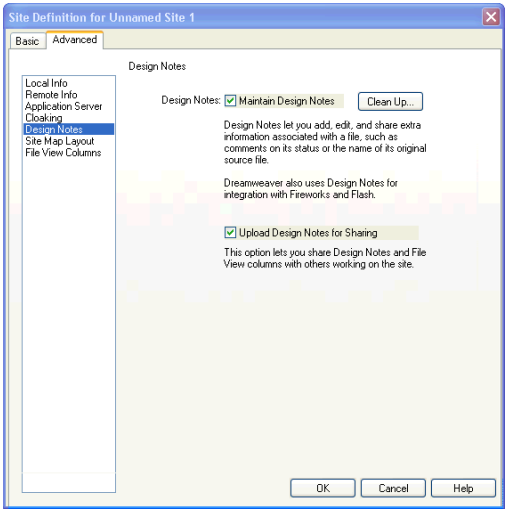
The Manage Sites dialog box appears.

2 Select a site, then click Edit.

The Site Definition dialog box appears.

3 Select Design Notes from the Category list on the left.

The Site Definition dialog box displays the Design Notes options.



4 Complete the dialog box.

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

5 Click OK.

Related topics

- "Design Notes" on page 79
- "Working with Design Notes" on page 110

Associating Design Notes with files

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 content, 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.

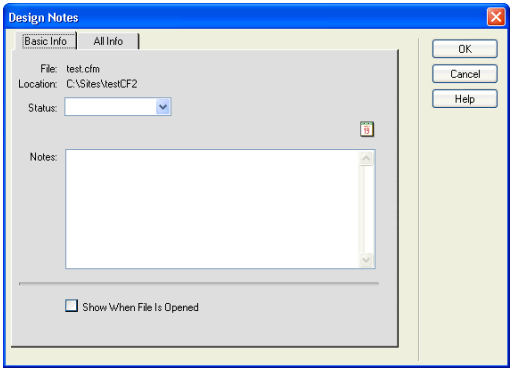
To add Design Notes to a document:

1 Do one of the following:

- Open the file in the Document window, then select File > Design Notes.
- In the Files panel, right-click (Windows) or Control-click (Macintosh) the file, then select 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 into and out of a remote folder" on page 99 or "Getting and putting files to and from your server" on page 101).

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.

Related topics

- "Design Notes" on page 79
- "Enabling and disabling Design Notes for a site" on page 108

Working with Design Notes

After you associate a Design Note with a file (see "Associating Design Notes with files" on page 109), you can open the Design Note, change its status, or delete it.

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

- Open the file in the Document window, then select File > Design Notes.
- In the Files panel, right-click (Windows) or Control-click (Macintosh) the file, then select Design Notes.
- In the Notes column of the Files 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. Select Site > Manage Sites.

   The Manage Sites dialog box appears.
2. Select the site, then click Edit.

   The Site Definition dialog box appears.
3. Select Design Notes from 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.

Related topics

- "Design Notes" on page 79
- "Enabling and disabling Design Notes for a site" on page 108

Testing your site

Dreamweaver provides several features to help you test your site, including previewing pages and checking for browser compatibility. You can also run various reports, such as a broken links report.

Site testing guidelines

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, plug-ins, or JavaScript (see "Checking for browser compatibility" on page 434). 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 378).
- 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 and testing pages in browsers" on page 270).
- Check your site for broken links, 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 for broken, external, and orphaned links" on page 336 and "Fixing broken links" on page 338).
- Monitor the file size of your pages and the time they take to download (see "Setting download time and size preferences" on page 272).

  Keep in mind that if a page consists of one large table, in some browsers, 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 "Testing your site" on page 111).
- Validate your code to locate tag or syntax errors (see "Validating your tags" on page 437).

- Once the bulk of the site has been published, continue to update and maintain it.

  Publishing your site—that is, 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 Dreamweaver includes or through an external version-control application.
- Use the Dreamweaver discussion forums found on the Macromedia website at 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.

Using reports to test your site

You can run site reports on workflow or HTML attributes, including accessibility, for the current document, selected files, or the entire site.

Workflow reports can improve collaboration among members of a web team. You can run workflow reports that display who has checked out a file, which files have Design Notes associated with them, and which files have been modified recently. You can further refine Design Note reports by specifying name/value parameters.

Note: You must have a remote site connection defined to run the workflow reports. To define a remote site, see "Setting up a remote folder" on page 65.

HTML reports enable you to compile and generate reports for several HTML attributes. You can check combinable nested font tags, accessibility, missing Alt text, redundant nested tags, removable empty tags, and untitled documents.

After you run a report, you can save it as an XML file, then import it into a template instance or a database or spreadsheet and print it, or display it on a website.

Note: You can also add different report types to Dreamweaver through the Macromedia Dreamweaver Exchange website (see "Adding extensions to Dreamweaver" on page 57).

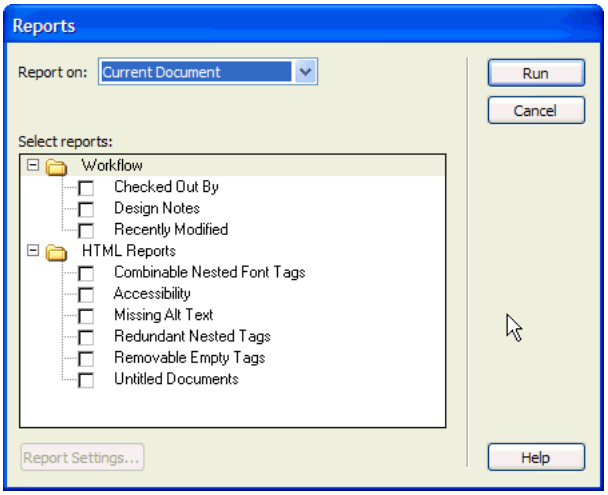
To use the Reports command to check links in your site, see "Checking for broken, external, and orphaned links" on page 336.

To run reports to test a site:

1 Select Site > Reports.

Tip: If you want to run only an accessibility report for your site, you can select File › Check Page › Check Accessibility and the report appears in the Site Reports panel of the Results panel group.

The Reports dialog box appears.



2 Select 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 the type of report you run, you might 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 Run a report (see the previous procedure).

2 In the Site Reports panel, do any of the following to view the report:

- Click the column heading you want to sort results by.

  You can sort by filename, line number, or description. You can also run several different reports and keep the different reports open.

- Select any line in the report, then 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).

- 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 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.

Tip: After running HTML reports, use the Clean Up HTML command to correct any HTML errors the reports listed (see "Cleaning up your code" on page 433).

Related topics

- "Reports in Dreamweaver" on page 38

CHAPTER 5
Managing Site Assets and Libraries

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. You can use Macromedia Dreamweaver MX 2004 to manage your site assets. You can 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 also drag an asset directly to your current document to insert it in a page.

Dreamweaver 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. 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. (For more information, see Chapter 11, "Managing Templates," on page 215.)

The following topics are covered in this chapter:

- "About library items" on page 115
- "Working with assets" on page 116
- "Creating and managing a list of favorite assets" on page 123
- "Working with library items" on page 125

About 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. These assets in a library are called library items. 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 files in a library.

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.

Dreamweaver stores library items in a Library folder within the local root folder for each site. Each site has its own library.

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.

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. That is, 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. Then, if you need to change some text for example, 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.

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.

There are special requirements for editing the behaviors in library items (see "Editing a behavior in a library item" on page 130). Library items cannot contain style sheets, because the code for those elements is part of the `head` section.

Related topics

- "Working with library items" on page 125

Working with assets

Assets include 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 copy them from a clip-art CD or graphics website.

Related topics

- "Creating and managing a list of favorite assets" on page 123
- "Working with library items" on page 125

Viewing assets in the Assets panel

You can view and manage assets in the current site using the Assets panel. The Assets panel displays assets for the site associated with the active document in the Document window.

Note: You must define a local site before you can view assets in the Assets panel. For more information about setting up a site, see "Setting up a new Dreamweaver site" on page 62.

The Assets panel provides two views:

The Site list shows all of the assets in your site, including colors and URLs that are used in any document in your site.

The Favorites list shows only the assets you've explicitly chosen.

In both lists, assets are divided into categories (along the left side of the Assets panel). The Site list and the Favorites list are both available for all categories of assets *except* templates and library items.

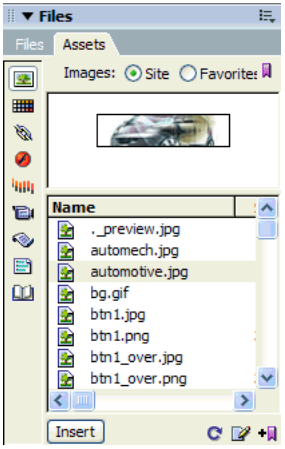
Note: Most of the Assets panel operations work the same in the Site list and in the Favorites list. There are a few tasks, however, that you can perform only in the Favorites list (see "Creating and managing a list of favorite assets" on page 123).

By default, assets in a given category are listed in alphabetical order by name. You can instead list the assets by any of several other criteria or change the size of the columns. You can also preview assets in a category, and change the size of the preview area.

To open the Assets panel:

- Select Window > Assets.

  The Assets panel appears. By default, the Images category is selected.



To view the Site list:

- In the Assets panel (Window > Assets), select the Site option at the top of the panel.

  Note: In the Templates and Library categories (on the left side of the Assets panel), the Site and Favorites choices are unavailable.

To view the Favorites list:

- In the Assets panel (Window > Assets), select the Favorites option at the top of the panel.

  The Favorites list is empty until you explicitly add assets to it.

  Note: In the Templates and Library categories (on the left side of the Assets panel), the Site and Favorites choices are unavailable.

To display assets in a particular category:

- Click the appropriate icon on the left side of the Assets panel (Window > Assets).

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:

- Drag the line that separates two column headings.

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.

Related topics

- "Adding an asset to a document" on page 119
- "Selecting and editing assets" on page 121

Refreshing the Assets panel

It can take a few seconds for the Assets panel to create the Site list because it needs 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 In the Assets panel (Window > Assets), select the Site option at the top of the panel to see the Site list.

2 Click the Refresh Site List button at the bottom of the Assets panel.

Dreamweaver creates the site cache or updates it as necessary, and the Assets panel updates to show the site assets.

To manually rebuild the site cache and refresh the Site list:

- In the Assets panel (Window > Assets), right-click (Windows) or Command-click (Macintosh) in the Assets list, then select Refresh Site List.

Related topics

- "Selecting and editing assets" on page 121
- "Reusing assets in another site" on page 122

Adding an asset to a document

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.)

To insert an asset in a document:

1 Place the insertion point in the Design view where you want the asset to appear.

2 In the Assets panel (Window > Assets), select the category on the left side of the panel for the type of asset you want to insert.

Note: Select any category except Templates. A template can be applied only to an entire document; it can't be inserted into a document. For more information about templates, see Chapter 11, "Managing Templates," on page 215.

3 Select either Site or Favorites at the top of the panel, then select the asset you want to insert.

There are no Site or Favorites lists for library items; skip this step if you're inserting a library item.

4 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, select View > Head Content.

- Select the asset in the panel, then click Insert at the bottom of the panel.

The asset is inserted in the document. (If the asset is a color, it applies starting at the insertion point; that is, subsequent typing appears in that color.)

Related topics

- "Applying a URL to an image or text using the Assets panel" on page 120
- "Selecting and editing assets" on page 121
- "Reusing assets in another site" on page 122

Applying a color to text using the Assets panel

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 "Adding assets to and removing assets from the Favorites list" on page 123.

To change the color of selected text in a document:

1 Select the text in the document.

2 In the Assets panel (Window > Assets), select the Colors category on the left side of the panel.

3 Select the desired color.

4 Click the Apply button at the bottom of the panel.

Related topics

- "Adding an asset to a document" on page 119
- "Selecting and editing assets" on page 121
- "Reusing assets in another site" on page 122

Applying a URL to an image or text using the Assets panel

You can use the Assets panel to make the selected text or image an active link.

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 In the Assets panel (Window > Assets), select the URLs category on the left side of the panel.

3 Select the desired URL.

4 Do one of the following:

- Drag the URL from the panel to the selection in the Design view.
- Select the URL, then click the Insert button.

Related topics

- "Adding an asset to a document" on page 119
- "Applying a color to text using the Assets panel" on page 120
- "Reusing assets in another site" on page 122

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:

1 In the Assets panel (Window > Assets), select one of the assets.

2 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 In the Assets panel (Window > Assets), do one of the following:

- Double-click the asset.
- Select the asset, then click the Edit button at the bottom of the panel.

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.

Note: If an external editor doesn't open for an asset that needs to use an editor, select Edit › Preferences (Windows) or Dreamweaver › Preferences (Macintosh), select the File Types/Editors category, and make sure you have an external editor defined for the asset's file type. (See "Starting an external editor for media files" on page 359.)

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.

  Note: To dismiss the color picker without picking a color, press Escape.

Related topics

- "Viewing assets in the Assets panel" on page 117
- "Refreshing the Assets panel" on page 118

Reusing assets in another site

The Assets panel shows all the assets (of recognized types) in your current site. 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.

You might need to locate the file in the Files panel that corresponds to an asset in the Assets panel before you transfer the asset to or from your remote site.

Note: The Files panel might 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 locate an asset's file in the Files panel:

1. In the Assets panel (Window > Assets), select the category on the left side of the panel for the type of asset you want to find.
2. Right-click (Windows) or Control-click (Macintosh) the asset's name or icon in the Assets panel, then select Locate in Site from the context menu.

   Note: Locate in Site is unavailable for colors and URLs, which do not correspond to files in the site.

   The Files panel opens, with the asset file selected. The Locate in Site command locates the file corresponding to the asset itself; it does not locate a file that uses that asset.

To copy assets from the Asset panel's Site list or Favorites list to another site:

1. In the Assets panel (Window > Assets), select the category on the left side of the panel for the type of asset you want to copy.
2. Right-click (Windows) or Control-click (Macintosh) the asset or assets to copy, select Copy to Site, then select the target site name from the submenu. (The submenu lists all the sites you've defined.)

   Note: 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 125.

   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.

Related topics

- "Viewing assets in the Assets panel" on page 117
- "Refreshing the Assets panel" on page 118
- "Selecting and editing assets" on page 121

Creating and managing a list of favorite assets

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. You can add frequently used assets to a Favorites list, 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.

Most of the operations of the Assets panel are the same in the Favorites list as in the Site list. However, there are several tasks that you can perform only 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 in the Assets panel.

Adding a color or URL to the Favorites list requires an extra step. 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.

Note: There are no Favorites lists for templates and library items.

To add assets to your Favorites list, do one of the following:

- Select one or more assets in the Site list of the Assets panel, then click the Add to Favorites button at the bottom of the panel.
- Select one or more assets in the Site list of the Assets panel, right-click (Windows) or Control-click (Macintosh), then select Add to Favorites.
- Select one or more files in the Files panel, right-click (Windows) or Control-click (Macintosh), then select Add to Favorites.

  Dreamweaver ignores files that don't fit a category in the Assets panel.
- Right-click (Windows) or Control-click (Macintosh) an element in the Document window's Design view, then select the context menu command to add the element to the appropriate Favorites category.

  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.

To add a new color or URL to the Favorites list:

1. In the Assets panel (Window > Assets), select the Colors or URLs category on the left side of the panel.
2. Select the Favorites option at the top of the panel to show the Favorites list.

3. Click the New Color or New URL button at the bottom of the panel.
4. Do one of the following
   - Select a color using the color picker, then give the color a nickname if desired (see "Creating a nickname for a favorite asset" on page 124).

     To exit from the color picker without selecting 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 260.
   - Enter a URL and a nickname in the Add New URL dialog box, then click OK.

To remove assets from your Favorites list:

1 In the Assets panel (Window > Assets), select the Favorites option at the top of the panel.

2 Select one or more assets (or a folder) in the Favorites list.

3 Click the Remove From Favorites button at the bottom of the 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.

Related topics

- "Viewing assets in the Assets panel" on page 117
- "Creating a nickname for a favorite asset" on page 124

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 to assets in the Favorites list in the Assets panel. 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 In the Assets panel (Window > Assets), select the category on the left side of the panel that contains your asset.

2 Select the Favorites option at the top of the panel 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, then select Edit Nickname.
- Click the asset's name once, pause, then click it again.

4 Type a nickname for the asset, then press Enter (Windows) or Return (Macintosh).

The nickname appears in the Nickname column.

Related topics

- "Viewing assets in the Assets panel" on page 117
- "Adding assets to and removing assets from the Favorites list" on page 123

Grouping assets in a Favorites folder

You can group assets in your Favorites list in folders in the Assets panel. 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 In the Assets panel (Window > Assets) panel, select the Favorites option at the top of the panel.

2 Click the New Favorites Folder button at the bottom of the panel.

3 Type a name for the folder, then press Enter (Windows) or Return (Macintosh).

4 Drag assets into the folder.

Related topics

- "Viewing assets in the Assets panel" on page 117
- "Adding assets to and removing assets from the Favorites list" on page 123
- "Creating a nickname for a favorite asset" on page 124

Working with 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 website. These elements are called library items.

Related topics

- "About library items" on page 115

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 based on a selection:

1 In the Document window, select a portion of a document to save as a library item.

2 Do one of the following:

- Drag the selection into the Library category of the Assets pane (Window > Assets).
- Click the New Library Item button at the bottom of the Library category of the Assets panel (Window > Assets).

- Select Modify > Library > Add Object to Library.

3 Type a name for the new library item, then press Enter (Windows) or Return (Macintosh).

Dreamweaver saves each library item as a separate file (with the file extension .lbi) in the Library folder of the site's local root folder.

To create an empty library item:

1 Make sure nothing is selected in the Document window.

If something is selected, it will be placed in the new library item.

2 In the Assets panel (Window > Assets), select the Library category on the left side of the panel.

3 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.

4 While the item is still selected, enter a name for it, then press Enter (Windows) or Return (Macintosh).

Related topics

- "Editing a library item" on page 127
- "Setting library highlighting preferences" on page 129

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 In the Assets panel (Window > Assets), select the Library category on the left side of the panel.

3 Do one of the following:

- Drag a library item from the Assets panel to the Document window.

  Tip: 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 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.

- Select a library item, then click the Insert button at the bottom of the panel.

Related topics

- "Creating a library item" on page 125
- "Editing the properties for a library item" on page 129
- "Making library items editable in a document" on page 130

Editing a library item

When you edit a library item, you can 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.

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 is unavailable when you are editing a library item, since library items can contain only `body` elements 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 In the Assets panel (Window > Assets), select the Library category on the left side of the panel.

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 Do one of the following:

- Click the Edit button at the bottom of the panel.
- 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, then save your changes.

5 In the dialog box that appears, select whether to update the documents on the local site that uses the edited library item:

- Select Update to update all documents in the local site with the edited library item.
- Select 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:

- Select Modify > Library > Update Current Page.

To update the entire site or all documents that use a particular library item:

1 Select Modify > Library > Update Pages.

The Update Pages dialog box appears.

2 In the Look In pop-up menu, do one of the following:

- Select Entire Site, then select 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.
- Select Files That Use, then select 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.

Tip: To update templates at the same time, make sure Templates is also selected. For more information, see "Opening a template for editing" on page 242.

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:

1 In the Assets panel (Window > Assets), select the Library category on the left side of the panel.

2 Select the library item you want to rename, pause, then 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 In the Assets panel (Window > Assets), select the Library category on the left side of the panel.

2 Select the library item you want to delete.

3 Do one of the following:

- Click the Delete button at the bottom of the panel, then confirm that you want to delete the item.
- Press the Delete key, then confirm that you want to delete the item.

Caution: 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.

Dreamweaver removes the library 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 (Window > Properties).

Related topics

- "Creating a library item" on page 125
- "Making library items editable in a document" on page 130
- "Editing a behavior in a library item" on page 130

Setting library highlighting preferences

You can customize the highlight color for library items and show or hide highlighting of library items by setting Highlighting preferences.

To change the highlight color for library items:

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

The Preferences dialog box appears.

2 Select the Highlighting category from the list on the left.

3 Click the Library Items color box, then select a highlight color using the color picker (or enter the hexadecimal value for the highlight color in the text box).

For information about using the color picker, see "Working with colors" on page 260.

4 Select Show to display the highlighted library color in the Document window.

5 Click OK.

To view highlight colors in the Document window:

- Select View > Visual Aids > Invisible Elements.

  Highlight colors appear in the document window only when View > Visual Aids > Invisible Elements is enabled and the appropriate options are enabled in Highlighting preferences.

Related topics

- "Creating a library item" on page 125
- "Editing a library item" on page 127
- "Editing the properties for a library item" on page 129

Editing the properties for a library item

You can use the Property inspector for a library item to open a library item for editing, detach a selected library item from its source file, or overwrite an item with the currently selected library item.

To edit the properties for a library item:

1 Select a library item in a document.

2 Click one of the buttons in the Property inspector (Window > Properties), depending on what you want to do.

For more information, click the Help icon in the Property inspector.

Related topics

- "Creating a library item" on page 125
- "Inserting a library item in a document" on page 126
- "Setting library highlighting preferences" on page 129

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 (Window > Properties).

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.

Related topics

- "Creating a library item" on page 125
- "Inserting a library item in a document" on page 126
- "Editing a library item" on page 127
- "Editing the properties for a library item" on page 129

Editing a behavior in a 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 371.

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 later.

2 Select the library item, then click Detach from Original in the Property inspector (Window > Properties).

3 Select the element that has the behavior attached to it.

4 In the Behaviors panel (Window > Behaviors), double-click the action you want to change.

5 In the dialog box that appears, make the necessary changes, then click OK.

6 In the Assets panel (Window > Assets), select the Library category on the left side of the panel.

7 Make sure you have recorded the exact name of the original library item; select the original library item, then click the Delete button at the bottom of the 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, then give the new item the same name as the item you deleted.

Be sure to use exactly the same spelling and capitalization.

10 To update the library item in your site's other documents, select 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, select the name of the library item you just created.

13 In the Update option, make sure Library Items is selected, then click Start.

14 When the updates are complete, click Close to exit the Update Pages dialog box.

Related topics

- "Creating a library item" on page 125
- "Inserting a library item in a document" on page 126
- "Editing a library item" on page 127

CHAPTER 6
Managing Contribute Sites with Dreamweaver

Macromedia Contribute combines a web browser with a basic web-page editor. It enables your colleagues or clients to browse to a page in a site that you created, and to edit or update the page if they have permission to do so. Contribute users can add and update basic web content, including formatted text, images, tables, and links. Contribute site administrators can limit what ordinary (non-administrator) users can do in a site.

Note: Most Macromedia Dreamweaver MX 2004 users who connect to a Contribute site are site administrators; this chapter assumes that you are a site administrator.

Before you give Contribute users the connection information they need to edit pages, you should perform various site-setup tasks in Dreamweaver. Then, you can export your site definition information as a connection key, a file that Contribute users can use to connect to the site.

Note: This chapter assumes that you understand how to set up a Dreamweaver site. For more information, see "Setting Up a Dreamweaver Site" on page 59. If you are setting up a dynamic site to use with Contribute, see Administering Contribute Help in Contribute for more information.

You can connect to a Contribute site using Dreamweaver and you can modify the files in the site just as you would in any Dreamweaver site.

This chapter contains the following sections:

- "About managing Contribute sites" on page 134
- "Preparing a site for use with Contribute" on page 138
- "Administering a Contribute site using Dreamweaver" on page 138
- "Troubleshooting a Contribute site" on page 140
- "Managing Contribute files using Dreamweaver" on page 141

About managing Contribute sites

You can use Dreamweaver to connect to a Contribute site and modify the files in the site just as you would in any Dreamweaver site. Most Dreamweaver capabilities work the same way with a Contribute site as with any other site. There are only a few aspects of working with files in a Contribute site that differ from working with files in other sites.

Site structure and page design for a Contribute site

To prepare your website to be edited by Contribute users, structure the site appropriately.

Create the folders that Contribute users should use for their pages, create index pages for those folders, and add basic navigation to the pages. You might want to create user-specific folders for users to practice in. You might also want to create CSS style sheets that define the styles for each page or folder. You can do some of this setup work in Contribute, but it's easier to set up a site using Dreamweaver.

In addition, you can create templates that Contribute users can use to create new pages (see "Creating templates for a Contribute site" on page 229).

The following suggestions may help you create a site that Contribute users can easily contribute to:

- Keep your site structure simple.

  Don't nest folders too deeply. Group related items together in a folder.
- Set up appropriate read and write permissions for the folders on the server.

  For more information, see "Enabling Contribute users to access templates without root folder access" on page 142.
- As you create your folder structure, add index pages to folders to encourage Contribute users to place new pages in the correct folders.

  For example, if Contribute users will be providing pages containing meeting minutes, you can create a folder in the site root folder named meeting_minutes, and create an index page in that folder. Then you can provide a link from your site's main page to the index page for meeting minutes. A Contribute user can then navigate to that index page and create a new page of minutes for a specific meeting, linked from that page; the new page is automatically created inside the meeting_minutes folder.
- On each folder's index page, provide a list of links to the individual content pages and documents in that folder.
- Keep page designs as simple as possible, minimizing fancy formatting.
- Name your CSS styles clearly.

  If the Contribute users working on your site use a standard set of styles in Microsoft Word, name your CSS styles with the same names as the corresponding Word styles, so that Contribute can map the styles when a user copies information from a Word document and pastes it into a Contribute page.

- To prevent a CSS style from being available to Contribute users, change the name of the style so that the name starts with mmhide_.

  For example, if you use a style named RightJustified in a page but you don't want Contribute users to be able to use that style, rename the style to mmhide_RightJustified.

  Note: You have to add mmhide_ to the style name in Code view; you cannot add it in the CSS panel.

- Use CSS rather than HTML tags.

  Contribute recognizes CSS styles and allows users to apply them.

- Use as few CSS styles as possible, to keep things simple and clean.
- If you use server-side includes for HTML page elements, such as headers or footers, create an unlinked simple HTML page that contains links to the include files.

  Contribute users can then bookmark that page and use it to navigate to the include files and edit them.

Related topics

- Chapter 2, "Setting Up a Dreamweaver Site," on page 59
- "Creating new documents" on page 70
- "Using Cascading Style Sheets styles" on page 290
- "Working with server-side includes" on page 448

Site connection for Contribute compatibility

In Dreamweaver, you can connect to an existing Contribute site and modify files in the site just as you would in any Dreamweaver site ("Setting up a new Dreamweaver site" on page 62). When you connect to a site that's been set up as a Contribute site (and that already has an administrator), Dreamweaver prompts you to enable Contribute compatibility.

If you are preparing an existing Dreamweaver site for Contribute users, then you need to explicitly enable Contribute compatibility to use Contribute-related features; Dreamweaver will not prompt you. For more information, see "Preparing a site for use with Contribute" on page 138.

Dreamweaver allows you to connect to a remote site, including a Contribute site, in a variety of ways, but not all connection types support Contribute compatibility. In particular, the following restrictions apply to connection types:

- If you're connecting to your remote site using WebDAV or SourceSafe, you can't enable Contribute compatibility, because those source-control systems aren't compatible with the Design Notes and Check In/Check Out systems that Dreamweaver uses for Contribute sites.
- If you use RDS to connect to your remote site, you can enable Contribute compatibility, but you then must customize your connection before you can share it with Contribute users.
- If you're using your local computer as a web server, you must set up the site using an FTP or network connection to your computer (rather than just a local folder path) to be able to share your connection with Contribute users.

File transfer to or from a Contribute site

Contribute uses a system much like the Dreamweaver Check In/Check Out system to ensure that only one user at a time can edit a given web page. When you enable Contribute compatibility in Dreamweaver, the Dreamweaver Check In/Check Out system is automatically enabled.

To transfer files to and from a Contribute site using Dreamweaver, always use the Check In and Check Out commands. If you instead use the Put and Get commands to transfer files, you may overwrite the modifications that a Contribute user has recently made to a file.

Note: If you do use the Put command in a Contribute site, Dreamweaver automatically checks the file in and then checks it out again, to reduce the chance that your changes will conflict with another user's.

Checking a file out from a Contribute site works just as it does in any other site.

When you check a file in to a Contribute site, Dreamweaver automatically makes a backup copy of the previously checked in version of the file in the _baks folder and adds your user name and a date stamp to a Design Notes file so others can see who checked in the file and when. For more information about automatic backup copies, see "Enabling Contribute users to access templates without root folder access" on page 142.

Related topics

- "Checking in and checking out files" on page 97
- "Preparing a site for use with Contribute" on page 138

Contribute file and folder permissions on the server

Contribute provides a way to manage file and folder permissions for each group of users that you define. For more information, see Administering Contribute Help in Contribute. Dreamweaver users are not affected by these permissions, but Contribute enforces these permissions for Contribute users.

However, Contribute doesn't provide a way to manage the underlying read and write permissions assigned to files and folders by the server. You can manage those permissions directly on the server.

You can think of Contribute permission groups as an overlay on the server's read and write permissions; for example, if a user doesn't have write permission to a folder on the server, then they can't save to that folder even if they're a member of a permission group that's allowed (according to Contribute permissions) to write to that folder.

If a Contribute user doesn't have read access on the server to a dependent file, such as an image displayed in a page, the contents of the dependent file don't appear in the Contribute window. For example, if a user doesn't have read access to an images folder, the images in that folder appear as broken image icons in Contribute. Similarly, Dreamweaver templates are stored in a subfolder of the site's root folder, so if a Contribute user doesn't have read access to the root folder, they can't use templates in that site unless you copy the templates to an appropriate folder.

When you set up a site, it's a good idea to give users read access on the server to the /_mm folder (the _mm subfolder of the root folder), the /Templates folder, and all of the folders containing assets that they will need to use.

If for some reason you can't give users read access to the /Templates folder, see "Enabling Contribute users to access templates without root folder access" on page 142 to supply users with templates.

Contribute special files

Contribute uses a variety of special files that aren't intended to be viewed by visitors to your site. These files include:

- The shared settings file, an XML file with a name like contribute.xml, which appears in a folder named _mm in the root folder of the site, and contains information that Contribute uses to manage the site
- Older versions of files, in folders named _baks (see "Enabling Contribute users to access templates without root folder access" on page 142)
- Temporary versions of pages, so that users can preview changes
- Temporary lock files, indicating that a given page is being edited or previewed
- Design Notes files containing metadata about the site's pages

In general, you shouldn't edit any of these Contribute special files using Dreamweaver; Dreamweaver manages them automatically.

If you don't want these Contribute special files to appear on your publicly accessible server, you can set up a staging server where Contribute users work on pages. Then you can periodically copy those web pages from the staging server to a production server that's on the web. If you take this staging-server approach, copy only web pages to the production server, not any of the above-listed Contribute special files. In particular, don't copy the _mm and _baks folders to the production server.

Note: For information about setting up a server to prevent visitors from seeing files in folders that begin with an underscore, see "Website security" in Administering Contribute Help in Contribute.

If you aren't using a staging server, you may want to manually delete old revisions in _baks folders now and then to ensure that those older file versions can't be viewed by visitors. In some circumstances, you might need to manually delete other Contribute special files.

For example, there might be circumstances in which Contribute fails to delete temporary preview pages when the user is finished previewing; in that case, you would have to manually delete those temporary pages. Temporary preview pages have filenames that begin with TMP.

Similarly, under some circumstances an outdated lock file may be accidentally left on the server. If that happens, you must manually delete the lock file to allow others to edit the page. For information on how to delete the lock file, see "Unlocking a file in a Contribute site" on page 142.

Preparing a site for use with Contribute

Before you can use Dreamweaver to administer a Contribute site, you must enable Contribute compatibility. Depending on the site you are connecting to, you might be automatically prompted to enable Contribute compatibility.

Note: When you enable Contribute compatibility, Dreamweaver automatically enables Design Notes (including the Upload Design Notes for Sharing option) and the Check In/Check Out system.

To enable Contribute compatibility for a defined Dreamweaver site:

1. Select Site > Manage Sites.

   The Manage Sites dialog box appears.
2. Select a site, then click Edit.

   The Site Definition dialog box appears.
3. Click the Advanced tab.
4. Select the Contribute category from the Category list on the left.
5. Select the Enable Contribute Compatibility option and complete the dialog box.

   For more information, click the Help button.
6. Click OK to close the Site Definition dialog box.
7. Click Done to close the Manage Sites dialog box.

Related topics

- "Site connection for Contribute compatibility" on page 135

Administering a Contribute site using Dreamweaver

After you enable Contribute compatibility (see "Preparing a site for use with Contribute" on page 138), you can use Dreamweaver to start Contribute to perform site administration tasks.

Note: You must have Contribute installed on the same machine as Dreamweaver.

As an administrator of a Contribute site, you can do any of the following:

- Change the sitewide settings for the site.

  Sitewide settings include administrator password, administrator contact e-mail, or number of revisions saved on the server.
- Change the permissions granted to groups of Contribute users.
- Set up Contribute users.

  Contribute users need certain information about the site to be able to connect to it. You can package all of this information in a file called a *connection key* to send to Contribute users.

  Note: A connection key is not the same as a Dreamweaver exported site file. To export site information for use with Dreamweaver, see"Importing and exporting sites" on page 96.

  Tip: Before you give Contribute users the connection information they need to edit pages, you should use Dreamweaver to create the basic folder hierarchy for your site (see "Site structure and page design for a Contribute site" on page 134), and to create any templates and CSS style sheets needed for the site (see "Creating templates for a Contribute site" on page 229).

To administer a Contribute website in Dreamweaver:

1 Select Site > Manage Sites.

The Manage Sites dialog box appears.

2 Select a site, then click Edit.

The Site Definition dialog box appears.

3 Click the Advanced tab.

4 Select the Contribute category from the category list on the left.

5 Click the Administer Site in Contribute button.

Note: This button does not appear if you have not enabled Contribute compatibility. For information about how to enable this option, see "Preparing a site for use with Contribute" on page 138.

If the site is password-protected, a dialog box appears asking for your user name and password. Type your user name and password, then click OK.

The Administrator Password dialog box appears.

6 Type the administrator password, then click OK.

If the site has no administrator, a dialog box appears asking if you want to become the website's administrator. Click Yes, then type and confirm an administrator password for the website, and click OK.

The Administer Website dialog box appears.

7 Do any of the following:

- To change administration settings, click the Sitewide Settings button, then change settings as necessary in the Sitewide Settings dialog box.
- To change permission-group settings, select a permission group (if necessary) and click one of the buttons in the Permission Groups sections, then complete the dialog box.
- To send a connection key to set up users, click the Send Connection Key button, then complete the Connection Wizard.

  Note: For more information about sitewide settings, managing users and permissions, or creating a connection key, see Administering Contribute Help in Contribute.

8 Click OK to close the Administer Website dialog box.

9 Click OK to close the Site Definition dialog box.

10 Click Done to close the Edit Sites dialog box.

Related topics

- "Managing Contribute files using Dreamweaver" on page 141

Troubleshooting a Contribute site

If you encounter a problem with a Contribute site, see the following specific topics for information on how to resolve the problem:

- For problems connecting to a Contribute site, see "Troubleshooting connection problems for a Contribute site" on page 140.
- For problems using the administration tools, see "Troubleshooting the Contribute administration tools" on page 140.

For more troubleshooting information, see the troubleshooting section in Administering Contribute Help in Contribute.

Troubleshooting connection problems for a Contribute site

When you click any button related to Contribute site administration, Dreamweaver verifies that it can connect to your remote site and that the Site Root URL you've given for the site is valid. If Dreamweaver can't connect, or if the URL isn't valid, an error message appears.

To check a Contribute connection:

1. Check the Site Root URL in the Contribute category of the Site Definition dialog box by opening that URL in a browser, to make sure that the correct page opens.
2. Use the Test button in the Remote Info category of the Site Definition dialog box to make sure that you can connect to the site.
3. If the URL is correct but the Test button results in an error message, ask your system administrator for help.

Troubleshooting the Contribute administration tools

If the administration tools aren't working properly, there might be something wrong with the _mm folder.

To check the _mm folder:

1. On the server, make sure that you have read and write permission for the _mm folder.
2. Make sure that the _mm folder contains a shared settings file with a name like contribute.xml.
3. If it doesn't, use the Connection Wizard to create a connection to the site and to become an administrator for the site.

   The shared settings file is created automatically when you become an administrator. For more information about becoming an administrator for an existing Contribute website, see Administering Contribute Help in Contribute.

Related topics

- "Troubleshooting connection problems for a Contribute site" on page 140

Managing Contribute files using Dreamweaver

Most Dreamweaver capabilities work the same way with a Contribute site as with any other site. However, when you use Dreamweaver with a Contribute site, Dreamweaver automatically performs certain file-management operations, such as saving multiple revisions of a document.

This section describes the aspects of working with files in a Contribute site that differ from working with files in other sites.

For information about transferring files to and from a Contribute site, see "File transfer to or from a Contribute site" on page 136. For information about Contribute special files, see "Contribute special files" on page 137.

Note: A Contribute site administrator may assign users to groups called *permission groups*, and may specify what actions members of each permission group can perform. When you use Dreamweaver to manage files in a Contribute site, permission-group restrictions don't apply to you; the only restrictions that apply are read and write permissions on the server. For more information about the different kinds of permissions, see "Contribute file and folder permissions on the server" on page 136.

Deleting, moving, or renaming a remote file in a Contribute site

Deleting a file from the remote server that hosts a Contribute site works much like deleting a file from the server for any Dreamweaver site. However, when you delete a file from a Contribute site, Dreamweaver asks whether to delete all older versions of the file. If you choose to keep the older versions, Dreamweaver saves a copy of the current version as well so you can restore it later.

Renaming a remote file or moving it from one folder to another in a Contribute site works the same way it works in any Dreamweaver site. In a Contribute site, Dreamweaver additionally renames or moves the associated previous versions of the file that are saved in the _baks folder.

To delete a remote file:

1 Select the file in the remote pane of the Files panel (Window > Files), then press Backspace (Windows) or Delete (Macintosh).

A dialog box appears asking you to confirm that you want to delete the file.

2 If the Delete Rollback Versions option appears in the confirmation dialog box, do one of the following:

- To delete all previous versions of the file as well as the current version, select the Delete Rollback Versions option.
- To leave previous versions on the server, deselect the Delete Rollback Versions option.

3 Click Yes to delete the file.

The file is deleted. If you chose to delete previous versions, they are deleted as well. If you chose not to delete previous versions, a copy of the current version is saved in the _baks folder as a new revision of the file.

Related topics

- "Unlocking a file in a Contribute site" on page 142

Enabling Contribute users to access templates without root folder access

In a Contribute site, you must manage underlying read and write permissions assigned to files and folders by the server directly on the server. For more information, see "Contribute file and folder permissions on the server" on page 136.

When you set up a site, it's a good idea to give users read access on the server to the /_mm folder (the _mm subfolder of the root folder), the /Templates folder, and all of the folders containing assets that they will need to use.

If for some reason you can't give users read access to the /Templates folder, you can still make the templates available to users.

To allow Contribute users to use templates without read access to the main site's root folder:

1. Set up the Contribute site so that its root folder is the folder you want users to see as the root.
2. Manually copy the template folder from the main site's root folder into the Contribute site's root folder, using the Files panel.
3. After you update templates for the main site, recopy the changed templates into appropriate subfolders as needed.

If you take this "subsite" approach, don't use site-root-relative links in the subsites. Site-root-relative links are relative to the main root folder on the server, not to the root folder you define in Dreamweaver. Contribute users can't create site-root-relative links. For more information about site-root-relative links, see "Understanding document locations and paths" on page 316.

If links in a Contribute page appear to be broken, it's possible that there's a problem with folder permissions, particularly if the links link to pages outside of the Contribute user's root folder. Check read and write permissions for folders on the server.

Unlocking a file in a Contribute site

Sometimes a remote file in a Contribute site appears to be checked out, but the file isn't actually locked on the user's computer. When this happens, unlock the file to enable users to edit it.

Note: Before following this procedure, be sure that the file really isn't checked out. If you unlock a file while a Contribute user is editing it, multiple users might edit the file at once, which can result in loss of data.

To unlock a checked-out file:

1. Do one of the following:
   - Open the file in the Document window, then select Site > Undo Check Out.
   - In the Files panel (Window > Files), right-click (Windows) or Control-click (Macintosh), then select Undo Check Out.

   A dialog box might appear, indicating who has the file checked out and asking you to confirm that you want to unlock the file.
2. If the dialog box appears, click Yes to confirm.

   The file is unlocked on the server.

PART III
Laying Out Pages

Use the visual design tools in Macromedia Dreamweaver MX 2004 to create sophisticated page layouts.

This part contains the following chapters:

Chapter 7, “Laying Out Pages with CSS,” on page 145

Chapter 8, “Presenting Content with Tables,” on page 163

Chapter 9, “Laying Out Pages in Layout Mode,” on page 183

Chapter 10, “Using Frames,” on page 199

Chapter 11, “Managing Templates,” on page 215

CHAPTER 7
Laying Out Pages with CSS

In Macromedia Dreamweaver MX 2004, you can use CSS to add content blocks to your document for layout. You can either insert `div` tags and apply CSS positioning styles to them, or you can use Dreamweaver layers to create your layout.

Note: Dreamweaver treats all `div` tags with absolute or relative position as layers.

Tip: You can use a Dreamweaver design file as a starting point for your CSS layout. Select a file from the Page Designs (CSS) category of the New Document dialog box (see "Creating a document based on a Dreamweaver design file" on page 70).

Whether you use CSS, tables, or frames to lay out your pages, Dreamweaver has rulers and grids for visual guidance in your layout. Dreamweaver also has a tracing image feature, which you can use to re-create a page design that was created in a graphics application.

Note: If you're unfamiliar with using layers and Cascading Style Sheets (CSS), but are familiar with using tables, try using tables or Layout mode for page layout (see "Presenting Content with Tables" on page 163 and "Laying Out Pages in Layout Mode" on page 183).

This chapter contains the following sections:

- "About layers in Dreamweaver" on page 146
- "Inserting content blocks for layout" on page 147
- "Changing highlight color for content blocks" on page 148
- "Working with content blocks for layout" on page 149
- "Inserting a layer" on page 150
- "Setting layer preferences and properties" on page 152
- "Managing layers" on page 153
- "Manipulating layers" on page 157
- "Converting layers to tables" on page 159
- "Using rulers and grids to lay out pages" on page 160
- "Using a tracing image" on page 161

About layers in Dreamweaver

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.

Note: A layer in Dreamweaver is a `div` tag with absolute or relative position. Layers, as described in this chapter, refer to the Dreamweaver layout concept, not the `layer` tag.

Understanding layers

With Dreamweaver, you can use layers to lay out your page. 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.

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 "Converting layers to tables" on page 159. If your audience is likely to be using more recent browsers, however, you can design layouts entirely with layers, without converting them to tables.

Related topics

- "Managing layers" on page 153

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. To change the default tag, see "Setting layer preferences" on page 152.

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.

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 "Setting layer preferences and properties" on page 152.

Related topics

- "Understanding layers" on page 146

Inserting content blocks for layout

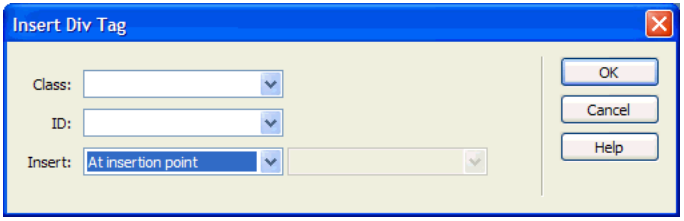
You can use `div` tags to position content blocks in your document. This is especially useful if you have an existing CSS style sheet with positioning styles attached to your document. Dreamweaver enables you to quickly insert a `div` tag and apply existing styles to it.

Note: You can use a Dreamweaver design file as a starting point for your CSS layout. Select a file from the Page Designs (CSS) category of the New Document dialog box (see "Creating a document based on a Dreamweaver design file" on page 70).

To insert a div tag:

1. In the Document window, place the insertion point where you want the `div` tag to appear.
2. Do one of the following:
   - Select Insert > Layout Objects > Div Tag.
   - In the Layout category of the Insert bar, click the Div Tag button.

   The Insert Div Tag dialog box appears.



3 Complete the dialog box.

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

4 Click OK.

The `div` tag appears as a box in your document with placeholder text. When you move the pointer over the box, Dreamweaver highlights it.

If the `div` tag is absolutely or relatively positioned, it acts like a Dreamweaver layer. For information about working with layers, see "Setting layer preferences and properties" on page 152, "Managing layers" on page 153, or "Manipulating layers" on page 157.

For information about working with `div` tags that aren't absolutely or relatively positioned, see "Working with content blocks for layout" on page 149.

Related topics

- "Working with content blocks for layout" on page 149

Changing highlight color for content blocks

When you move the pointer over a content block, Dreamweaver highlights the block. Dreamweaver highlights blocks that have a class, id, or style associated with them. You can turn highlighting on or off as necessary, or change the highlight color in preferences.

To change highlighting preferences for content blocks:

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

The Preferences dialog box appears.

2 Select Highlighting from the category list on the left.

3 Make either of the following changes:

- To change the highlighting color for content blocks, click the Mouse-over color box, then select a highlight color using the color picker (or enter the hexadecimal value for the highlight color in the text box).

  For information about using the color picker, see "Working with colors" on page 260.

- To turn highlighting on or off for content blocks, select or deselect the Show checkbox for Mouse-Over.

Note: These options affect all objects, such as tables, that Dreamweaver highlights when you move the pointer over them.

4 Click OK.

Working with content blocks for layout

After you insert a `div` tag (see "Inserting content blocks for layout" on page 147), you can manipulate the content block or you can add content to it.

Note: Content block `div` tags that are absolutely or relatively positioned act like Dreamweaver layers. For information about working with these `div` tags, skip this section and see "Setting layer preferences and properties" on page 152, "Managing layers" on page 153, or "Manipulating layers" on page 157.

Content block `div` tags that are *not* absolutely or relatively positioned do not have visible borders, but when you move the pointer over a content block, Dreamweaver highlights the block (as long as it has a `class`, `id`, or `style` associated with it). To change the highlight color or turn highlighting off, see "Changing highlight color for content blocks" on page 148.

When you select a content block that is not absolutely or relatively positioned, you can view and edit rules for it in the Tag Inspector. You can add content to the block by simply placing your insertion point in the block, then adding content just as you would add content to a page.

To view and edit rules applied to a content block:

1 Do one of the following to select the `div` tag:

- Click the border of the block.

  Tip: Look for the highlighting to see the border.

- Click inside the block's borders, then select the `div` tag from the tag selector at the bottom of the Document window.

2 Select Window > Tag Inspector to open the Tag Inspector if it is not already open; then click the CSS tab if it is not already open.

  Rules applied to the `div` tag appear in the panel.

3 Make edits as necessary.

To place the insertion point in a block to add content:

- Click anywhere inside the block's borders.

To change the placeholder text in a content block:

- Select the text, then type over it or press the Delete key.

Note: You can add content to the block just as you would add content to a page.

Related topics

- "Inserting content blocks for layout" on page 147

Inserting a layer

Dreamweaver lets you create layers on your page easily and position them precisely. You can also create nested layers. For more information, see "Nesting layers" on page 151.

Note: You can use a Dreamweaver design file as a starting point for your CSS layout. Select a file from the Page Designs (CSS) category of the New Document dialog box (see "Creating a document based on a Dreamweaver design file" on page 70).

When you insert a layer, a layer-code marker appears in Design view. Dreamweaver shows the layer's border, by default, and highlights the block (as long as it has a `class`, `id`, or `style` associated with it) when you move the pointer over it. You can turn off the layer-code marker by disabling invisible elements and hide layer borders. To change the highlight color or to turn highlighting off, see "Changing highlight color for content blocks" on page 148.

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.

After you create a layer, you can add content to it by simply placing your insertion point in the layer, then adding content just as you would add content to a page.

To draw a single layer or multiple layers consecutively:

1. In the Layout category of the Insert bar, click the Draw Layer button.
2. In the Document window's Design view, do one of the following:
   - Drag to draw a single layer.
   - Control-drag (Windows) or Command-drag (Macintosh) to draw multiple layers consecutively.

     You can continue to draw new layers as long as you do not release the Control or Command key.

To insert a layer at a particular place in the document:

- Place the insertion point in the Document window, then select Insert > 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 154.

To show or hide layer-code markers:

- Select View > Visual Aids > Invisible Elements.

To show or hide layer borders:

- Select View > Visual Aids > Layer Borders.

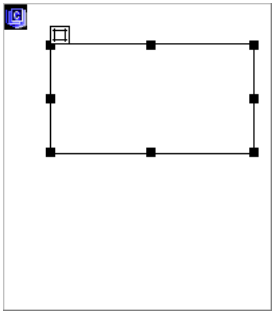
Related topics

- "Setting layer preferences and properties" on page 152
- "Managing layers" on page 153
- "Manipulating layers" on page 157

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.

You can turn the Nesting option on to automatically nest when you draw a layer starting inside another layer.



To draw a nested layer:

1. In the Layout category of the Insert bar, click the Draw Layer button.
2. In the Document window's Design view, 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 nest 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 insert a nested layer:

- Place the insertion point inside an existing layer in the Document window's Design view, then select Insert > Layer.

To nest an existing layer inside another layer using the Layers panel:

1. Select Window > 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.
3. Release the mouse button when the name of the target layer is highlighted.

To nest layers automatically when you draw a layer starting inside another layer:

- Select the Nesting option in the Layer preferences.

  For more information, see "Setting layer preferences" on page 152.

Related topics

- "Inserting a layer" on page 150
- "Managing layers" on page 153
- "Manipulating layers" on page 157

Setting layer preferences and properties

You can specify default settings for your layers. You can also view and set various attributes of one or more layers using the Property inspector.

Setting layer 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 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

The Preferences dialog box appears.

2 Select Layers from the Category list on the left.

3 Make changes as necessary.

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

4 Click OK.

Related topics

- "Viewing and setting properties for multiple layers" on page 153

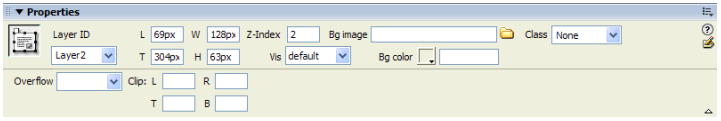
Viewing and setting properties for a single layer

When you select a layer, the Property inspector displays layer properties.

To view and set layer properties:

1 Select a layer (see "Selecting layers" on page 154).

2 In the Property inspector (Window > Properties), click the expander arrow in the lower right corner, if it isn't already expanded, to see all properties.



3 Change the layer's attributes by setting properties as necessary.

For more information, click the Help button in the Property inspector.

Related topics

- "Setting layer preferences" on page 152
- "Managing layers" on page 153

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.

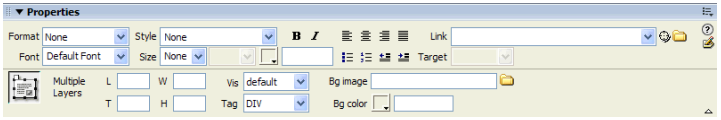
To select multiple layers:

- Hold down Shift while selecting layers (see "Selecting layers" on page 154).

To view and set properties for multiple layers:

1 Select multiple layers.

2 In the Property inspector (Window > Properties), click the expander arrow in the lower right corner, if it isn't already expanded, to see all properties.



3 Change the layers' attributes by setting properties.

For more information, click the Help button in the Property inspector.

Related topics

- "Setting layer preferences" on page 152
- "Viewing and setting properties for a single layer" on page 152

Managing layers

You can select layers to work with them. You can change the stacking order and visibility of layers. The Layers panel provides a convenient way to manage layers.

Using the Layers panel

The Layers panel is a way to manage the layers in your document. 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.

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.

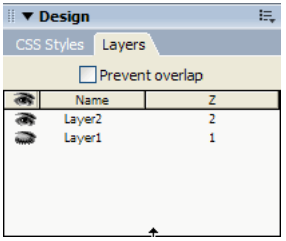
To open the Layers panel:

- Select Window > Layers.

To show or hide nested layers:

- Click the Plus (+) or Minus (–) buttons (Windows) or the expander arrow (Macintosh) next to the layer name.

Nested layers appear as names connected to parent layers.



Related topics

- "Inserting a layer" on page 150
- "Changing the stacking order of layers" on page 155
- "Changing layer visibility" on page 156

Selecting layers

You can select one or more layers to manipulate them or change their properties.

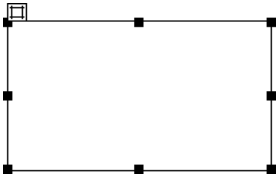
To select a layer in the Layers panel:

- In the Layers panel (Window > Layers), click the name of the layer.

To select a layer in the Document window, do one of the following:

- 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, select View > Visual Aids > Invisible Elements.

To select multiple layers, do one of the following:

- In the Layers panel (Window > Layers), Shift-click two or more layer names.
- In the Document window, 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.

Related topics

- "Inserting a layer" on page 150
- "Setting layer preferences and properties" on page 152
- "Managing layers" on page 153
- "Manipulating layers" on page 157

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 Property inspector.

To change the stacking order of layers in the Layers panel:

1. Select Window > Layers to open the Layers panel.
2. Do one of the following:
   - 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; then type a higher 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. Select Window > 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 (Window > Properties), 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.

Related topics

- "Using the Layers panel" on page 153
- "Selecting layers" on page 154

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.

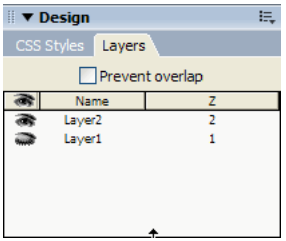
To change layer visibility:

1 Select Window > 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:

- In the Layers panel (Window > Layers), 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 topics

- "Using the Layers panel" on page 153
- "Changing the stacking order of layers" on page 155

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 159).

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 159).

To resize a layer:

1 In the Design view, select a layer (see "Selecting layers" on page 154).

2 Do one of the following to resize the layer:

- 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.

  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 "Using rulers and grids to lay out pages" on page 160.
- In the Property inspector (Window > Properties), 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 "Setting layer preferences and properties" on page 152.

To resize multiple layers at once:

1 In the Design view, select two or more layers (see "Selecting layers" on page 154).

2 Do one of the following:

- Select 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 (Window > Properties), under Multiple Layers, enter width and height values.

The values are applied to all selected layers.

Related topics

- "Aligning layers" on page 158
- "Using rulers and grids to lay out pages" on page 160

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 159.)

To move one or more selected layers:

1 In the Design view, select a layer or multiple layers (see "Selecting layers" on page 154).

2 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 the Shift key while pressing an arrow key to move the layer by the current grid snapping increment. For information about setting the grid snapping increment, see "Using rulers and grids to lay out pages" on page 160.

Related topics

- "Resizing layers" on page 157
- "Using rulers and grids to lay out pages" on page 160

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 In the Design view, select the layer (see "Selecting layers" on page 154).

2 Select 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).

Related topics

- "Resizing layers" on page 157
- "Moving layers" on page 158
- "Using rulers and grids to lay out pages" on page 160

Converting layers to tables

Instead of using tables or Layout mode to create your layout, some web designers prefer to work with layers. Dreamweaver enables you to create your layout using layers, then (if you want) convert them into tables. For example, you might need to convert your layers to tables if you need to support browsers before version 4.0.

Note: 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.

You can convert back and forth between layers and tables to adjust the layout and optimize page design. You cannot convert a specific table or layer on a page; you must convert layers to tables and tables to layers for an entire page.

Note: Converting from layers to tables may result in tables with a large number of empty cells.

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, use the Prevent Overlap option to constrain layer movement and positioning so that layers don't overlap.

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.

To prevent layers from overlapping, do one of the following:

- In the Layers panel (Window > Layers), select the Prevent Overlaps option.
- In the Document window, select Modify > Arrange > Prevent Layer Overlaps.

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.

Before you convert to tables, make sure layers do not overlap (see "Preventing layer overlaps" on page 159).

To convert layers to a table:

1 Select Modify > Convert > Layers to Table.

The Convert Layers to Tables dialog box appears.

2 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 Select Modify > Convert > Tables to Layers.

The Convert Tables to Layers dialog box appears.

2 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.

Using rulers and grids to lay out pages

Use rulers and grids as visual guides for drawing, positioning, or resizing elements in the Document window's Design view.

Rulers can appear on the left and top borders of the page, marked in pixels, inches, or centimeters.

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 change ruler settings, do one of the following:

- To toggle rulers on and off, select View > Rulers > Show.
- To change the origin, drag the ruler-origin icon at the upper left corner of the Design view of the Document window anywhere on the page.

  To reset the origin to its default position, select View > Rulers > Reset Origin.
- To change the unit of measure, select Pixels, Inches, or Centimeters from the View > Rulers submenu.

To show or hide the grid:

- Select View > Grid > Show Grid.

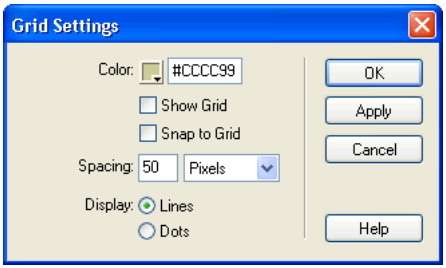
To enable or disable snapping:

- Select View > Grid > Snap to Grid.

To change grid settings:

1 Select 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.

Related topics

- "Resizing layers" on page 157
- "Moving layers" on page 158
- "Aligning layers" on page 158

Using a tracing image

You can use a tracing image as a guide to re-create a page design that was created in a graphics application such as Macromedia Freehand or Fireworks.

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.

To place a tracing image in the Document window:

1 Do one of the following:

- Select View > Tracing Image > Load.
- Select Modify > Page Properties, then click the Browse button next to the Tracing Image text box.

2 In the dialog box that appears, select an image file, then click Select (Windows) or Choose (Macintosh).

The Page Properties dialog box appears.

3 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, select Modify > Page Properties.

To show or hide the tracing image:

- Select View > Tracing Image > Show.

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 change the position of a tracing image:

1. Select View > Tracing Image > Adjust Position.
2. 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 5 pixels at a time, press Shift and an arrow key.

To reset the position of the tracing image:

- Select 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. Select 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.

CHAPTER 8
Presenting Content with Tables

Tables are a powerful tool for presenting tabular data and for laying out text and graphics on a page.

Many designers use tables to lay out web pages. Macromedia Dreamweaver MX 2004 provides two ways to view and manipulate tables: Standard mode, in which tables are presented as a grid of rows and columns, and Layout mode, which allows you to draw, resize, and move boxes on the page while still using tables for the underlying structure (see Chapter 9, "Laying Out Pages in Layout Mode," on page 183).

Note: You can also lay out your pages using CSS positioning (see Chapter 7, "Laying Out Pages with CSS," on page 145).

This chapter contains the following sections:

- "About tables" on page 164
- "Inserting a table and adding content" on page 165
- "Importing and exporting tabular data" on page 166
- "Selecting table elements" on page 167
- "Using Expanded Tables mode for easier table editing" on page 170
- "Formatting tables and cells" on page 170
- "Resizing tables, columns, and rows" on page 173
- "Adding and removing rows and columns" on page 176
- "Splitting and merging cells" on page 178
- "Copying, pasting, and deleting cells" on page 179
- "Nesting tables" on page 181
- "Sorting tables" on page 181

About 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 enables you to manipulate columns as well as rows and cells.

Dreamweaver displays the table width and the column width for each table column when the table is selected or when the insertion point is in the table. Next to the widths are arrows for the table header menu and the column header menus. Use the menus for quick access to some common table-related commands. You can turn the widths and menus on or off as necessary (see "Displaying table and column widths and menus" on page 176).

If you do not see a width for the table or for a column, then that table or column does not have a specified width in the HTML code. If two numbers appear, then the visual width as it appears in Design view doesn't match the width specified in the HTML code. This can happen when you resize a table by dragging its lower right corner or when you add content to a cell that's larger than its set width.

For example, if you set a column's width to 200 pixels, then add content that stretches the width to 250 pixels, two numbers appear for that column: 200 (the width specified in the code) and (250) in parentheses (the visual width of the column as it's rendered on your screen).

To reconcile the column widths, see "Making column widths in code consistent with visual widths" on page 175.

Related topics

- "Inserting a table and adding content" on page 165

Table formatting precedence in HTML

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, 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.

Related topics

- "Formatting tables and cells" on page 170

About splitting and merging table 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 into a single cell that spans two rows.

Related topics

- "Splitting and merging cells" on page 178

Inserting a table and adding content

Use the Insert bar or the Insert menu to create a new table. Then, add text and images to table cells the same way that you add text and images outside of a table (see Chapter 13, "Inserting and Formatting Text," on page 273 and Chapter 14, "Inserting Images," on page 303).

To insert a table:

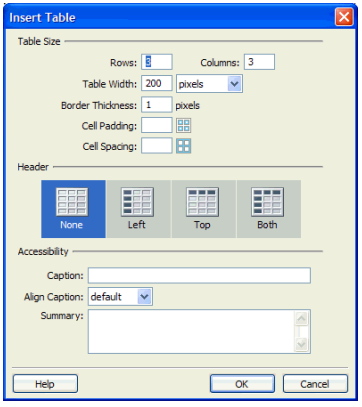
1 In the Design view of the Document window, place the insertion point where you want the table to appear.

Note: If your document is blank, then the only place you can place the insertion point is at the beginning of the document.

2 Do one of the following:

- Select Insert > Table.
- In the Common category of the Insert bar, click the Table button.

The Insert Table 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.

Related topics

- "Viewing and setting table, cell, row, and column properties" on page 171
- "Editing accessibility attributes for a table" on page 171

Importing and exporting 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.

You can also 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.

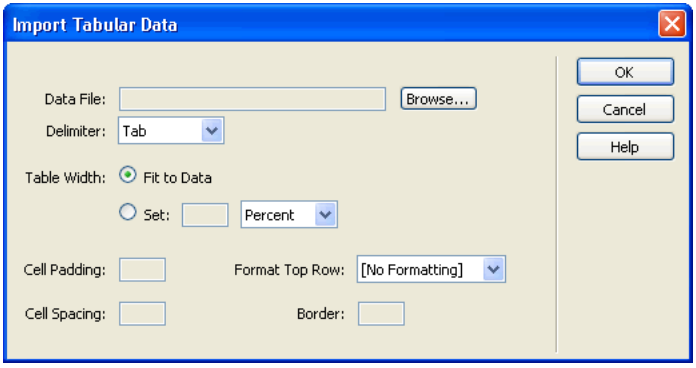
Tip: If you want only some of the data from a table—for example, the first six rows or the first six columns—then copy the cells containing that data, paste those cells outside of the table (to create a new table), and export the new table.

To import table data:

1 Do one of the following:

- Select File > Import > Tabular Data.
- Select 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.

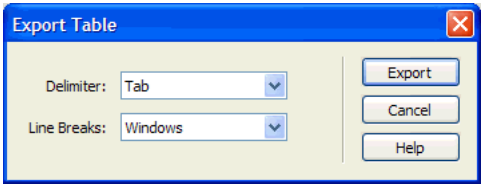
3 Click OK.

To export a table:

1 Place the insertion point in any cell of the table.

2 Select 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.

The Export Table As dialog box appears.

5 Enter a name for the file.

6 Click Save.

Selecting table elements

You can select an entire table, row, or column at once. You can also select one or more individual cells.

When you move your pointer over a table, row, column, or cell, Dreamweaver highlights all the cells in that selection so that you know exactly which cells will be selected if you click the selection. This is useful when you have tables without borders, cells that span multiple columns or rows, or nested tables. You can change the highlight color in preferences.

Tip: If you position the pointer over a table's border, then hold the Control key (Windows) or Command key (Macintosh), the entire structure of the table—that is, all cells in the table—is highlighted. This is useful when you have nested tables and want to see the structure of one of the tables.

To change the highlight color for selecting table elements:

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

The Preferences dialog box appears.

2 Select Highlighting from the category list on the left.

3 Make either of the following changes:

- To change the highlighting color for table elements, click the Mouse-over color box, then select a highlight color using the color picker (or enter the hexadecimal value for the highlight color in the text box).

  For information about using the color picker, see "Working with colors" on page 260.

- To turn highlighting on or off for table elements, select or deselect the Show checkbox for Mouse-over.

Note: These options affect all objects, such as layers and Layout mode view tables and cells, that Dreamweaver highlights when you move the pointer over them.

4 Click OK.

Related topics

- "Formatting tables and cells" on page 170
- "Resizing tables, columns, and rows" on page 173
- "Copying, pasting, and deleting cells" on page 179

Selecting a table

There are several ways you can select an entire table at once.

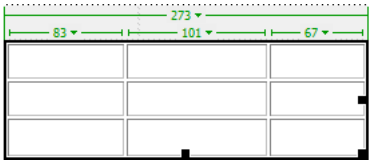
To select an entire table, do one of the following:

- Click the upper left corner of the table, anywhere on the top or bottom edge of the table, or on a row or column's border.

Note: The pointer changes to the table grid icon when you can select the table (unless you click a row or column border).

- Click in a table cell, then select the `<table>` tag in the tag selector at the lower left corner of the Document window.
- Click in a table cell, then select Modify > Table > Select Table.
- Click in a table cell, click the table header menu, then select Select Table.

  Selection handles appear on the selected table's lower and right edges.



Related topics

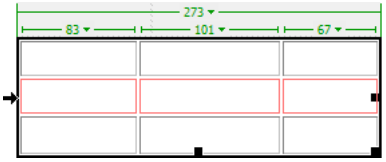
- "Selecting cells" on page 169

Selecting rows or columns

You can select an individual row or column or multiple rows or columns.

To select individual or multiple rows or columns:

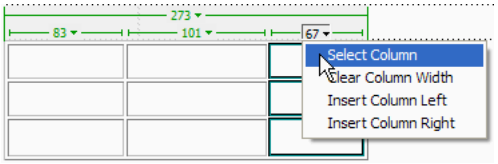
1. Position the pointer to point to the left edge of a row or the top edge of a column.
2. When the pointer changes to a selection arrow, click to select a row or column, or drag to select multiple rows or columns.



To select a single column:

1 Click in the column.

2 Click the column header menu, then select Select Column.



Related topics

- "Selecting a table" on page 168

Selecting cells

You can select a single cell, a line or block of cells, or nonadjacent cells. To select entire rows or columns, see "Selecting rows or columns" on page 168.

To select a single cell, do one of the following:

- Click in the cell, then select the `<td>` tag in the tag selector at the lower left corner of the Document window.
- Control-click (Windows) or Command-click (Macintosh) in the cell.
- Click in the cell, then select Edit > Select All.

  Tip: Select Edit › Select All again when a cell is selected to select the entire table.

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, Control-click (Windows) or Command-click (Macintosh) in the same cell to select it, 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.

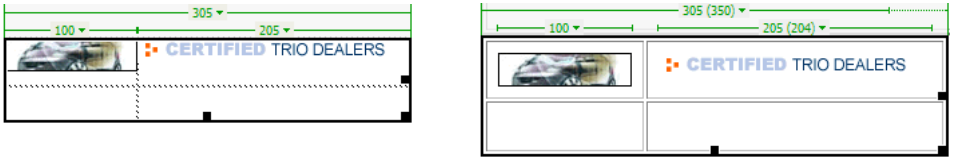
Related topics

- "Selecting a table" on page 168

Using Expanded Tables mode for easier table editing

Expanded Tables mode temporarily adds cell padding and spacing to all tables in a document and increases the tables' borders to make editing easier. This mode enables you to select items in tables or precisely place the insertion point.

For example, you might expand a table to place the insertion point to the left or right of an image, without inadvertently selecting the image or table cell.



Note: Once you make your selection or place the insertion point, you should return to the Standard mode of Design view to make your edits. Some visual operations, such as resizing, will not give expected results in Expanded Tables mode.

To switch into Expanded Tables mode:

1 If you are working in Code view, select View > Design or View > Code and Design.

You cannot switch to Expanded Tables mode in Code view.

2 Do one of the following:

- Select View > Table Mode > Expanded Tables Mode.
- In the Layout category of the Insert bar, click the Expanded Tables Mode button.

A bar labeled Expanded Tables Mode appears across the top of the Document window. Dreamweaver adds cell padding and spacing to all tables on the page and increases the tables' borders.

To switch out of Expanded Tables mode, do one of the following:

- Click Exit in the bar labeled Expanded Tables Mode at the top of the Document window.
- Select View > Table Mode > Standard Mode.
- In the Layout category of the Insert bar, click the Standard Mode button.

Dreamweaver returns to Standard mode.

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. Before you set table and cell properties, it's a good idea to understand which properties have precedence over the others (see "Table formatting precedence in HTML" on page 164).

Tip: To format the text inside a table cell, use the same procedures you would use to format text outside of a table. For more information, see Chapter 13, "Inserting and Formatting Text," on page 273.

Related topics

- "Resizing tables, columns, and rows" on page 173

Viewing and setting table, cell, row, and column properties

When a table or cell is selected, the Property inspector lets you view and change its properties.

Note: Before you change properties for table elements, it's a good idea to understand which properties have precedence over the others (see "Table formatting precedence in HTML" on page 164).

To view and set table or table element properties:

1. Select a table, cell, row, or column (see "Selecting table elements" on page 167).
2. In the Property inspector (Window > Properties), click the expander arrow in the lower right corner to see all the properties.
3. Change properties as necessary.

   For more information, click the Help button in the Property inspector.

   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.

Editing accessibility attributes for a table

If you did not add accessibility attributes for your table when you inserted it (see "Inserting a table and adding content" on page 165) and need to add them later, or if you need to edit accessibility attributes, you can do so in Code view or Design view.

To add or edit accessibility values for a table in Code view:

- Edit the appropriate attributes in the code.

  Tip: To quickly locate the tags in the code, click in the table, then select the `<table>` tag in the tag selector at the bottom of the Document window.

To add or edit accessibility values for a table in Design view, do any of the following:

- To edit the table caption, highlight the caption, then type a new caption.
- To edit the caption alignment, place the insertion point in the table caption, right-click (Windows) or Control-click (Macintosh), then select Edit Tag Code.
- To edit the table summary, select the table, right-click (Windows) or Control-click (Macintosh), then select Edit Tag Code.

Related topics

- "Viewing and setting table, cell, row, and column properties" on page 171

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.

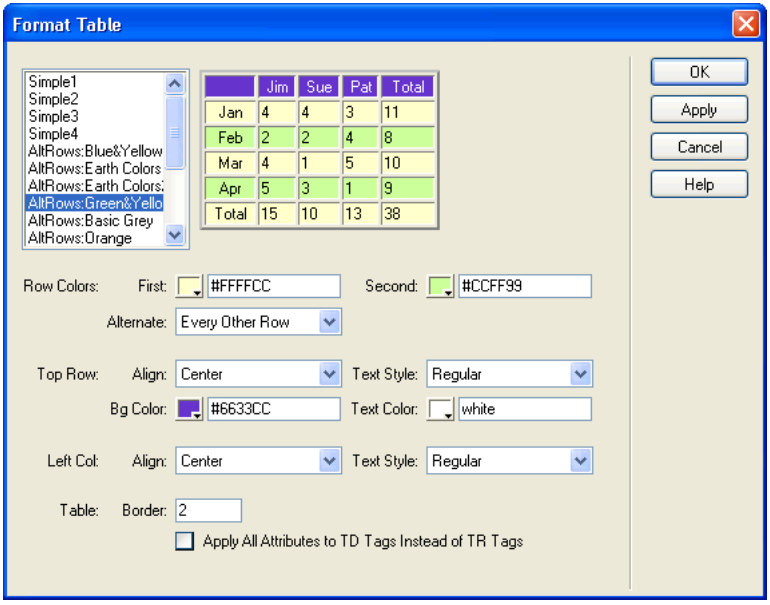
Note: Only simple tables can be formatted using preset designs. You can't use these designs to format tables that contain merged cells (`colspan` or `rowspan`) or column groups, or any other table that doesn't consist of a simple rectangular grid of cells.

To use a preset table design:

1 Select a table (see "Selecting a table" on page 168).

2 Select Commands > Format Table.

The Format Table dialog box appears.



3 Customize options as desired.

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

4 Click Apply or OK to format your table with the selected design.

Resizing tables, columns, and rows

You can resize an entire table or individual rows and columns. If you have trouble resizing, you can clear the column widths or row heights and start over.

Sometimes the column widths set in the HTML code do not match their apparent widths on the screen. When this happens, you can make the widths consistent. Table and column widths and header menus appear in Dreamweaver to help you lay out your tables; you can turn the widths and menus on and off as necessary.

Resizing a table

You can resize a table by dragging one of its selection handles. Dreamweaver displays the table width, along with a table header menu, at the top or bottom of the table when the table is selected or the insertion point is in the table.

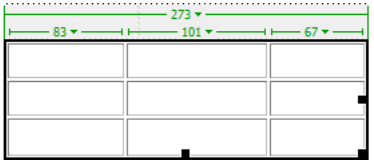
Note: You can turn off the table width and table header menu if necessary (see "Displaying table and column widths and menus" on page 176).

When you resize an entire table, all of the cells in the table change size proportionately. 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 clear set widths or heights, see "Clearing set widths and heights" on page 175.

To resize a table:

1 Select the table (see "Selecting a table" on page 168).

Selection handles appear on 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 the table in both dimensions, drag the selection handle at the lower right corner.

Resizing columns and rows

You can change the width of a column or the height of a row in the Property inspector or by dragging the borders of the column or row. If you have trouble resizing, you can clear the column widths or row heights and start over (see "Clearing set widths and heights" on page 175).

Note: You can also change cell widths and heights directly in the HTML code using Code view. For more information, see "Coding in Dreamweaver" on page 415.

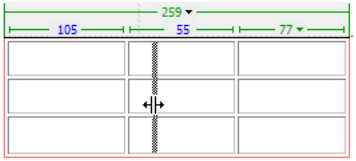
Dreamweaver displays column widths, along with column header menus, at the tops or bottoms of columns when the table is selected or the insertion point is in the table. For more information, see "About tables" on page 164.

Note: You can turn off the column widths and column header menus if necessary (see "Displaying table and column widths and menus" on page 176).

To change a column's width and keep the overall table width:

- Drag the right border of the column you want to change.

  The width of the adjacent column also changes, so you actually resize two columns. Visual feedback shows you how the columns will adjust; the overall table width does not change.

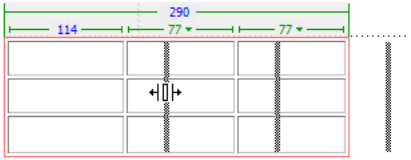


Note: In tables with percentage-based widths (not pixels), if you drag the right border of the rightmost column, the whole table's width changes, and all of the columns grow wider or narrow proportionately.

To change a column's width and maintain the size of the other columns:

- Hold the Shift key, then drag the column's border.

  The width of one column changes. Visual feedback shows you how the columns will adjust; the overall table width changes to accommodate the column you are resizing.



To change a row's height visually:

- 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 (see "Selecting rows or columns" on page 168).
2. In the Property inspector (Window > Properties), enter a value in the W text field for the column's width or in the H text field for the row's height.

   For more information, click the Help button in the Property inspector.

   Tip: You can specify widths and heights as either pixels or percentages, and you can convert from pixels to percentages and back.

3. Press Tab or Enter (Windows) or Return (Macintosh) to apply the value.

Related topics

- "Resizing a table" on page 173
- "Clearing set widths and heights" on page 175
- "Displaying table and column widths and menus" on page 176

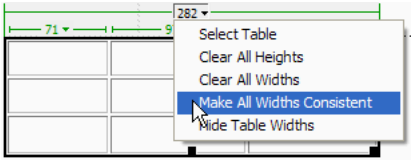
Making column widths in code consistent with visual widths

If you see two numbers for a column's width, then the column width set in the HTML code does not match the column's apparent width on the screen. You can make the width specified in the code match the visual width. For more information, see "About tables" on page 164.

To make widths consistent:

1 Click in a cell.

2 Click the table header menu, then select Make All Widths Consistent.



Dreamweaver resets the width specified in the code to match the visual width.

Related topics

- "Resizing a table" on page 173
- "Resizing columns and rows" on page 173

Clearing set widths and heights

You might want to clear set widths and heights before you resize a table, or if you have trouble resizing a table or individual columns or rows and want to start over.

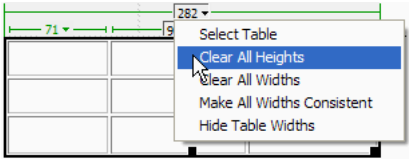
Note: When you resize a table by dragging one of its selection handles, you change the visual size of the cells in the table, but you do not change any specified widths or heights of cells. It's a good idea to clear set widths and heights before resizing.

To clear all set widths or heights in a table:

1 Select the table (see "Selecting a table" on page 168).

2 Do one of the following:

- Select Modify > Table > Clear Cell Widths or Modify > Table > Clear Cell Heights.
- In the Property inspector (Window > Properties), click the Clear Row Heights button or the Clear Column Widths button.
- Click the table header menu, then select Clear All Heights or Clear All Widths.



To clear a column's set width:

1 Click in the column.

2 Click the column header menu, then select Clear Column Width.

Related topics

- “Resizing a table” on page 173
- “Resizing columns and rows” on page 173
- “Making column widths in code consistent with visual widths” on page 175

Displaying table and column widths and menus

Dreamweaver displays table and column widths, along with arrows to access the table header menu and the column header menus, when a table is selected or when the insertion point is in the table. (For more information, see “About tables” on page 164.) You can turn on and off the widths and menus as necessary.

To turn on or off table and column widths and menus, do one of the following:

- Select View > Visual Aids > Table Widths.
- Right-click (Windows) or Control-click (Macintosh) in the table, then select Table > Table Widths.

Related topics

- “Resizing a table” on page 173
- “Resizing columns and rows” on page 173

Adding and removing rows and columns

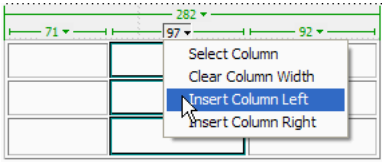
To add and remove rows and columns, use the Modify > Table or column header menu.

Tip: Pressing Tab in the last cell of a table automatically adds another row to the table.

To add a single row or column:

1. Click in a cell.
2. Do one of the following:
   - Select Modify > Table > Insert Row or Modify > Table > Insert Column.

     A row appears above the insertion point or a column appears to the left of the insertion point.
   - Click the column header menu, then select Insert Column Left or Insert Column Right.

     A column appears to the left or right of the insertion point.

To add multiple rows or columns:

1 Click in a cell.

2 Select Modify > Table > Insert Rows or Columns.

The Insert Rows or Columns dialog box appears.

3 Select Rows or Columns, then complete the dialog box.

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

4 Click OK.

The rows or columns appear in the table.

To delete a row or column, do one of the following:

- Click in a cell within the row or column you want to delete, then select Modify > Table > Delete Row or Modify > Table > Delete Column.
- Select a complete row or column (see "Selecting rows or columns" on page 168), then select Edit > Clear or press the Delete key.

The entire row or column disappears from the table.

To add or delete rows or columns using the Property inspector:

1 Select the table (see "Selecting a table" on page 168).

2 In the Property inspector (Windows > Properties), 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

- "Inserting a repeating table" on page 236
- "Displaying multiple recordset results" on page 566

Splitting and merging cells

Use the Property inspector or the commands in the Modify > Table submenu to split or merge cells. For more information, see "About splitting and merging table cells" on page 165.

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 in a contiguous line 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 Do one of the following:

- Select Modify > Table > Merge Cells.
- In the expanded Property inspector (Window > Properties), click the Merge Cells button.

  Note: If you don't see the button, click the arrow in the lower right corner of the Property inspector so that you see all the options.

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 Do one of the following:

- Select Modify > Table > Split Cell.
- In the expanded Property inspector (Window > Properties), click the Split Cell button.

  Note: If you don't see the button, click the arrow in the lower right corner of the Property inspector so that you see all the options.

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 Do one of the following:

- Select Modify > Table > Increase Row Span or Modify > Table > Increase Column Span.
- Select Modify > Table > Decrease Row Span or Modify > Table > Decrease Column Span.

Copying, pasting, and deleting cells

You can copy, paste, or delete 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 cell or multiple cells in a contiguous line 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 Select Edit > Cut or Edit > Copy.

Note: If you selected an entire row or column and you select Edit › Cut, the entire row or column is removed from the table (not just the contents of the cells).

To paste table cells:

1 Select 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.

  Note: If you have less than a full row or column of cells in the clipboard, and you click in a cell and 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.
- To create a new table with the pasted cells, place the insertion point outside of the table.

2 Select 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.

Note: Make sure the selection does not consist entirely of complete rows or columns.

2 Select Edit > Clear or press Delete.

Note: If only complete rows or columns are selected when you select Edit › Clear or press Delete, the entire rows or columns—not just their contents—are removed from the table.

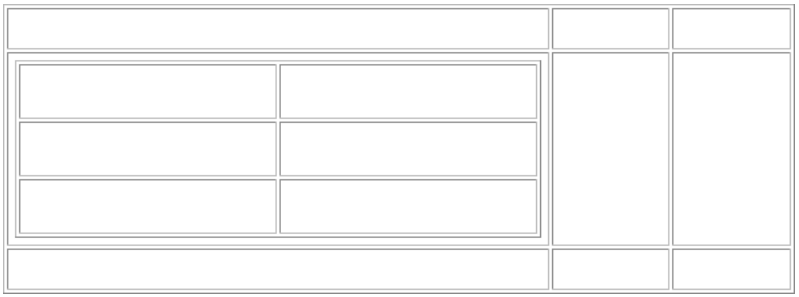
To delete rows or columns that contain merged cells:

1 Select the row or column.

2 Select Modify > Table > Delete Row or Modify > Table > Delete Column.

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.



To nest a table within a table cell:

1 Click in a cell of the existing table.

2 Select Insert > Table.

The Insert Table 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 existing table.

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 178.

To sort a table:

1 Select the table (see "Selecting a table" on page 168) or click in any cell.

2 Select Commands > Sort Table.

The Sort Table dialog box appears.

3 Complete the dialog box.

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

4 Click OK.

CHAPTER 9
Laying Out Pages in Layout Mode

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, Macromedia Dreamweaver MX 2004 provides Layout mode.

In Layout mode, 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.

Note: For more information about working with tables in Standard mode, see Chapter 8, "Presenting Content with Tables," on page 163. Or, as an alternative to using tables in Standard or Layout mode, you can use CSS positioning to lay out pages (see "Inserting content blocks for layout" on page 147).

This chapter contains these sections:

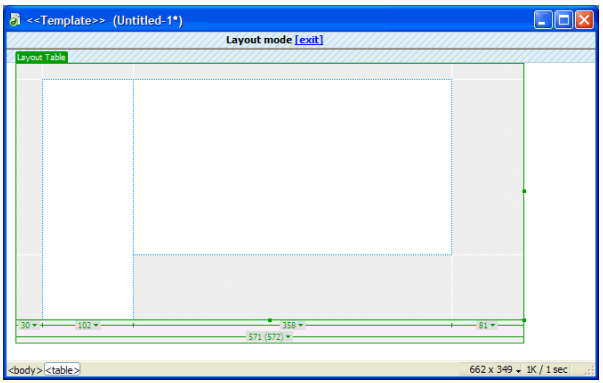
- "About Layout mode" on page 184
- "Switching from Standard to Layout mode" on page 187
- "Drawing in Layout mode" on page 187
- "Adding content to a layout cell" on page 190
- "Clearing automatically set cell heights" on page 192
- "Resizing and moving layout cells and tables" on page 192
- "Formatting layout cells and tables" on page 194
- "Setting column width" on page 195
- "Setting preferences for Layout mode" on page 197

About Layout mode

In Layout mode, you use layout cells and tables to lay out your page before adding 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. As you add content, you can move cells around as necessary to adjust your layout.

Tip: For maximum flexibility, you can draw each cell only when you're ready to put content into it. This enables you to leave more blank space in the layout table for a longer time, so that you can move or resize cells more easily.

Layout tables appear outlined in green on your page; layout cells appear outlined in blue on your page. (To change the default outline colors, see "Setting preferences for Layout mode" on page 197.) When you move the pointer over a table cell, Dreamweaver highlights the cell. (To turn highlighting on or off or to change the highlight color, see "Drawing layout cells and tables" on page 187.)



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, separate 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 (see "Drawing a nested layout table" on page 189). This structure enables 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.

Related topics

- "Switching from Standard to Layout mode" on page 187
- "Drawing in Layout mode" on page 187

About viewing table and cell widths in Layout mode

The widths of layout tables and cells (in pixels, or as a percentage of the page width) appear at the top or bottom of the table when the table is selected or when the insertion point is in the table. Beside the widths are arrows for the table header menu and the column header menus. Use the menus for quick access to some common commands.

Note: To turn off column widths, along with table tabs and header menus, you must turn off all visual aids (View › Visual Aids › Hide All).

Sometimes, the width might not appear for a column; you might see any of the following:

- No width. If you do not see a width for the table or for a column, then that table or column does not have a specified width in the HTML code. To specify a fixed width, see "Making a column autostretch or fixed-width" on page 195.
- Two numbers. If two numbers appear, then the visual width as it appears in Design view doesn't match the width specified in the HTML code. This can happen when you resize a table by dragging its lower right corner or when you add content to a cell that's larger than its set width.

  For example, if you set a column's width to 200 pixels, then add content that stretches the width to 250 pixels, two numbers appear at the top of that column: 200 (the width specified in the code) and (250) in parentheses (the visual width of the column as it's rendered on your screen).

  To reconcile the column widths, see "Making column widths in code consistent with visual widths" on page 197.
- Wavy line. A wavy line appears for columns that are set to autostretch. For information about setting a column to autostretch, see "Making a column autostretch or fixed-width" on page 195.
- Double bars. Columns that contain spacer images have double bars around the column width. For information about spacer images, see "Using spacer images" on page 196.

Related topics

- "Setting column width" on page 195

Layout table and cells grid lines

When you draw a layout cell in a layout table, a light grid of lines appears, extending from the edges of any 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.

You can also use the Dreamweaver grid, which is set and doesn't change based on the placement of cells, to help you lay out your page (see "Using rulers and grids to lay out pages" on page 160).

Related topics

- "About Layout mode" on page 184

Fixed column width and autostretch columns

A table column in Layout mode can have either a fixed width or a width that automatically expands to fill as much of the browser window as possible (*autostretch*).

Fixed-width columns have a specific numeric width, such as 300 pixels. Dreamweaver displays the column width for each fixed-width column at the top or bottom of the column.

Autostretch columns change automatically depending on the browser window's width. If your layout includes an autostretch column, the layout always fills the entire width of the visitor's browser window. You can make only one column in a given layout table autostretch. An autostretch column displays a wavy line in the column width area.

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.

Related topics

- "Making a column autostretch or fixed-width" on page 195

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 where the column width appears.

You can manually insert and remove spacer images in specific columns or remove all spacer images in the page.

Related topics

- "Using spacer images" on page 196

Switching from Standard to Layout mode

Before you can draw layout tables or layout cells, you must switch from Standard mode into Layout mode. When you create tables for layout in Layout mode, but it's a good idea to switch back to Standard mode before you add content to or edit your tables.

Note: If you create a table in Standard mode, then switch into Layout mode, the resulting layout table might contain empty layout cells. You might need to delete these empty layout cells before you can create new layout cells or move layout cells around.

To switch into Layout mode:

1 If you are working in Code view, select View > Design or View > Code and Design.

You cannot switch to Layout mode in Code view.

2 Do one of the following:

- Select View > Table Mode > Layout Mode.
- In the Layout category of the Insert bar, click the Layout Mode button.

A bar labeled Layout Mode appears across the top of the Document window. If there are tables on your page, they appear as layout tables.

To switch out of Layout mode, do one of the following:

- Click Exit in the bar labeled Layout Mode at the top of the Document window.
- Select View > Table Mode > Standard Mode.
- In the Layout category of the Insert bar, click the Standard Mode button.

Dreamweaver returns to Standard mode.

Drawing in Layout mode

Layout mode allows you to draw cells and tables, including tables nested in other tables. The snapping grid helps you to align cells (see "Using rulers and grids to lay out pages" on page 160).

Drawing layout cells and tables

You can draw layout cells and tables on your page in Layout mode. 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: In Layout mode, you can't use the Insert Table and Draw Layer tools that you can use in Standard mode. To use those tools, you must first switch to Standard mode.

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, then dragging the table's resize handles.

When you move the pointer over a layout cell, Dreamweaver highlights it. You can turn highlighting on or off or change the highlight color in preferences.

To draw a layout cell:

1 Make sure you are in Layout mode (see "Switching from Standard to Layout mode" on page 187).

2 In the Layout category of the Insert bar, click Draw Layout Cell button.

The pointer changes to a cross hair (+).

3 Position the pointer where you want to start the cell on the page, then drag to create the layout cell.

Tip: To draw more than one layout cell without having to repeatedly select Draw Layout Cell, Control-drag (Windows) or Command-drag (Macintosh) when you draw the layout cell. As long as you continue to hold Control or Command, you can draw layout cells one after the other.

If you draw the cell close to the edge of the layout table, cell edges automatically snap to the edges of the containing layout table. To temporarily disable snapping, hold down Alt (Windows) or Option (Macintosh) while drawing the cell.

The cell appears outlined in blue on your page. To change the outline color, see "Setting preferences for Layout mode" on page 197.

To draw a layout table:

1 Make sure you are in Layout mode (see "Switching from Standard to Layout mode" on page 187).

2 In the Layout category of the Insert bar, click the Draw Layout Table button.

The pointer changes to a cross hair (+).

3 Position the pointer on the page, then drag to create the layout table.

Tip: To draw more than one layout table without having to repeatedly select Draw Layout Table, Control-drag (Windows) or Command-drag (Macintosh) when you draw the layout table. As long as you continue to hold Control or Command, you can draw layout tables one after the other.

You can create a layout table in an empty area of your page layout, around existing layout cells and tables, or nested inside an existing layout table. If your page contains content, and you want to add a layout table in an empty area of your page layout, you can draw a new layout table only below the bottom of the existing content.

Tip: If you try to draw a layout table below existing content and 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.

Note: 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 189.

The Layout table appears outlined in green on your page. To change the outline color, see "Setting preferences for Layout mode" on page 197.

To change highlighting preferences for layout cells:

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

The Preferences dialog box appears.

2 Select Highlighting from the category list on the left.

3 Make either of the following changes:

- To change the highlighting color, click the Mouse-over color box, then select a highlight color using the color picker (or enter the hexadecimal value for the highlight color in the text box).

  For information about using the color picker, see "Working with colors" on page 260.

- To turn highlighting on or off, select or deselect the Show checkbox for Mouse-Over.

Note: These options affect all objects, such as tables and layers, that Dreamweaver highlights when you move the pointer over them.

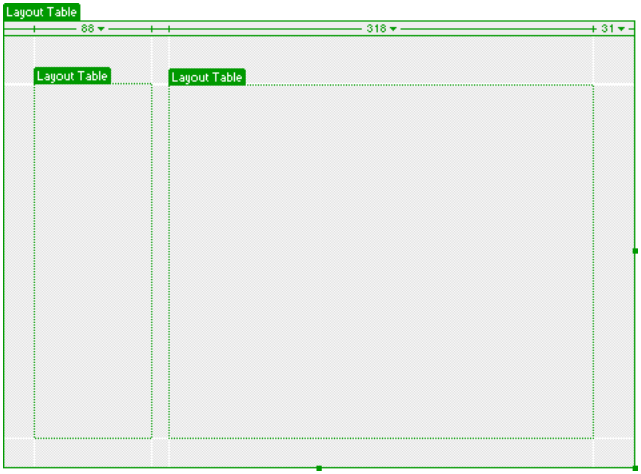
4 Click OK.

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 mode (see "Switching from Standard to Layout mode" on page 187).

2 In the Layout category of the Insert bar, click the Draw Layout Table button.

The pointer changes to a cross hair (+).

3 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 mode (see "Switching from Standard to Layout mode" on page 187).

2 In the Layout category of the Insert bar, click the Draw Layout Table button.

The pointer changes to a cross hair (+).

3 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.

Adding content to a layout cell

You can add text, images, and other content to layout cells in Layout mode just as you would add content to table cells in Standard mode. 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 in Layout mode" on page 187).

To add text to a layout cell:

1 Place the insertion point in the layout cell where you want to add text.

2 Do one of the following:

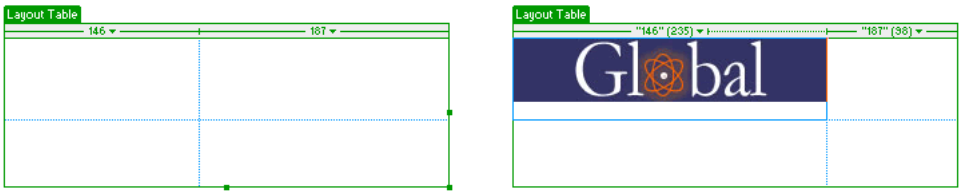
- Type text into the cell.

  The cell automatically expands as you type, if necessary.

- Paste text copied from another document.

  For more information, see "Inserting text" on page 280.

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 width 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 195.



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:

- Select Insert > Image.
- In the Common category of the Insert bar, click the arrow on the Images button, then select Image.

  Tip: If the Image button is showing (as in the following example) in the Insert bar, you can click the button instead of using the pop-up menu.

The Select Image Source dialog box appears.

3 Select an image file.

For more information, see "Inserting an image" on page 305.

4 Click OK.

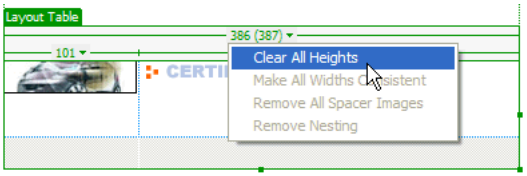
The image appears in the layout cell.

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 might 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:

- Click the table header menu, then select Clear All Heights.



- 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 (Window > Properties).

Dreamweaver clears all the heights specified in the table. Some of the table cells might shrink vertically.

Resizing and moving 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.)

Note: To use the Dreamweaver grid as a guide while you move or resize your cells and tables, see "Using rulers and grids to lay out pages" on page 160.

Resizing and moving layout cells

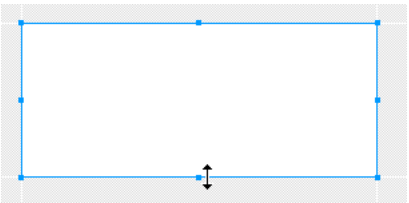
You can resize or move layout cells, but they cannot overlap. You cannot move or resize a cell to make it cross the boundaries of the layout table that contains it. A layout cell cannot be made smaller than its contents.

For information on how to resize or move layout tables, see "Resizing and moving layout tables" on page 193.

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.

Tip: Hold down Shift while pressing an arrow key to move the cell 10 pixels at a time.

Resizing and moving layout tables

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 or move layout cells, see "Resizing and moving layout cells" on page 192.

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.

Note: You can move a layout table only if it's nested inside another layout table.

2 Do one of the following:

- Drag the table to another location on the page.
- Press the arrow keys to move the table 1 pixel at a time.

Tip: Hold down Shift while pressing an arrow key to move the table 10 pixels at a time.

Formatting layout cells and tables

You can change the appearance of any layout cell or table by using the Property inspector.

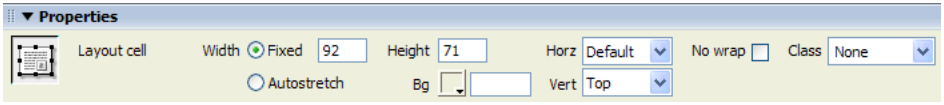
Formatting layout cells

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 (Window > Properties) if it isn't already open.



3 Change the cell's formatting by setting properties.

For more information, click the Help button in the Property inspector.

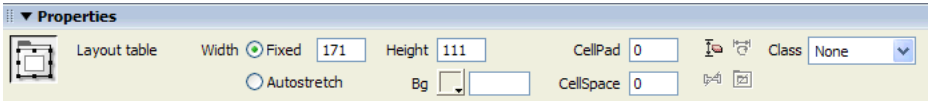
Formatting layout tables

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.

2 Open the Property inspector (Window > Properties) if it isn't already open.



3 Change the table's formatting by setting properties.

For more information, click the Help button in the Property inspector.

Setting column width

You can set a column to a specific width or make it stretch to fill as much of a browser window as possible. You can also specify a minimum width for a column using a spacer image.

Sometimes the column widths width set in the HTML code do not match their apparent widths on the screen. When this happens, you can make the widths consistent.

Making a column autostretch or fixed-width

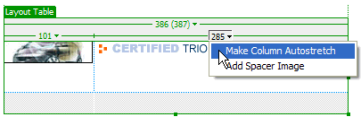
A column in a table can be either fixed-width or autostretch. For more information, see "Fixed column width and autostretch columns" on page 186.

Making a column autostretch before your layout is complete might 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.)

To make a column autostretch:

1 Do one of the following:

- Click the column header menu, then select Make Column Autostretch.



- Select a cell in the column by clicking an edge of the cell, then click Autostretch in the Property inspector.

Note: You can make only one column in a given table autostretch.

If you have not set a spacer image for this site, the Choose Spacer Image dialog box appears.

2 If the Choose Spacer Image dialog box appears, select an option, then click OK.

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

A wavy line appears at the top or bottom of the autostretch column. Double bars appear at the tops or bottoms of columns that contain spacer images.

To set a column to a fixed width, do one of the following:

- Click the column header menu, then select Make Column Fixed Width.

  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, then click Fixed and type a numeric value in the Property inspector.

  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.

The width of the column appears at the top or bottom of the column.

Related topics

- "Making column widths in code consistent with visual widths" on page 197

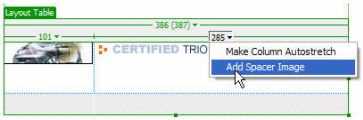
Using spacer images

To require a column to be a certain minimum width, you can insert a spacer image into that column. For more information, see "Spacer images" on page 186. You can remove spacer images from a single column or from the entire table.

The first time you insert a spacer image, you set up a spacer image for the site. You can set preferences for spacer images (see "Setting preferences for Layout mode" on page 197).

To insert a spacer image into a column:

1 Click the column header menu, then select Add Spacer Image.



If you have not set a spacer image for this site, the Choose Spacer Image dialog box appears.

2 If the Choose Spacer Image dialog box appears, select an option, then click OK.

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

Dreamweaver inserts the spacer image into the column. The image is not visible, but the column might shift slightly and a double bar appears at the top or bottom of the column to indicate that it contains a spacer image.

To remove a spacer image from a single column:

- Click the column header menu, then select Remove Spacer Image.

Dreamweaver removes the spacer image. The column might shift.

To remove all spacer images from a table, do one of the following:

- Click the table header menu, then select Remove All Spacer Images.
- Select the table, then click the Remove All Spacers button in the Property inspector (Window > Properties).

The layout of your whole table might shift. If there is no content in some columns, the columns might disappear completely from the Design view.

Making column widths in code consistent with visual widths

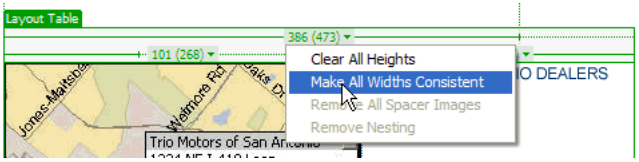
If you see two numbers for a column's width, then the column width set in the HTML code does not match the column's apparent width on the screen. You can make the width specified in the code match the visual width. For more information, see "About viewing table and cell widths in Layout mode" on page 185.

To make widths consistent:

1 Click in a cell.

2 Do one of the following:

- Click the table header menu, then select Make All Widths Consistent.



- Select the table, then click the Remove All Spacers button in the Property inspector (Window > Properties).

Dreamweaver resets the width specified in the code to match the visual width.

Related topics

- "Making a column autostretch or fixed-width" on page 195
- "Using spacer images" on page 196

Setting preferences for Layout mode

You can specify preferences for spacer image files and for the colors that Dreamweaver uses to draw layout tables and layout cells.

To set Layout mode preferences:

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

The Preferences dialog box appears.

2 Select Layout Mode from the category list on the left.

3 Make changes as necessary.

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

4 Click OK.

CHAPTER 10
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.

Note: 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 200
- "Working with framesets in the Document window" on page 203
- "Creating frames and framesets" on page 204
- "Selecting frames and framesets" on page 207
- "Opening a document in a frame" on page 209
- "Saving frame and frameset files" on page 209
- "Viewing and setting frame properties and attributes" on page 210
- "Viewing and setting frameset properties" on page 212
- "Controlling frame content with links" on page 213
- "Handling browsers that can't display frames" on page 214
- "Using JavaScript behaviors with frames" on page 214

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 that initially appears in each frame. The frameset file itself doesn't contain HTML content that displays in a browser, except in the `noframes` section (see "Handling browsers that can't display frames" on page 214); the frameset file simply provides information to the browser about how a set of frames should look and what documents should appear in them.

Related topics

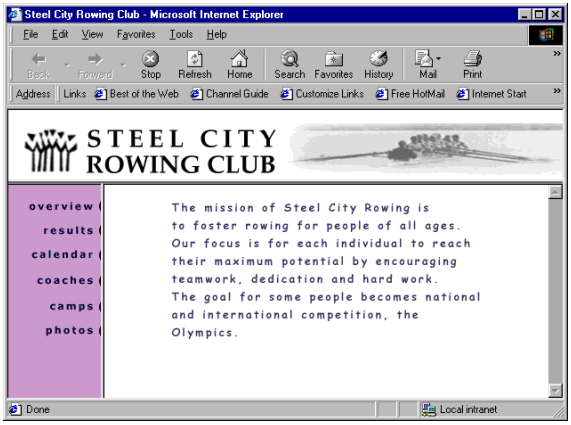
- "Working with framesets in the Document window" on page 203
- "Creating frames and framesets" on page 204

Understanding how frames and framesets work

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.

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.

The following example shows a frame layout consisting 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.

Note that a frame is not a file. It's easy to think of the document that currently appears in a frame as an integral part of the frame, but the document isn't actually part of the frame. The frame is a container that holds the document—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 appear 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.

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 in 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.

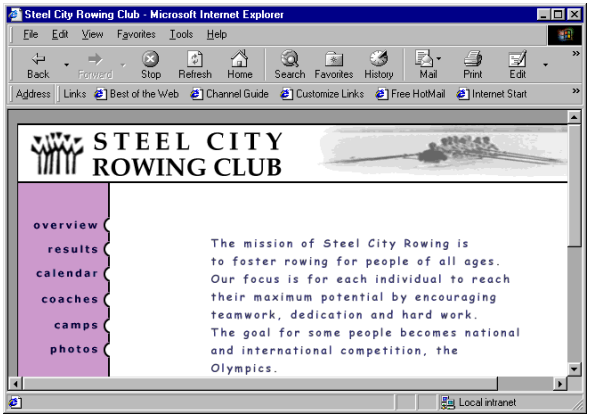
Related topics

- "Understanding nested framesets" on page 202

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 216.) 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 (see "Handling browsers that can't display frames" on page 214). 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 scroll bar (if the content is too large to fit in a window), so a visitor can scroll the frames independently.

  For example, a visitor who scrolls 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 don't appear in browsers, so it can be difficult for a visitor to bookmark a specific page (unless you provide server code that enables them to load a framed version of a particular page).

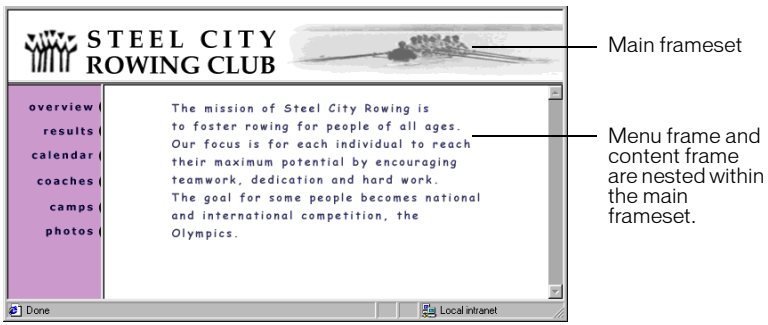
Related topics

- "Understanding how frames and framesets work" on page 200
- "Handling browsers that can't display frames" on page 214

Understanding 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 "Designing a frameset" on page 205.

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.

Related topics

- "Understanding how frames and framesets work" on page 200
- "Deciding whether to use frames" on page 201

Working with framesets in the Document window

Dreamweaver enables you to view and edit all of the documents associated with a set of frames in one Document window. This approach enables you to 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 204).
2. Save every file that's going to appear in a frame (see "Saving frame and frameset files" on page 209).

   Remember that each frame displays a separate HTML document, and you must save each document, along with the frameset file.
3. Set the properties for each frame and for the frameset (see "Viewing and setting frame properties and attributes" on page 210 and "Viewing and setting frameset properties" on page 212).

   This includes naming each frame, setting scrolling and non-scrolling options, and more.
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 213).

Creating frames and framesets

There are two ways to create a frameset in Dreamweaver: you can select from several predefined framesets or you can design it yourself.

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.

Using a predefined frameset

Predefined framesets make it easy for you to select the type of frameset you want to create. If you prefer to design your own frameset, see "Designing a frameset" on page 205.

There are two ways to create a predefined frameset:

- The Insert bar enables 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.

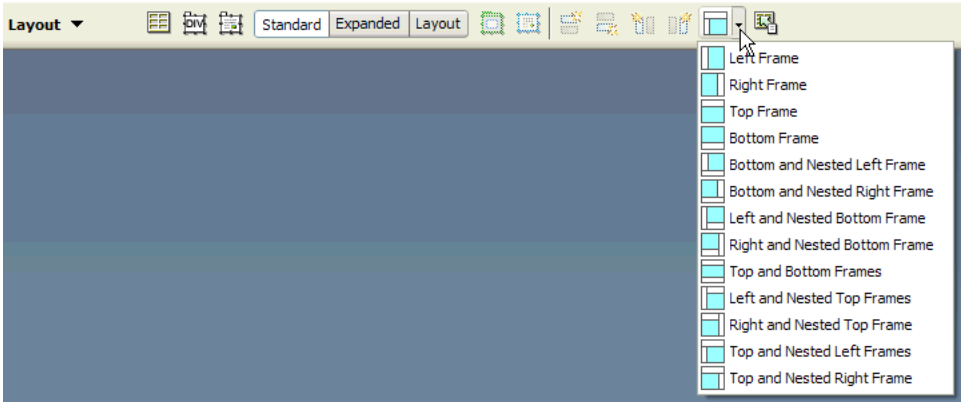
To create a predefined frameset and display an existing document in a frame:

1 Place the insertion point in a document.

2 Do one of the following:

- Select a predefined frameset from the Insert > HTML > Frames submenu.
- In the Layout category of the Insert bar, click the drop down arrow on the Frames button, then select a predefined frameset.

The frameset icons provide a visual representation of each frameset as applied to the current document. The blue area of a frameset icon represents the current document, and the white areas represent frames that will display other documents.



Note: When you apply a frameset, 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 Frame Tag Accessibility Attributes dialog box appears, if you have set up Dreamweaver to prompt you for frame accessibility attributes (see "Optimizing the workspace for accessible page design" on page 53).

3 If the Frame Tag Accessibility Attributes dialog box appears, complete the dialog box for each frame, then click OK.

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

Note: If you click Cancel, the frameset appears in the document, but Dreamweaver does not associate accessibility tags or attributes with it.

To edit frame tag accessibility attributes, see "Viewing and setting frame properties and attributes" on page 210.

To create a new empty predefined frameset:

1 Select File > New.

2 In the New Document dialog box, select the Framesets category.

3 Select a frameset from the Framesets list.

4 Click Create.

The frameset appears in your document and the Frame Tag Accessibility Attributes dialog box appears, if you have activated the dialog box in Preferences (see "Optimizing the workspace for accessible page design" on page 53).

5 If the Frame Tag Accessibility Attributes dialog box appears, complete the dialog box for each frame, then click OK.

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

Note: If you press Cancel, the frameset appears in the document, but Dreamweaver does not associate accessibility tags or attributes with it.

To edit frame tag accessibility attributes, see "Viewing and setting frame properties and attributes" on page 210.

Designing a frameset

You can design your own frameset in Dreamweaver by adding "splitters" to the window. If you prefer to use a predefined frameset, see "Using a predefined frameset" on page 204.

Tip: Before creating a frameset or working with frames, make the frame borders visible in the Document window's Design view by selecting View ›Visual Aids › Frame Borders.

To create a frameset:

- Select Modify > Frameset, then select a splitting item (such as Split Frame Left or Split Frame Right) from the submenu.

  Dreamweaver splits the window into frames. If you had an existing document open, it 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, select 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, 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.

To resize a frame, do one of the following:

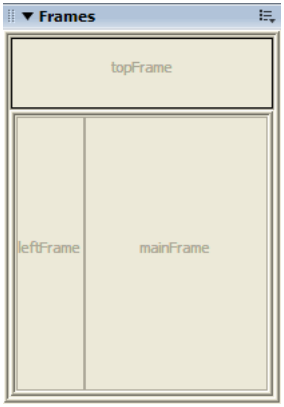
- To set approximate sizes for frames, drag frame borders in the Document window's Design view.
- To specify exact sizes, and to specify how much space the browser allocates to a row or column of frames when the browser window size doesn't allow the frames to display at full size, use the Property inspector (see "Viewing and setting frameset properties" on page 212).

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.

Selecting frames and framesets in the Frames panel

The Frames panel provides a visual representation of the frames within a frameset. It 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:

- Select Window > Frames.

To select a frame in the Frames panel:

- Click the frame in the Frames panel.

  A selection outline appears around the frame in both the Frames panel and the Document window's Design view.

To select a frameset in the Frames panel:

- Click the border that surrounds the frameset in the Frames panel.

  A selection outline appears around the frameset in both the Frames panel and the Document window's Design view.

Related topics

- "Viewing and setting frame properties and attributes" on page 210
- "Viewing and setting frameset properties" on page 212

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.

To select a frame in the Document window:

- Alt-click (Windows) or Option-Shift-click (Macintosh) inside a frame in Design view.

  A selection outline appears around the frame.

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; select View >Visual Aids > Frame Borders to make frame borders visible if they aren't.)

  A selection outline appears around the frameset.

  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 207.

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).

Related topics

- "Selecting frames and framesets in the Frames panel" on page 207
- "Viewing and setting frame properties and attributes" on page 210
- "Viewing and setting frameset properties" on page 212

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 Select File > Open in Frame.

3 Select a document to open in the frame, then click OK (Windows) or Choose (Macintosh).

The document appears in the frame.

4 (Optional) To make this 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 will display 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.

Note: 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.

To save a frameset file:

1 Select the frameset in the Frames panel or the Document window (see "Selecting frames and framesets" on page 207).

2 Do one of the following:

- To save the frameset file, select File > Save Frameset.
- To save the frameset file as a new file, select 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 select File > Save Frame or File > Save Frame As.

To save all files associated with a set of frames:

- Select 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 enables you to select a filename. For each frame that hasn't yet been saved, a heavy border appears around the frame and a dialog box enables you to select 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 and attributes

Use the Property inspector to view and set most frame properties.

You may also want to set some frame attributes, such as the title attribute (which is not the same as the name attribute), to improve accessibility. You can enable the accessibility authoring option for frames to set attributes when you create frames (see "Optimizing the workspace for accessible page design" on page 53), or you can set attributes after you insert a frame. To edit accessibility attributes for a frame, you can use the Tag inspector to edit the HTML code directly.

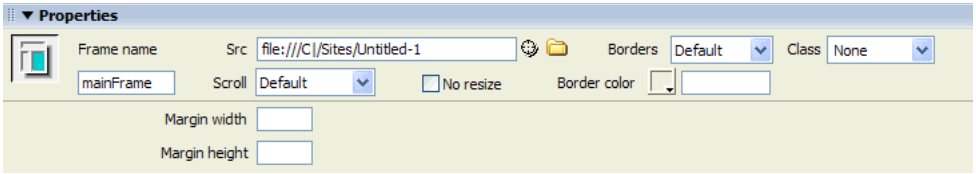
To set properties for a frameset, see "Viewing and setting frameset properties" on page 212.

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 (Window > Frames).

2 In the Property inspector (Window > Properties), click the expander arrow in the lower right corner to see all of the frame properties.



3 Make changes as necessary.

For more information, click the Help button in the Property inspector.

Tip: To change the background color of a frame, set the background color of the document in the frame in page properties.

To set accessibility values for a frame:

1 In the Frames panel (Window > Frames), select a frame by placing the insertion point in one of the frames.

2 Select Modify > Edit Tag.

The Tag editor appears.

3 Select Style Sheet/Accessibility from the category list on the left.

4 Enter values as desired.

5 Click OK.

To edit accessibility values for a frame:

1 Display Code view or Code and Design views for your document, if you're currently in Design view.

2 In the Frames panel (Window > Frames), select a frame by placing the insertion point in one of the frames.

Dreamweaver highlights the frame tag in the code.

3 Right-click (Windows) or Control-click (Macintosh) in the code, then select Edit Tag.

The tag editor appears.

4 Make changes as necessary.

5 Click OK.

To change the background color of a document in a frame:

1 Place the insertion point in the frame.

2 Select Modify > Page Properties.

The Page Properties dialog box appears.

3 Click the Background color pop-up menu, then select a color.

4 Click OK.

Related topics

- "Creating frames and framesets" on page 204

Viewing and setting frameset properties

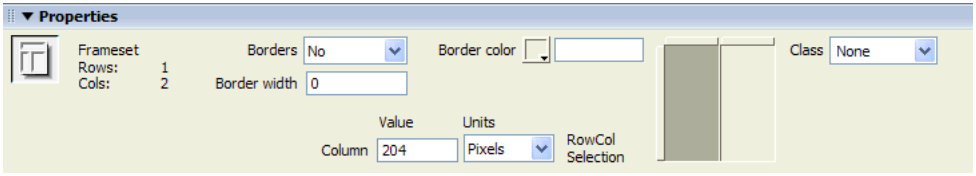
Use the Property inspector to view and set most frameset properties. To set properties for a frame, see "Viewing and setting frame properties and attributes" on page 210.

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 (Window > Frames).

2 In the Property inspector (Window > Properties), click the expander arrow in the lower right corner to see all of the frameset properties.



3 Make changes as necessary.

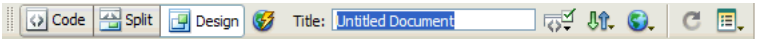
For more information, click the Help button in the Property inspector.

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 (Window > Frames).

2 In the Title field of the Document toolbar, type a name for the frameset document.



When a visitor views the frameset in a browser, the title appears in the browser's title bar.

Related topics

- "Creating frames and framesets" on page 204

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 target a frame:

1. In Design view, select text or an object.
2. In the Link field in the Property inspector (Window > Properties), do one of the following:
   - Click the folder icon, then select the file to link to.
   - Drag the Point to File icon to the Files panel to select the file to link to.
3. In the Target pop-up menu in the Property inspector, select the frame or window in which the linked document should appear:
   - `_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.

   Frame names also 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, 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 in the Target text box.

   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 Select 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.
- Select Window > Code Inspector, place the insertion point between the `body` tags that appear inside the `noframes` tags, then type the HTML code for the content.

3 Select 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 388.)

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 383.)

Insert Navigation Bar 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 333.)

Insert Jump Menu sets 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 opens. (See "Inserting jump menus" on page 331.)

CHAPTER 11
Managing Templates

A Macromedia Dreamweaver MX 2004 template is a special type of document that you use to design a "fixed" page layout; you can then create documents based on the template that inherit the page layout of the template. As you design a template, you specify which areas of documents based on that template users can edit.

Note: Templates enable you to control a large design area and reuse complete layouts. If you want to reuse individual design elements, such as a site's copyright information or a logo, you can create library items. For more information, see Chapter 5, "Managing Site Assets and Libraries," on page 115.

The chapter contains the following sections:

- "About Dreamweaver templates" on page 216
- "Creating a Dreamweaver template" on page 226
- "Creating templates for a Contribute site" on page 229
- "Creating editable regions" on page 232
- "Creating repeating regions" on page 234
- "Using optional regions" on page 237
- "Defining editable tag attributes" on page 239
- "Creating a nested template" on page 241
- "Editing and updating templates" on page 242
- "Managing templates" on page 245
- "Exporting and importing template XML content" on page 246
- "Exporting a site without template markup" on page 247
- "Applying or removing a template from an existing document" on page 247
- "Editing content in a template-based document" on page 248

About Dreamweaver templates

A template author designs a "fixed" page layout in a template. The author then creates regions in the template that are editable in documents based on that template; if the author does not define a region as editable, then template users cannot edit content in that area. Templates enable template authors to control which page elements template users—such as writers, graphic artists, or other web developers—can edit. There are several types of template regions the template author can include in a document.

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 documents based on it.

Related topics

- "Creating a Dreamweaver template" on page 226

Types of template regions

Dreamweaver automatically locks most regions of a document when you save the document as a template. As a template author you specify 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:

An editable region is an unlocked region in a template-based document—a section a template user can edit. A template author can specify 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 232.

A repeating region is a section of the layout in a document that is set to repeat. For example, you can set a table row to repeat. Usually repeating sections are editable so that the template user can edit the content in the repeating element, while the design itself is under the control of the template author. The template user uses repeat region control options to add or delete copies of the repeated region in a document based on the template as necessary.

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 234. For information about creating a repeating table, see "Inserting a repeating table" on page 236. 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 250.

An optional region is a section of a template that you specify as optional, to hold content such as optional text or an image that may or may not appear in a document based on the template. In the template-based page, the template user usually controls whether the content is displayed. For information about setting optional regions in a template, see "Inserting an optional region" on page 237. For information about editing optional regions in a template-based page, see "Modifying template properties" on page 249.

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 template user set the alignment to left, right, or center. For information about setting editable tag attributes, see "Specifying editable tag attributes in a template" on page 240. For information about editing the tags in a template-based page, see "Modifying template properties" on page 249.

Related topics

- "Creating editable regions" on page 232
- "Creating repeating regions" on page 234
- "Using optional regions" on page 237
- "Defining editable tag attributes" on page 239

Templates and template-based documents in Design and Code views

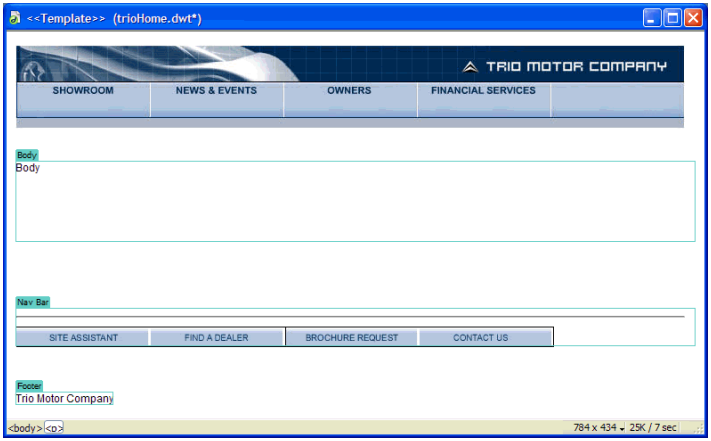
You can view templates and template-based documents in Design or Code view.

Viewing templates in Design view

In templates, editable regions appear in the Document window's Design view surrounded by rectangular outlines in a preset highlight color. A small tab appears at the upper left corner of each region, showing the name of the region.

Note: For information about setting highlighting color preferences, see "Setting highlighting preferences for template regions" on page 229.

You can identify a template file by looking at the title bar in the Document window. The title bar for a template file contains the word <<*Template*>> in the title bar, and the filename extension for the file is .dwt.



Related topics

- "Viewing template-based documents in Design view" on page 218
- "Viewing template-based documents in Code view" on page 219

Viewing templates in Code view

In Code view, you can make changes to both editable and locked HTML source code in a template.

Tip: You can use code color preferences to set your own color scheme so you can easily distinguish template regions when you view a document in Code view (see "Customizing code coloring preferences for a template" on page 228).

Editable content regions are marked in HTML with the following comments:

`<!-- TemplateBeginEditable>` and `<!-- TemplateEndEditable -->`

Everything between these comments will be editable in documents based on the template. The HTML source code for an editable region 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>
```

Note: Be careful when you edit template code in Code view not to change any of the template-related comment tags that Dreamweaver relies on.

Related topics

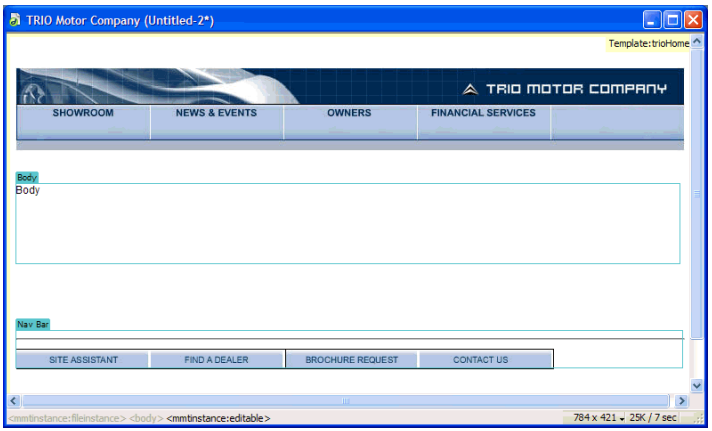
- "Viewing templates in Design view" on page 217
- "Viewing template-based documents in Code view" on page 219

Viewing template-based documents in Design view

In a document based on a template (a template-based document), editable regions appear in the Document window's Design view surrounded by rectangular outlines in a preset highlight color. A small tab appears at the upper left corner of each region, showing the name of the region.

Note: For information about setting highlighting color preferences, see "Setting highlighting preferences for template regions" on page 229.

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.



Related topics

- "Viewing templates in Design view" on page 217
- "Viewing templates in Code view" on page 218

Viewing template-based documents in Code view

In Code view, editable regions of a document derived from a template appear 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; Dreamweaver prevents you from typing in locked regions.

Editable content is marked in HTML with the following Dreamweaver comments:

`<!-- InstanceBeginEditable>` and `<!-- InstanceEndEditable -->`

Everything between these comments is editable in a template-based document. The HTML source code for an editable region might look like this:

```
<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>
```

The default color for non-editable text is gray; you can select a different color for the editable and non-editable regions in the Preferences dialog box. For more information, see "Customizing code coloring preferences for a template" on page 228.

Related topics

- "Viewing templates in Design view" on page 217
- "Viewing templates in Code view" on page 218
- "Viewing template-based documents in Design view" on page 218

Template tag syntax

This section describes the general syntax rules and lists the HTML comment tags Dreamweaver uses to specify regions in templates and template-based documents. Dreamweaver automatically inserts template tags in the code when you insert a template object.

Note: Dreamweaver uses HTML comment tags to define template regions, so template-based documents are still valid HTML files.

General syntax rules

Following are general syntax rules:

- Wherever a space appears, you can substitute any amount of whitespace (spaces, tabs, line breaks). The whitespace is mandatory except at the very beginning or end of a comment.
- Attributes can be given in any order. For example, in a TemplateParam, you can specify the type before the name.
- Comment and attribute names are case sensitive.
- All attributes must be in quotation marks. Single or double quotes can be used.

For information about checking your syntax, see "Checking template syntax" on page 244.

Related topics

- "Instance tags" on page 221

Template tags

Dreamweaver uses the following template tags:

```
<!-- TemplateBeginEditable name="..." -->
<!-- TemplateEndEditable -->

<!-- TemplateParam name="..." type="..." value="..." -->

<!-- TemplateBeginRepeat name="..." -->
<!-- TemplateEndRepeat -->

<!-- TemplateBeginIf cond="..." -->
<!-- TemplateEndIf -->

<!-- TemplateBeginPassthroughIf cond="..." -->
<!-- TemplateEndPassthroughIf -->

<!-- TemplateBeginMultipleIf -->
<!-- TemplateEndMultipleIf -->
```

```
<!-- TemplateBeginPassthroughMultipleIf -->
<!-- TemplateEndPassthroughMultipleIf -->

<!-- TemplateBeginIfClause cond="..." -->
<!-- TemplateEndIfClause -->

<!-- TemplateBeginPassthroughIfClause cond="..." -->
<!-- TemplateEndPassthroughIfClause -->

<!-- TemplateExpr expr="..." --> (equivalent to @@...@@)

<!-- TemplatePassthroughExpr expr="..." -->

<!-- TemplateInfo codeOutsideHTMLIsLocked="..." -->
```

Related topics

- "General syntax rules" on page 220

Instance tags

Dreamweaver uses the following instance tags:

```
<!-- InstanceBegin template="..." codeOutsideHTMLIsLocked="..." -->
<!-- InstanceEnd -->

<!-- InstanceBeginEditable name="..." -->
<!-- InstanceEndEditable -->

<!-- InstanceParam name="..." type="..." value="..." passthrough="..." -->

<!-- InstanceBeginRepeat name="..." -->
<!-- InstanceEndRepeat -->

<!-- InstanceBeginRepeatEntry -->
<!-- InstanceEndRepeatEntry -->
```

Related topics

- "General syntax rules" on page 220
- "Template tags" on page 220

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 320.

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.

Nested templates

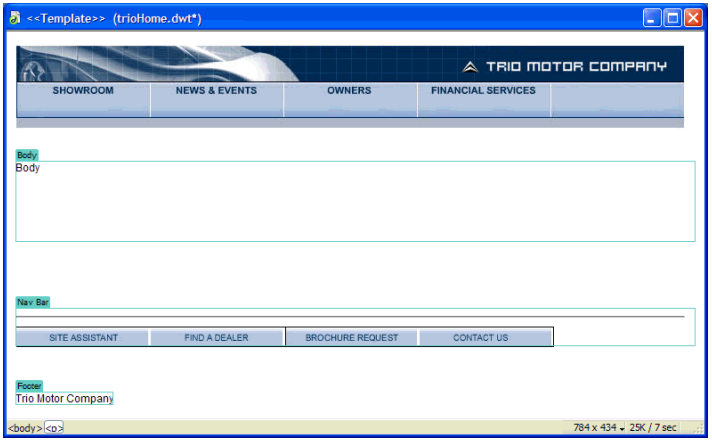
A nested template is a template whose design and editable regions are based on another template. To create a nested template, you must first save the original or base template, then create a new document based on the template, and then save that document as a template. In the new template, you can further 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.

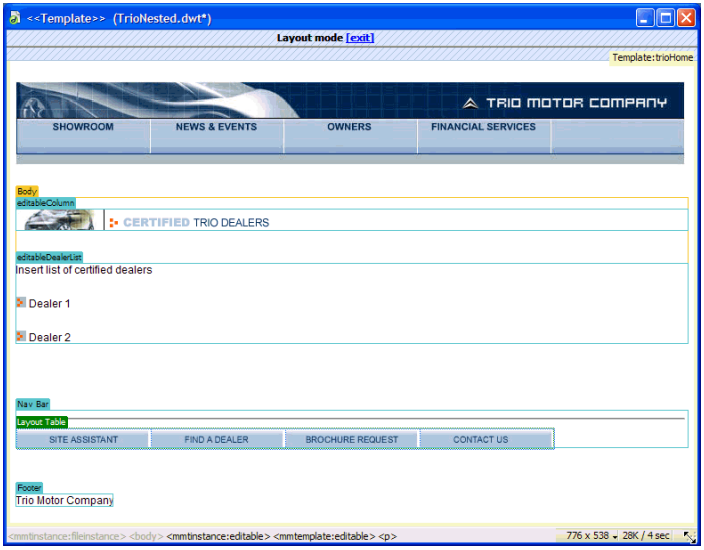
Editable regions in a base template are passed through to the nested template, and remain editable in pages created from a nested template unless new template regions are inserted in these regions.

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.

In the following example, the template contains three editable regions, named `Body`, `Nav Bar`, 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 *TrioNested*. In the nested template, we added two editable regions, with content, in the editable region named `Body`.



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 `editableColumn`, 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.

Related topics

- "Creating a nested template" on page 241

Server scripts in templates and template-based documents

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 are not 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 are not copied. As a result, Dreamweaver warns you if you change 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 in a template, changes to scripts before the `<html>` tag or after the `</html>` tag are copied to documents based on that template. However, you will no longer be able 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.

Template parameters

Template parameters indicate values for controlling content in documents based on a template. You can use template parameters with optional regions or editable tag attributes, or to set values you want to pass to an attached document. For each parameter, you select a name, a data type, and a default value. Each parameter must have a unique name; each name is case sensitive.

Template parameters are passed to the document as instance parameters. 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 determine what appears in the document, based on the value of a template expression.

Related topics

- "Using optional regions" on page 237
- "Defining editable tag attributes" on page 239

Template expressions

Template expressions are statements you use to compute or evaluate a value.

You can use an expression to store a value and display it in a document. For example, an expression can be as simple as the value of a parameter, such as `@@(Param)@@`, or complex enough to compute values which alternate the background color in a table row, such as `@@((_index & 1) ? red : blue)@@`.

You can also define expressions for if and multiple-if conditions (for an example, see "The Multiple If condition in template code" on page 226). When an expression is used in a conditional statement, Dreamweaver evaluates it as `true` or `false`. If the condition is true, the optional region appears in the template-based document; if it is false, it doesn't appear.

You can define expressions in Code view or in the Optional Region dialog box when you insert an optional region. For more information about writing template expressions, see "The template expression language" on page 225.

In Code view, there are two ways to define template expressions: use the `<!-- TemplateExpr expr="your expresson"-->` comment or `@@(your expression)@@`. When you insert the expression in the template code, an expression marker appears in Design view. When you apply the template, Dreamweaver evaluates the expression and displays the value in the template-based document.

The template expression language

The template expression language is a small subset of JavaScript, and uses JavaScript syntax and precedence rules. You can use JavaScript operators to write an expression like this:

```
@@(firstName+lastName)@@
```

The following features and operators are supported:

- numeric literals, string literals (double-quote syntax only), Boolean literals (`true` or `false`)
- variable reference (see Expression Object Model for the defined variables)
- field reference (the "dot" operator)
- unary operators: +, -, ~, !
- binary operators: +, -, *, /, %, &, |, ^, &&, ||, <, <=, >, >=, ==, !=, <<, >>
- conditional operator: ?:
- parentheses: ()

The following data types are used: Boolean, IEEE 64-bit floating point, string, and object. Dreamweaver templates do not support the use of JavaScript "null" or "undefined" types. Nor do they allow scalar types to be implicitly converted into an object; thus, the expression `"abc".length` would trigger an error, instead of yielding the value 3.

The only objects available are those defined by the expression object model. The following variables are defined:

_document

Contains the document-level template data. There is a field for each parameter in the template, as well as several fields providing built-in information about the document.

_repeat

Only defined for expressions which appear inside a repeating region. Provides built-in information about the region:

_index The numerical index (from 0) of the current entry

_numRows The total number of entries in this repeating region

_isFirst True if the current entry is the first entry in its repeating region

_isLast True if the current entry is the last entry in its repeating region

_prevRecord The `_repeat` object for the previous entry. It is an error to access this property for the first entry in the region.

_nextRecord The `_repeat` object for the next entry. It is an error to access this property for the last entry in the region.

_parent In a nested repeated region, this gives the _repeat object for the enclosing (outer) repeated region. It is an error to access this property outside of a nested repeated region.

During expression evaluation, all fields of the _document object and _repeat object are implicitly available. For example, you can enter `title` instead of `_document.title` to access the document's title parameter.

In cases where there is a field conflict, fields of the _repeat object take precedence over fields of the _document object. Therefore, you shouldn't need to explicitly reference _document or _repeat except that _document might be needed inside a repeat region to reference document parameters that are hidden by repeated region parameters.

When nested repeated regions are used, only fields of the innermost repeated regions are available implicitly. Outer regions must be explicitly referenced using _parent.

The Multiple If condition in template code

You can define template expressions for if and multiple-if conditions (see "Template expressions" on page 224). This example demonstrates defining a parameter named `"Dept"`, setting an initial value, and defining a Multiple If condition which determines which logo to display.

The following is an example of the code you might enter in the `head` section of the template:

```
<!-- TemplateParam name="Dept" type="number" value="1" -->
```

The following condition statement checks the value assigned to the Dept parameter. When the condition is true or matches, the appropriate image is displayed.

```
<!-- TemplateBeginMultipleIf -->
<!-- checks value of Dept and shows appropriate image-->

  <!-- TemplateBeginClause cond="Dept == 1" --> <img src=".../sales.gif"> <!--
    TemplateEndIfClause -->
  <!-- TemplateBeginIfClause cond="Dept == 2" --> <img src=".../support.gif">
    <!-- TemplateEndIfClause-->
  <!-- TemplateBeginIfClause cond="Dept == 3" --> <img src=".../hr.gif"> <!--
    TemplateEndIfClause -->
  <!-- TemplateBeginIfClause cond="Dept != 3" --> <img src=".../spacer.gif">
    <!-- TemplateEndIfClause -->

<!-- TemplateEndMultipleIf -->
```

When you create a template-based document, the template parameters are automatically passed to it. The template user determines which image to display (see "Modifying template properties" on page 249).

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.

After you create a template, you can insert template regions (see "Types of template regions" on page 216). You can also set template preferences for code coloring and template region highlight color (see "Customizing code coloring preferences for a template" on page 228 and "Setting highlighting preferences for template regions" on page 229).

Tip: You can store additional information about a template (such as who created it, when it was last changed, or why you made certain layout decisions) in a Design Notes file for the template (see "Associating Design Notes with files" on page 109). Documents based on a template do not inherit the template's Design Notes.

To create a template:

1 Open the document you want to save as a template:

- To open an existing document, select File > Open, then select the document.
- To open a new blank document, select File > New. In the dialog box that appears, select Basic Page or Dynamic Page, select the type of page you want to work with, then click Create.

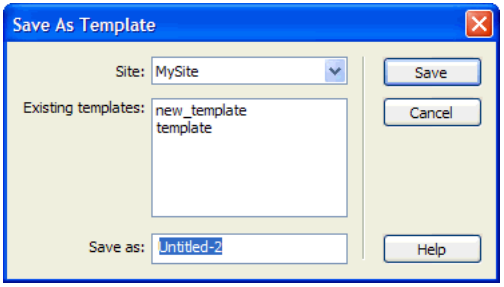
  Note: For more information about creating a new document, see "Creating a new blank document" on page 70.

2 When the document opens, do one of the following:

- Select File > Save as Template.
- In the Common category of the Insert bar, click the arrow on the Templates button, then select Make Template.

Note: Unless you selected Don't Show This Dialog Again in the past, 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.

The Save As Template dialog box appears.



3 Select a site to save the template in from the Site pop-up menu, then enter a unique name for the template in the Save As text box.

4 Click Save.

Dreamweaver saves the template file in the site's Templates folder in the local root folder of the site, with a .dwt filename extension. 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.

To use the Assets panel to create a new template

1 In the Assets panel (Window > Assets), select the Templates category on the left side of the panel.

The Templates category of the Assets panel appears.

2 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 Assets panel.

3 While the template is still selected, enter a name for the template, then press Enter (Windows) or Return (Macintosh).

Dreamweaver creates a new blank template in the Assets panel and in the Templates folder.

Related topics

- "Setting highlighting preferences for template regions" on page 229
- "Creating templates for a Contribute site" on page 229
- "Inserting an editable region" on page 232

Customizing code coloring preferences for a template

Code color preferences control the text, background color, and style attributes of the text displayed in Code view. You can set your own color scheme so you can easily distinguish template regions when you view a document in Code view.

To set a Code view color scheme for templates:

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

The Preferences dialog box appears.

2 Select Code Coloring from the category list on the left.

3 Select HTML from the Document Type list, then click the Edit Coloring Scheme button.

The Edit Code Coloring Scheme dialog box appears.

4 In the Styles For list select Template Tags.

5 Set color, background color, and style attributes for the Code view text by doing the following:

- If you want to change the text color, in the Text Color text box type the hexadecimal value for the color you want to apply to the selected text, or use the color picker to select a color to apply to the text. Do the same in the Background field to add or change an existing background color for the selected text.
- If you want to add a style attribute to the selected code, click the B (bold), I (italic), or U (underline) buttons to set the desired style.

6 Click OK.

Setting highlighting preferences for template regions

You can use the Dreamweaver highlighting preferences to customize the highlight colors for the outlines around the editable and locked regions of a template in Design view. The editable region color appears in the template as well as in documents based on the template.

For more information about viewing templates and template-based documents in Design view, see "Viewing templates in Design view" on page 217 and "Viewing template-based documents in Design view" on page 218.

To change template highlight colors:

1. Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

   The Preferences dialog box appears.
2. Select Highlighting from the category list on the left.
3. Click the Editable Regions, Nested Regions, or Locked Regions color box, then select a highlight color using the color picker (or enter the hexadecimal value for the highlight color in the text box).

   For information about using the color picker, see "Working with colors" on page 260.
4. (Optional) Repeat the process for other template region types, as necessary.
5. Click the Show option to enable or disable displaying colors in the Document window.

   Note: Nested Region does not have a Show option; its display is controlled by the Editable Region option.
6. Click OK.

To view highlight colors in the Document window:

- Select View > Visual Aids > Invisible Elements.

  Highlight colors appear in the document window only when View > Visual Aids > Invisible Elements is enabled and the appropriate options are enabled in Highlighting preferences.

  Note: If invisible elements are showing but the highlight colors are not, select Edit › Preferences (Windows) or Dreamweaver › Preferences (Macintosh), then select the Highlight category. Make sure that the Show option next to the appropriate highlight color is selected. Also make sure that the desired color is visible against the background color of your page.

Related topics

- "Customizing code coloring preferences for a template" on page 228

Creating templates for a Contribute site

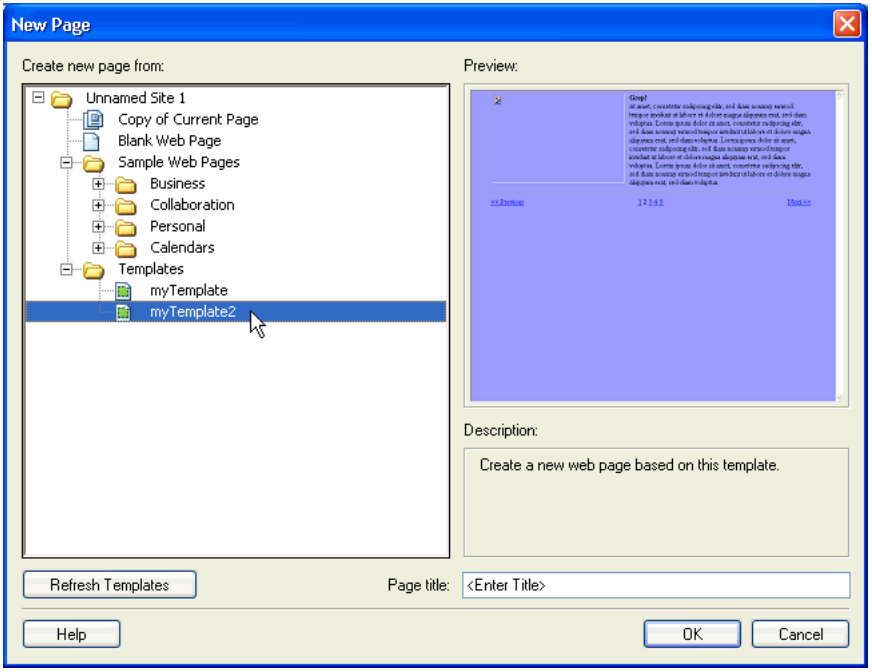
Using Dreamweaver, you can create templates to help Macromedia Contribute users create new pages, to provide a consistent look and feel for your site, and to enable you to update the layout of many pages at once.

When you create a template and upload it to the server, it becomes available to all Contribute users who connect to your site, unless you've set restrictions on template use for Contribute permission groups. If you have set restrictions on template use, you might need to add each new template to the list of templates a Contribute user can use (see Administering Contribute Help in Macromedia Contribute).

Note: Make sure that the site root folder defined in each Contribute user's site definition is the same as the site root folder defined in your site definition in Dreamweaver. If a user's site root folder doesn't match yours, that user won't be able to use templates.

In addition to Dreamweaver templates, you can create non-Dreamweaver templates using the Contribute administration tools. A non-Dreamweaver template is an existing page that Contribute users can use to create new pages; it's similar to a Dreamweaver template, except that pages that are based on it don't update when you change the template. Also, non-Dreamweaver templates can't contain Dreamweaver template elements such as editable, locked, repeating, and optional regions.

When a Contribute user creates a new document within a site containing Dreamweaver templates, Contribute lists the available templates (both Dreamweaver and non-Dreamweaver templates) in the New Page dialog box.



To include pages that use encodings other than Latin-1 in your site, you might need to create templates (either Dreamweaver templates or non-Dreamweaver templates). Contribute can edit pages that use any encoding, but when a Contribute user creates a new blank page, it uses the Latin-1 encoding. To create a page that uses a different encoding, a Contribute user can create a copy of an existing page that uses a different encoding, or can use a template that uses a different encoding. However, if there are no pages or templates in the site that use other encodings, then you must first create a page or template in Dreamweaver that uses that other encoding.

For information about creating, editing, and updating Dreamweaver templates, see "About Dreamweaver templates" on page 216.

To create a non-Dreamweaver template:

1 Select Site > Manage Sites.

The Manage Sites dialog box appears.

2 Select a site, then click Edit.

The Site Definition dialog box appears.

3 Click the Advanced tab.

4 Select the Contribute category from the category list on the left.

5 If you haven't already done so, you need to enable Contribute compatibility.

Select the Enable Contribute Compatibility option, then enter a site root URL.

For more information, click the Help button.

6 Click the Administer Site in Contribute button.

7 Do one of the following:

- If the Administrator Password dialog box appears, type the administrator password and click OK.
- If the site has no administrator, a dialog box appears asking if you want to become the website's administrator. Click Yes, then type and confirm an administrator password for the website, and click OK.

The Administer Website dialog box appears.

8 Select a permission group, then click the Edit Group button.

9 Select the New Pages category, then add existing pages to the list under the Create a New Page by Copying a Page from This List option.

For more information, see Administering Contribute help in Contribute.

10 Click OK to close the Permission Group dialog box.

11 Click OK to close the Administer Website dialog box.

Related topics

- "Creating a Dreamweaver template" on page 226

Creating editable regions

Editable template regions control which areas of a template-based page a user can edit.

Related topics

- "Types of template regions" on page 216

Inserting an editable region

Before you insert an editable region, you should save the document you are working in as a template (see "Creating a Dreamweaver template" on page 226).

Note: 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.

You can place an editable region anywhere in your page, but consider the following if you are making a table or a layer editable:

- 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 a `<td>` tag is selected, the editable region includes the region around the cell; if not, the editable region affects only 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, but making a layer's contents editable lets you change only the content of the layer, not its position.

To insert 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:
   - Select Insert > Template Objects > Editable Region.
   - Right-click (Windows) or Control-click (Macintosh), then select Templates > New Editable Region.
   - In the Common category of the Insert bar, click the arrow on the Templates button, then select Editable Region.

   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 text box.

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. A tab at the upper left corner of the region shows the name of the region. If you insert an empty editable region in the document, the name of the region also appears inside the region.

   Note: For information about setting template highlighting options, see "Setting highlighting preferences for template regions" on page 229.

Related topics

- "Removing an editable region" on page 233
- "Changing an editable region's name" on page 234

Selecting editable regions

You can easily identify and select template regions in both the template document and template-based documents.

To select an editable region in the Document window:

- Click the tab in the upper left corner of the editable region.

To find an editable region and select it in the document:

- Select Modify > Templates, then select the name of the region from the list at the bottom of that submenu.

  Note: 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.

The editable region is selected in the document.

Related topics

- "Setting highlighting preferences for template regions" on page 229
- "Inserting an editable region" on page 232
- "Changing an editable region's name" on page 234

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 in template-based documents) again, use the Remove Template Markup command.

To remove an editable region:

1. Click the tab in the upper left corner of the editable region to select it.
2. Do one of the following:
   - Select Modify > Templates > Remove Template Markup.
   - Right-click (Windows) or Control-click (Macintosh), then select Templates > Remove Template Markup.

   The region is now no longer editable.

Related topics

- "Inserting an editable region" on page 232

Changing an editable region's name

After you insert an editable region, you can later change its name.

To change the name of an editable region:

1 Click the tab in the upper left corner of the editable region to select it.

2 In the Property inspector (Window > Properties), enter a new name.

3 Press Enter (Windows) or Return (Macintosh).

Dreamweaver applies the new name to the editable region.

Related topics

- "Inserting an editable region" on page 232

Creating repeating regions

A repeating region is a section of a template that can be duplicated as often as desired in a template-based page. You typically use repeating regions with tables; however, you can define a repeating region for other page elements as well.

Repeating regions enable you to control your page layout by repeating certain items, such as a catalog item and description layout, or a row for data such as a list of items.

There are two repeating region template objects you can use: repeating region and repeating table.

Related topics

- "Types of template regions" on page 216

Creating a repeating region in a template

Repeating regions enable template users to duplicate a specified region in a template as often as desired. 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 in a template-based document), you must insert an editable region in the repeating region (see "Inserting an editable region" on page 232).

For information about creating an editable repeating table, see "Inserting a repeating table" on page 236.

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:

- Select Insert > Template Objects > Repeating Region.
- Right-click (Windows) or Control-click (Macintosh), then select Templates > New Repeating Region.
- In the Common category of the Insert bar, click the arrow on the Templates button, then select Repeating Region.

The New 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 template.

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 232.

Inserting a repeating table

You can use a repeating table to create an editable region (in table format) with repeating rows. You can define table attributes and set which table cells are editable.

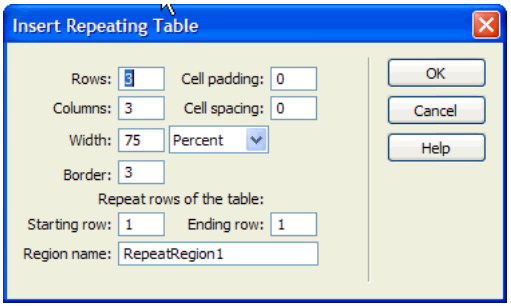
To insert a repeating table:

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:

- Select Insert > Template Objects > Repeating Table.
- In the Common category of the Insert bar, click the arrow on the Templates button, then select Repeating Table.

The Insert 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.

The repeating table appears in the template.

Setting alternating background colors in a repeating table

After you insert a repeating table in a template (see "Inserting a repeating table" on page 236), you can customize it by alternating the background color of the table rows.

To set alternating table row background colors:

1 In the Document window, select a row in the repeating table.

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.

4 Save the template.

The following 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>
```

Using optional regions

An optional region is a region in a template that users can 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.

When you insert an optional region, you can set specific values for a template parameter or define conditional statements in a template; you can later modify the optional region if necessary. Based on the conditions you define, template users can edit the parameters in template-based documents they create and control whether the optional region displays or not (see "Modifying template properties" on page 249).

Related topics

- "Types of template regions" on page 216

Inserting an optional region

Use an optional region to control content which may or may not be shown in a template-based document. There are two optional region objects:

- An optional region enables template users to show and hide specially marked regions without enabling them to edit the content.

  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 enables template users to set whether the region shows or hides and enables them to edit content in the 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. An editable region is controlled by a conditional statement.

Note: To set optional regions to show or hide in template-based documents, see "Modifying template properties" on page 249.

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:

- Select Insert > Template Objects > Optional Region.
- Right-click (Windows) or Control-click (Macintosh) the selected content, then select Templates > New Optional Region.
- In the Common category of the Insert bar, click the arrow on the Templates button, then select Optional Region.

The Optional Region dialog box appears.

3 Specify 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, place the insertion point where you want to insert the optional region.

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:

- Select Insert > Template Objects > Editable Optional Region.
- In the Common category of the Insert bar, click the arrow on the Templates button, then select Editable Optional Region.

The Optional Region dialog box appears.

3 Enter a name for the optional region, then click the Advanced tab if you want to set other options.

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.

Note: To set optional regions to show or hide in template-based documents, see "Modifying template properties" on page 249.

To reopen the Optional Region dialog box:

1 In the Document window, do one of the following:
- In Design view, click the template tab of the optional region you want to modify.
- In Design view, place the insertion point in the template region; then in the tag selector at the bottom of the Document window, select the template tag, `<mmtemplate:if>`.
- In Code view, click the comment tag of the template region you want to modify.

2 In the Property inspector (Window > Properties), click Edit.

The Optional Region dialog box appears.

3 Make changes as necessary.

For information about option in the dialog box, click the Help button in the dialog box.

4 Click OK.

Defining editable tag attributes

You can enable 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 enable template users to set a different background color for pages they create. Users can update only the attributes you specify as editable.

Note: To modify editable tag attributes in template-based documents, see "Modifying template properties" on page 249.

Related topics

- "Types of template regions" on page 216

Specifying editable tag attributes in a template

You can set multiple editable attributes in a page so that template users can modify the attributes in template-based documents. The following data types are supported: text, Boolean (`true`/`false`), color, and URL.

To define an editable tag attribute:

1. In the Document window, select an item you want to set an editable tag attribute for.
2. Select Modify > Templates > Make Attribute Editable.

   The Editable Tag Attributes dialog box appears.
3. Complete the dialog box for each attribute you want to make editable.

   For information about completing the dialog box, click the Help button in the dialog box.
4. Click OK.

   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 template-based document (see "Modifying template properties" on page 249).

Making an editable tag attribute uneditable

A tag previously marked as editable can be marked as uneditable.

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. Select Modify > Templates > Make Attribute Editable.

   The Editable Tag Attributes dialog box appears.
3. In the Attributes pop-up menu, select the attribute you want to affect.
4. Deselect the Make Attribute Editable checkbox.
5. Click OK.
6. Update documents based on the template.

Related topics

- "Specifying editable tag attributes in a template" on page 240

Creating a nested template

Nested templates let you create variations of a base template. You create a nested template by saving a document based on a template, then saving that document as a new template. You can nest multiple templates to define increasingly specific layouts. For more information, see "Nested templates" on page 222.

By default, all editable template regions from the base template pass through the nested template to the document based on the nested template. That means that if you create an editable region in a base template, then create a nested template, the editable region appears in documents based on the nested template (if you did not insert any new template regions in that region in the nested template).

In nested templates, pass-through editable regions have a blue border. You can insert template markup inside an editable region so that it won't pass through as an editable region in documents based on the nested template. Such regions have an orange border instead of a blue border.

To create a nested template:

1 Create a document from the template on which you want to base the nested template:

- In the Assets panel's Templates category, right-click (Windows) or Control-click (Macintosh) the template you want to create a new document from, then select New From Template from the context menu.
- Select File > New. In the New Document dialog box, click the Templates tab, then select the site that contains the template you want to use; in the document list, double-click the template to create a new document.

A new document appears in the Document window.

2 Do one of the following to save the new document as a nested template:

- Select File > Save as Template.
- In the Common category of the Insert bar, click the arrow on the Templates button, then select Make Nested Template.

The Save As Template dialog box appears.

3 Enter a name in the Save As text box, then click OK.

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.

To prevent an editable region from passing through to a nested template:

1 In Code view, locate the editable region you want to prevent from passing through.

Editable regions are defined by template comment tags.

2 Wrap the editable region (including the comment tags) with the following markers:

```
@@"  "@@
```

Editing and updating templates

When you make changes to and save a template, Dreamweaver automatically updates all documents attached to the template. You can also manually update documents based on a template, if necessary.

Note: To edit a template for a Contribute site, you must use Dreamweaver; you cannot edit templates in Contribute.

Dreamweaver automatically checks template syntax when you save a template. You might also want to check template syntax when you edit a template.

Note: For information about editing template-based documents, see "Editing content in a template-based document" on page 248.

Opening a template for editing

You can open a template file directly for editing, or you can open a template-based document, then open the attached template for editing.

When you make a change to a template, Dreamweaver prompts you to update the documents based on the template.

Note: You can also manually update the documents for template changes if necessary (see "Manually updating documents that are based on templates" on page 243).

To open and edit a template file:

1 In the Assets panel (Window > Assets), select the Templates category on the left side of the panel.

The Assets panel lists all of the available templates for your site and displays a preview of the selected template.

2 In the list of available templates, do one of the following:

- Double-click the name of the template you want to edit.
- Select a template to edit, then click the Edit button at the bottom of the Assets panel.

The template opens in the Document window.

3 Modify the content of the template as desired.

Tip: To modify the template's page properties, select Modify › Page Properties. (Documents based on a template inherit the template's page properties.)

4 Save the template.

Dreamweaver prompts you to update pages based on the template.

5 Click Update to update all documents based on the modified template; click Don't Update if you don't want to update documents based on the modified template.

Dreamweaver displays a log indicating the files that were updated.

To open and modify the template attached to the current document:

1 Open the template-based document in the Document window.

2 Do one of the following:

- Select Modify > Templates > Open Attached Template.
- Right-click (Windows) or Control-click (Macintosh), then select Templates > Open Attached Template.

3 Modify the content of the template as desired.

Tip: To modify the template's page properties, select Modify › Page Properties. (Documents based on a template inherit the template's page properties.)

4 Save the template.

Dreamweaver prompts you to update pages based on the template.

5 Click Update to update all documents based on the modified template; click Don't Update if you don't want to update documents based on the modified template.

Dreamweaver displays a log indicating the files that were updated.

Related topics

- "Updating templates in a Contribute site" on page 244
- "Checking template syntax" on page 244

Manually updating documents that are based on templates

When you make a change to a template, Dreamweaver prompts you to update the documents based on the template, but you can manually update the current document or the entire site if necessary. Manually updating template-based documents is the same as reapplying the template.

To apply template changes to the current template-based document:

1 Open the document in the Document window.

2 Select Modify > Templates > Update Current Page.

Dreamweaver updates the document with any template changes.

To update the entire site or all documents that use a specified template:

1 Select Modify > Templates > Update Pages.

The Update Pages dialog box appears.

2 Complete the dialog box, then click Start.

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

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.

3 Click Close to close the dialog box.

Related topics

- "Opening a template for editing" on page 242
- "Checking template syntax" on page 244

Updating templates in a Contribute site

Contribute users can't make changes to a Dreamweaver template. You can, however, use Dreamweaver to change a template for a Contribute site

The following are important factors to keep in mind when updating templates in a Contribute site:

- Contribute retrieves new and changed templates from the site only when Contribute starts up and when a Contribute user changes their connection information. If you make changes to a template while a Contribute user is editing a file based on that template, the user won't see the changes to the template until they restart Contribute.
- If you remove an editable region from a template, a Contribute user editing a page based on that template might be confused about what to do with the content that was in that editable region.

To edit a template in a Contribute site:

1 Edit the template using Dreamweaver.

For more information, see "Opening a template for editing" on page 242.

2 Tell all of the Contribute users who are working on the site to exit Contribute and restart it.

Checking template syntax

Dreamweaver automatically checks the template syntax when you save a template, but you can manually check the template syntax prior to saving a template. For example, if you add a template parameter or expression in Code view, you can check that the code follows correct syntax.

To check for valid template syntax:

1 Open the document you want to check in the Document window.

2 Select Modify > Templates > Check Template Syntax.

An error message 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.

Related topics

- "Template tag syntax" on page 220
- "Template expressions" on page 224

Managing templates

Use the Templates category of the Assets panel to manage existing templates, including renaming template files and deleting template files.

Note: You can also use the Assets panel to apply a template to a document (see"Editing and updating templates" on page 242) or edit a template (see "Opening a template for editing" on page 242).

To rename a template in the Assets panel:

1 In the Assets panel (Window > Assets), select the Templates category on the left side of the panel.

2 Click the name of the template to select it.

3 Click the name again so that the text is selectable, then enter a new name.

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.

4 Click in another area of the Asset panel, or press Enter (Windows) or Return (Macintosh) for the change to take effect.

Dreamweaver asks if you want to update documents that are based on this template.

5 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.

To delete a template file:

1 In the Assets panel (Window > Assets), select the Templates category on the left side of the panel.

2 Click the name of the template to select it.

3 Click the Delete button at the bottom of the panel, then confirm that you want to delete the template.

Caution: Once you delete a template file, you can't retrieve it. The template file is deleted from your site.

Documents that are based on a deleted 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, see "Detaching a document from a template" on page 248.

Related topics

- "Creating a Dreamweaver template" on page 226
- "Applying or removing a template from an existing document" on page 247
- "Editing and updating templates" on page 242

Exporting and importing template XML content

You can think of a document based on a template as containing data represented by name/value pairs. Each pair consists of the name of an editable region, and the contents of that region.

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.

To export a document's editable regions as XML:

1. Open a template-based document that contains editable regions.
2. Select File > Export > Template Data as XML.

   The Export Template Data as XML dialog box appears.
3. Select one of the Notation options:
   - If the template contains repeating regions or template parameters, select Use Standard Dreamweaver XML Tag.
   - If the template does not contain repeating regions or template parameters, select Use Editable Region Names as XML Tags.
4. Click OK.

   A dialog box appears for you to save the XML file.
5. Select a folder location, enter a name for the XML file, then 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. Select File > Import > Import XML into Template.

   The Import XML dialog box appears.
2. Select the XML file, then 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.

Tip: If your XML file isn't set up exactly the way Dreamweaver expects, you might 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.

Exporting a site without template markup

If you do not want to include template markup in the template-based documents you export to another site, use the Export Site Without Template Markup command.

To export a site without template markup:

1 Select Modify > Templates > Export Without Markup.

The Export Site Without Template Markup dialog box appears.

2 Select a folder to export the site to, then select additional export options as desired.

Note: You must select a folder outside of the current site.

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

3 Click OK.

Applying or removing a template from an existing document

When you apply a template to an existing document, Dreamweaver matches the content to regions of the template or asks you to resolve mismatches. You can later remove the template if you need to make changes to locked regions.

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.

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 so you can select which region or regions to move the current page's content to, or you can delete the mismatched content.

You can apply a template to an existing document using the Assets panel or from the Document window. You can undo a template application if necessary.

To apply a template to an existing document using the Assets panel:

1 Open the document you want to apply the template to.

2 In the Assets panel (Window > Assets), select the Templates category on the left side of the panel.

3 Do one of the following:

- Drag the template you want to apply from the Assets panel to the Document window.
- Select the template you want to apply, then click the Apply button at the bottom of the Assets panel.

If content exists in the document that can't be automatically assigned to a template region, the Inconsistent Region Names dialog box appears.

4 If you have unresolved content, select a destination for the content, then click OK.

For information about moving existing content to editable regions in the document, see Resolving inconsistent region names in Using Dreamweaver Help.

To apply a template to an existing document from the Document window:

1 Open the document you want to apply the template to.

2 Select Modify > Templates > Apply Template to Page.

The Select Template dialog box appears.

3 Choose a template from the list, then click Select.

If content exists in the document that can't be automatically assigned to a template region, the Inconsistent Region Names dialog box appears.

4 If you have unresolved content, select a destination for the content, then click OK.

For information about moving existing content to editable regions in the document, see Resolving inconsistent region names in Using Dreamweaver Help.

To undo the template changes:

- Select Edit > Undo Apply Template.

The document reverts to its state before the template was applied.

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 entire document becomes editable.

To detach a document from a template:

1 Open the template-based document you want to detach.

2 Select Modify > Templates > Detach from Template.

The document is detached from the template and all template code is removed.

Related topics

- "Applying a template to an existing document" on page 247

Editing content in a template-based document

Dreamweaver templates specify regions that are locked (uneditable) and others that are editable for template-based documents (see "About Dreamweaver templates" on page 216).

In pages based on templates (see "Creating a document based on an existing template" on page 71), template users can edit content in editable regions only. You can easily identify and select editable regions to edit content (see "Selecting editable regions" on page 233). Template users cannot edit content in locked 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.

Template users can also modify properties and edit entries for a repeating region in template-based documents.

Modifying template properties

When template authors create parameters in a template (see "Template parameters" on page 224), 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).

To modify an editable tag attribute:

1 Open the template-based document.

2 Select Modify > Template Properties.

The Template Properties dialog box opens, showing a list of available properties. The dialog box shows optional regions and editable tag attributes.

3 In the Name list, select the property.

The bottom area of the dialog box updates to show the selected property's label and its assigned value.

4 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 that do not appear in the Name list are not updateable in the template-based document.

5 Select the Allow Nested Templates to Control This checkbox, if you want to pass the editable property along to a documents based on the nested template.

To modify optional region template parameters:

1 Open the template-based document.

2 Select Modify > Template Properties.

The Template Properties dialog box opens, showing a list of available properties. The dialog box shows optional regions and editable tag attributes.

3 In the Name list, select a property.

The dialog box updates to show the selected property's label and its assigned value.

4 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,.

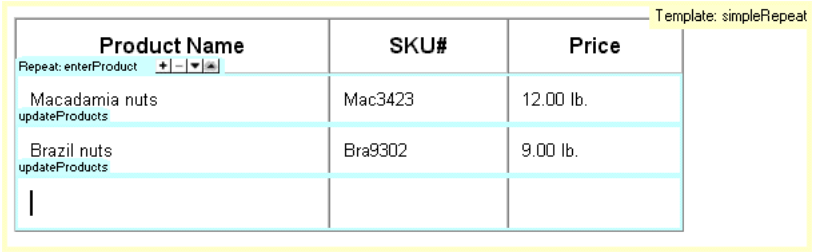
5 Select the Allow Nested Templates to Control This checkbox, if you want to pass the editable property along to a documents based on the nested template.

Related topics

- "Defining editable tag attributes" on page 239
- "Using optional regions" on page 237

Adding, deleting, and changing the order of a repeating region entry

Use repeating region controls to add, delete, or change the order of entries in template-based documents. When you add a repeating region entry, a copy of the entire repeating region is added. To update the content in the repeating regions, the original template must include an editable region in the repeating region.



To add, delete, or change the order of a repeating region:

1 Place the insertion point in the repeating region to select it.

2 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.

Note: Alternatively, you can select Modify › Template, then select one of the repeating entry options near the bottom of the context 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:

1 Place the insertion point in the repeating region to select it.

2 Do one of the following:

- To cut a repeating entry, select Edit > Repeating Entries > Cut Repeating Entries.
- To copy a repeating entry, select Edit > Repeating Entries > Copy Repeating Entries.
- To remove a repeating entry, select Edit > Repeating Entries > Delete Repeating Entries.
- To paste a repeating entry, select Edit > Paste.

  Note: Pasting inserts a new entry; it does not replace an existing entry.

Related topics

- "Creating repeating regions" on page 234

PART IV
Adding Content to Pages

Use the visual tools in Macromedia Dreamweaver MX 2004 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 12, “Working with Pages,” on page 253

Chapter 13, “Inserting and Formatting Text,” on page 273

Chapter 14, “Inserting Images,” on page 303

Chapter 15, “Linking and Navigation,” on page 315

Chapter 16, “Working with Other Applications,” on page 341

Chapter 17, “Adding Audio, Video, and Interactive Elements,” on page 355

Chapter 18, “Using JavaScript Behaviors,” on page 371

CHAPTER 12
Working with Pages

Macromedia Dreamweaver MX 2004 provides many features to help you create new web pages. These page creation features can help you more easily specify web page properties such as page titles, background images and colors, and text and link colors. In addition, Dreamweaver MX 2004 provides tools to help you maximize website performance, and to create and test pages to ensure compatibility with different web browsers.

This chapter contains the following sections:

- "About working with pages" on page 253
- "Saving web pages" on page 257
- "Specifying HTML instead of CSS" on page 258
- "Setting page properties" on page 258
- "Working with colors" on page 260
- "Selecting elements in the Document window" on page 262
- "Using the History panel" on page 263
- "Automating tasks" on page 265
- "Using JavaScript behaviors to detect browsers and plug-ins" on page 270
- "Previewing and testing pages in browsers" on page 270
- "Setting download time and size preferences" on page 272

About working with pages

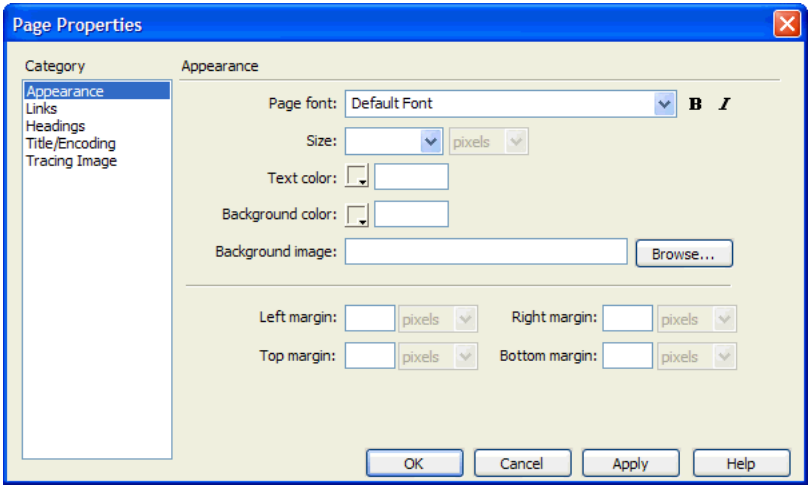
When creating a web page, you must consider what browsers and operating system users will view your web page with, and what language sets you may need to support. The following sections will help you understand how to select colors that are displayed correctly in different web browsers, the encoding of different characters (letter forms) for different languages, and how to check that a web browser is compatible with your website.

About setting page properties

For each page you create in Dreamweaver, you can specify layout and formatting properties using the Page Properties dialog box (Modify > Page Properties). The Page Properties dialog box lets you specify the default font family and font size, background color, margins, link styles, and many other aspects of page design. You can assign new page properties for each new page you create, and modify those for existing pages

By default Dreamweaver formats text using CSS (Cascading Style Sheets). You can change the page-formatting preferences to HTML formatting using the Preferences dialog box (Edit > Preferences). When using CSS page properties, Dreamweaver uses CSS tags for all properties defined in the Appearance, Links, and Headings categories of the Page Properties dialog box. The CSS tags defining these attributes are embedded in the `head` section of the page.

Note: The page properties you choose apply only to the active document. If a page uses an external CSS style sheet, Dreamweaver will not overwrite the tags set in the style sheet, as this affects all other pages using that style sheet.



CSS versus HTML page properties

By default, Dreamweaver uses CSS to assign page properties. If you want to use HTML tags instead, you must specify this in the Preferences dialog box (see "Specifying HTML instead of CSS" on page 258 for more information).

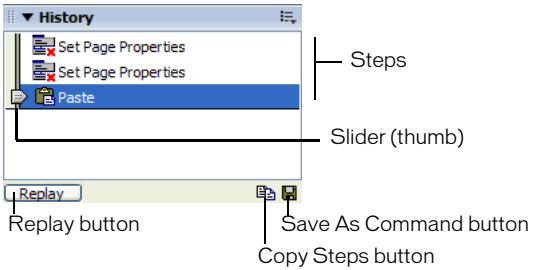
If you choose to use HTML instead of CSS, the Property inspector still displays the Style pop-up menu. However, the font, size, color, and alignment controls will only show properties set using HTML tags. The values of CSS properties applied to the current selection will no longer be visible, and the Size pop-up menu will be disabled.

Related topics

- "Specifying HTML instead of CSS" on page 258
- "Setting page properties" on page 258

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 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.

Related topics

- "Using the History panel" on page 263

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).

When web browsers first made their appearance, most computers displayed only 265 colors (8-bit). Today, the majority of computers display thousands or millions of colors (16- and 32-bit), so the justification for using the browser-safe palette is greatly diminished if you are developing your site for users with current computer systems.

One reason to use the web-safe color palette is if you will be developing for alternative web devices such as PDA and cell phone displays. Many of this devices offer only black and white (1-bit) or 256 color (8-bit) displays.

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 select 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.

Related topics

- "Working with colors" on page 260

Understanding document encoding

Document encoding specifies the encoding used for characters in the document. Document encoding is specified in a `meta` tag in the head of the document; it tells the browser and Dreamweaver how the document should be decoded and what fonts should be used to display the decoded text.

For example, if you specify Western European (Latin1), this `meta` tag is inserted: `<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">`. Dreamweaver displays the document using the fonts you specify in Fonts Preferences for the Western European (Latin1) encoding; a browser displays the document using the fonts the browser user specifies for the Western European (Latin1) encoding.

If you specify Japanese (Shift JIS), this `meta` tag is inserted: `<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">`. Dreamweaver displays the document using the fonts you specify for the Japanese encodings; a browser displays the document using the fonts the browser user specifies for the Japanese encodings.

To change document encoding for a page, see "Setting page properties" on page 258. To change the default encoding that Dreamweaver uses to create new documents, see "Setting a default new document type" on page 72. To change the fonts that Dreamweaver uses to display each encoding, see "Setting Fonts preferences for Dreamweaver display" on page 56.

About using saved commands versus playing back steps

Dreamweaver enables you to record a temporary command for short-term use, or you can play back steps from the History panel.

When you record a temporary command:

- 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.

Related topics

- "Automating tasks" on page 265
- "Repeating steps" on page 265
- "Recording commands" on page 269

Saving web pages

You can save a document using its current name and location, or save a copy of a document using a different name and location.

When naming files, 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 Do one of the following:
   - To overwrite the current version on the disk, and save any changes you have made, select File > Save.
   - To save the file in a different folder or using a different name, select File > Save As.
2 In the Save As dialog box that appears, navigate to the folder where you want to save the file.
3 In the File Name text box, type a name for the file.
4 Click Save to save the file.

To save all open documents:

1 Select File > Save All.
2 If there are any unsaved documents open, the Save As dialog box is displayed for each unsaved document.

   In the dialog box that appears, navigate to the folder where you want to save the file.
3 In the File Name text box, type a name for the file.
4 Click Save to save the file.

To revert to the last saved version of a document:

1 Select File > Revert.

   A dialog box will prompt you, asking if you want to discard your changes, and revert to the previously saved version.
2 Click Yes to revert to the previous version; click No to keep your changes.

   Note: If you save a document, and then exit Dreamweaver, you cannot revert to the previous version of the document when you restart Dreamweaver.

Related topics

- "Creating new documents" on page 70
- "Saving a new document" on page 72

Specifying HTML instead of CSS

By default, Dreamweaver uses CSS tags to assign page properties. If you want to use HTML tags instead, you must deselect the Use CSS Instead of HTML Tags option in the General category of the Preferences dialog box.

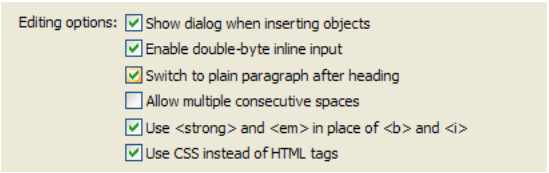
To specify HTML instead of CSS tags for page properties:

1 Select Edit > Preferences.

The Preferences dialog box is displayed.

2 In the General category of the Page Properties dialog box, deselect the Use CSS Instead of HTML Tags checkbox.

The checkbox is located in the Editing options section of the General Preferences panel.



3 Click OK.

Related topics

- "About setting page properties" on page 254

Setting page properties

Page titles, background images and colors, text and link colors, and margins are basic properties of every web document. You can set or change page properties using the Page Properties dialog box.

To set document properties:

1 Select Modify > Page Properties, or click the Page Properties button in the text Property inspector.

The Page Properties dialog box opens.

2 Make changes to the page properties, as desired.

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

3 Click OK.

Related topics

- "About setting page properties" on page 254
- "Setting CSS link underline styles" on page 259
- "Setting a background image or background page color" on page 260

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: 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 "Testing your site" on page 111.)

To change the title of a page:

1 With the document open, do one of the following:
- Select Modify > Page Properties.
- Select View > Toolbar > Document (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 (Windows) or Return (Macintosh).

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 Document 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.

Related topics

- "About setting page properties" on page 254

Setting CSS link underline styles

If you are using the default CSS Page Properties dialog box, Dreamweaver makes it easy to specify special CSS link styles. Using the CSS link options, you can choose to never underline links, underline links only when the pointer passes over the link, or turn off links when the pointer passes over them.

To set a CSS link style:

1 Select Modify > Page Properties, or select Page Properties from the context menu in the Design view of the Document window.

2 Select the Links category in the CSS Page Properties dialog box.

3 From the Underline Style pop-up menu, select the underline style you would like to use as the default for your page.

4 Click OK.

Related topics

- "About setting page properties" on page 254
- Chapter 15, "Linking and Navigation," on page 315

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 Select Modify > Page Properties, or select Page Properties from the context menu in the Design view of the Document window.

2 Select the Appearance category in the Page Properties dialog box.

3 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.)

4 To set a background color, click the Background color box and select a color from the color picker.

Related topics:

- "About setting page properties" on page 254

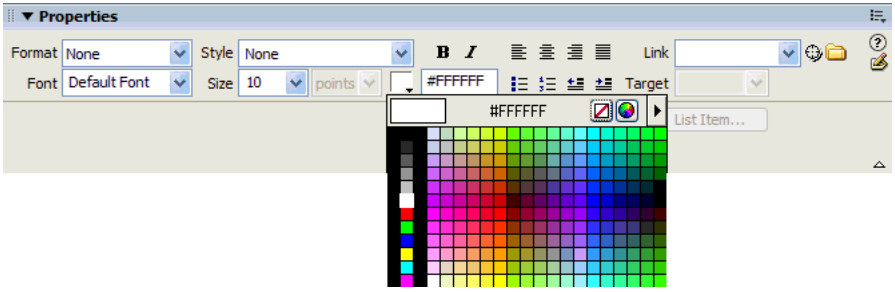
Working with colors

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 select a color for a page element.

To select a color in Dreamweaver:

1 Click a color box in any dialog box or in the Property inspector.

The color picker appears.



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 255.

- Use the eyedropper to pick up a color from anywhere on your screen—even outside the Dreamweaver windows. To pick up a color from the desktop or another application, press and hold the mouse button; this allows the eyedropper to retain focus, and select a color outside of Dreamweaver. If you click the desktop or another application, Dreamweaver picks up the color where you clicked. However, if you switch to another application, you may need to click a Dreamweaver window to continue working in 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: 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 select 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 Default Colorbutton.

To open the system color picker, click the Color Wheel button. For more information, see "About web-safe colors" on page 255.

Defining default text colors

Define default colors for regular text, links, visited links, and active links in the Page Properties dialog box, or select a preset color scheme to define the page background and text colors. (See "Working with colors" on page 260.)

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:

- Select Modify > Page Properties and then select colors for the Text Color, Link Color, Visited Links, and Active Links options.
- Select Commands > Set Color Scheme and then select a background color and a color set for text and links.

  The sample box shows how the color scheme will look in the browser.

Note: If you define these settings using the Page Properties dialog box, using the default CSS tags, the Color Scheme command will not affect your page's appearance. This is because CSS tags take priority over HTML tags.

Related topics

- "About setting page properties" on page 254

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. For more information about invisible elements, see "Showing and hiding invisible elements" on page 263.

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, select 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 "Showing and hiding invisible elements" on page 263.)
- 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 containing the paragraph you want to select, select the `<table>` tag to the left of the `<p>` tag.

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 > Code Inspector.

For more information about Code view, see "Viewing your code" on page 401.

When you select something in either code editor (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 401.

Showing and hiding 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 enables you to specify whether it shows icons marking the location of invisible elements in the Design view of the Document window. To indicate which element markers appear when you select View > Visual Aids > Invisible Elements, you can set options in Invisible Elements preferences. For example, you can specify that named anchors be visible, but not line breaks.

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.

To show or hide marker icons for invisible elements:

- Select View > Visual Aids > Invisible Elements.

Note: 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 change Invisible Elements preferences:

1. Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh), then click Invisible Elements.
2. Select which elements should be made visible.

   A check mark next to the name of the element in the dialog box means the element is visible when View > Visual Aids > Invisible Elements is selected.

   For an explanation of each Invisible Elements preference, see Setting Invisible Elements preferences in Dreamweaver Help.
3. Click OK.

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.

To undo the last operation you performed in a document, select 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 "Automating tasks" on page 265.

To open the History panel:

- Select Window > 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. Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).
2. Select General from the Category list on the left.
3. 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, select 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.)

Related topics

- "About the History panel" on page 255

Automating tasks

While creating your documents, you may want to perform the same task numerous times.

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 255.) 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.

Tip: 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.

Related topics

- "Applying steps to another object" on page 266
- "Applying steps to multiple objects" on page 266
- "Copying and pasting steps between documents" on page 268
- "Creating new commands from history steps" on page 268
- "Recording commands" on page 269

Repeating steps

You can use the History panel to repeat the last step you performed, repeat a series of adjacent steps, or repeat a series of nonadjacent steps. (For basic information about the History panel, see "About the History panel" on page 255.)

To repeat one step, do one of the following:

- Select Edit > Redo.

  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 Redo Typing.
- 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.

The steps played are the steps that are selected (highlighted), not necessarily the step the slider currently points to.

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, 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.

Related topics

- "Applying steps to multiple objects" on page 266

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 and 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.

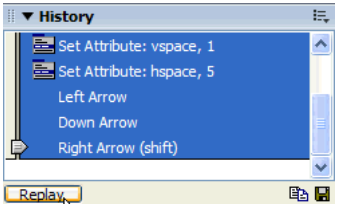
4 In the Property inspector, enter numbers in the V Space and H Space text boxes to set the image's spacing.

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.

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.

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 268) before you close the document.

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.

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.

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.

4 Open the other document.

5 Place the insertion point where you want it, or select an object to apply the steps to.

6 Select 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.

If you pasted steps into a text editor or into 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 423.

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 a chance you'll 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. 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 select 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. If you are using Dreamweaver on a multiuser operating system, the file will be saved in the specific user's 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 Select the command from the Commands menu.

To edit the names of commands in the Commands menu:

1 Select 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 Select Commands > Edit Command List.

2 Select a command.

3 Click Delete, then click Close.

Recording commands

You can record a temporary command for short-term use. 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. For more information on comparing using recorded commands to playing back steps from the History panel, see "Using the History panel" on page 263.

To temporarily record a series of frequently used steps:

1 Select 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, select Commands > Stop Recording, or press Control+Shift+X (Windows) or Command+Shift+X (Macintosh).

To play back a recorded command:

- Select Commands > Play Recorded Command, or press Control+Shift+R (Windows) or Command+Shift+R (Macintosh).

To save a recorded command:

1 Select 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.

The command appears in the Commands menu.

Using JavaScript 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 Chapter 18, "Using JavaScript Behaviors," on page 371.

Check Browser sends visitors to different pages depending on their browser brands and versions (see "Check Browser" on page 378). For example, you may 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 379). For example, you may want visitors to go to one page if they have Macromedia Shockwave and another page if they do not.

Previewing and testing pages in browsers

You can preview a document in your target browsers at any time; you don't have to save the document first.

You can define up to 20 browsers for previewing. It's a good idea to preview in the following browsers: Internet Explorer 6.0, Netscape Navigator 7.0, and the Macintosh-only Safari browser. In addition to these more popular, graphical browsers, you may want to test your pages using a text-only browser such as Lynx.

To preview and test your document in a browser:

1 Do one of the following to preview the page:

- Select File > Preview in Browser, then select one of the listed browsers.

  Note: If you haven't selected a browser yet, select Edit › Preferences or Dreamweaver › Preferences (Macintosh), 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.

2 Click links and test content in your page.

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.

Note: 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 and testing pages in browsers" on page 270.) This is because browsers don't recognize site roots—servers do.

Tip: To preview content linked with root-relative paths, put the file on a remote server, then select File > Preview in Browser to view it (see "Site root-relative paths" on page 318).

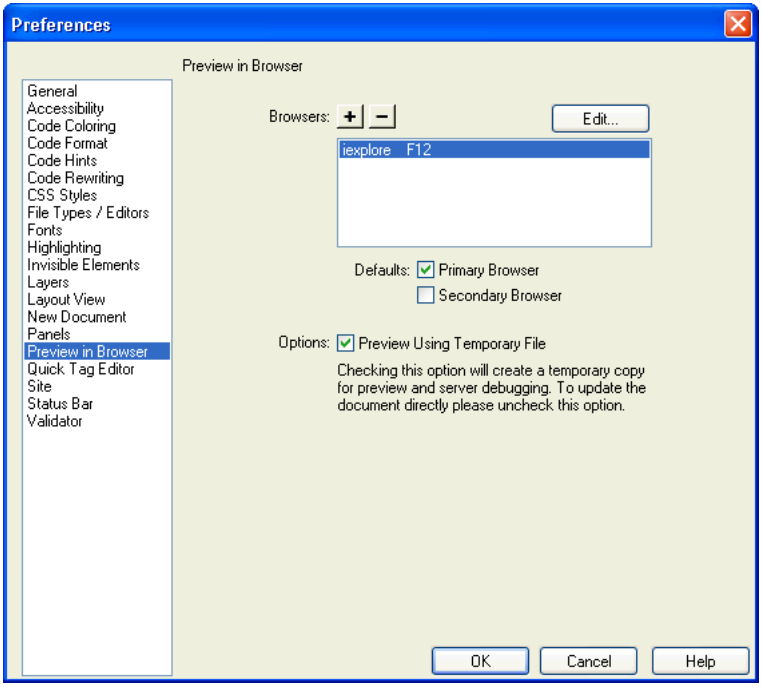
3 Close the page in the browser when you finish testing.

To set or change preferences for your primary and secondary browsers:

1 Do one of the following to open the Preview in Browser options:

- Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh), and then select Preview in Browser from the category list on the left.
- Select 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.

Setting download time and size preferences

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.

Tip: 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.

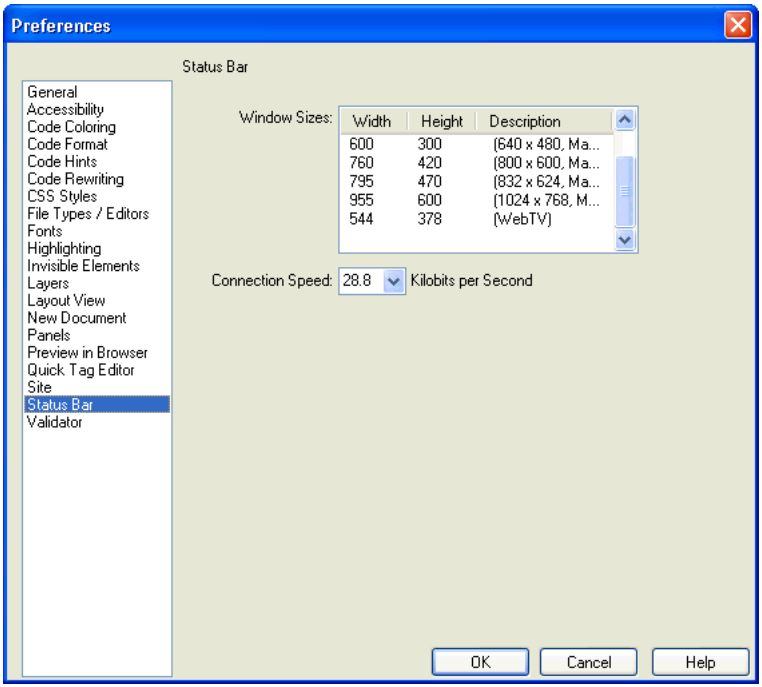
To set download time and size preferences:

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

The Preferences dialog box appears.

2 Select Status Bar from the Category list on the left.

The Status Bar preference options appear.



3 Select 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 select 1500 (T1 speed).

For more information about status bar preferences, see "Setting Status Bar preferences" on page 41.

4 Click OK.

CHAPTER 13
Inserting and Formatting Text

Macromedia Dreamweaver MX 2004 offers several ways for you to add and format text in a document. You can insert text, set font type, size, color, and alignment attributes, as well as create and apply your own custom styles using Cascading Style Sheet (CSS) styles.

This chapter covers the following topics:

- "About text formatting in Dreamweaver" on page 273
- "Inserting text" on page 280
- "Formatting paragraphs and document structure" on page 282
- "Formatting text" on page 285
- "Using Cascading Style Sheets styles" on page 290
- "Updating CSS style sheets in a Contribute site" on page 300
- "Checking spelling" on page 301
- "Searching for and replacing text" on page 301

About text formatting in Dreamweaver

Dreamweaver provides several tools and commands that let you format text using either CSS styles or HTML tags.

About inserting text

Dreamweaver lets you add text to web pages by typing the text directly into a page, copying and pasting text from another document, or dragging text from another application. Typical document types that web professionals receive with text content that needs incorporating into web pages include ASCII text files, rich format text files, and MS Office documents. Dreamweaver lets you take text from any of these document types, and incorporate it into a web page.

Related topics

- "Inserting text" on page 280
- "Adding text to a document" on page 280
- "Importing tabular data documents" on page 281
- "Copying and pasting text from MS Office documents" on page 281

About formatting text

Formatting text in Dreamweaver is similar to using a standard word processor. You can set default formatting styles (Paragraph, Heading 1, Heading 2, and so on) for a block of text, change the font, size, color, and alignment of selected text, or apply text styles such as bold, italic, code (monospace), and underline.

By default, Dreamweaver formats text using Cascading Style Sheets (CSS) styles. CSS styles give web designers and developers greater control over web page design, while providing improved features for accessibility and reduced file size. As you style and align text using Dreamweaver formatting commands, CSS rules are embedded in the current document. This lets you more easily re-use existing styles that you create, as well as allowing you to name the styles you use. CSS is becoming the preferred method by which to format text and lay out web pages.

If you prefer, you can use HTML markup tags to format and align text in your web pages. If you need to use HTML tags instead of CSS styles, you must change the Dreamweaver default text formatting preferences. (For more information, see "Specifying HTML instead of CSS" on page 258.)

Using CSS styles is a way to control the style of a web page without compromising its structure. By separating visual design elements (fonts, colors, margins, and so on) from the structural logic of a web page, CSS style sheets give web designers both visual and typographic control without sacrificing the integrity of the content. In addition, defining typographic design and page layout from within a single, distinct block of code—without having to resort to image maps, <font> tags, tables, and spacer GIFs—allows for faster downloads, streamlined site maintenance, and a central point from which to control design attributes across multiple web pages.

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`).

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 Style pop-up menu 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.

Note: You can combine CSS styles and HTML 3.2 formatting within the same page. Formatting is applied in a hierarchical manner: HTML 3.2 formatting overrides formatting applied by external CSS style sheets, and CSS styles embedded in a document override external CSS styles. See "Using Cascading Style Sheets styles" on page 290.

Related topics

- "Inserting text" on page 280
- "Formatting text" on page 285

Understanding Cascading Style Sheets

Cascading Style Sheets (CSS) are a collection of formatting rules that 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.

You can define the following types of CSS styles in Dreamweaver:

- Custom CSS styles, also called class styles, let you set style attributes to any range or block of text (see "Applying a Class style" on page 295).
- 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 tags can reside in the following locations:

External CSS style sheets are collections of CSS styles stored in a separate, external file .css file (not an HTML file). This file is linked to a one or more page pages in a website using a link in the `head` section of a document.

Internal (or Embedded) CSS style sheets are collections of CSS styles included in a `style` tag in the `head` portion of an HTML document.

Inline styles are defined with specific instances of tags throughout an HTML 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, select Help > Reference and select O'Reilly CSS Reference from the pop-up menu in the Reference panel.

Manual HTML formatting overrides formatting applied with CSS styles. For CSS styles to control the formatting of a paragraph, you must remove all manual HTML formatting.

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, Netscape Navigator, Opera, and Apple Safari, and some are not currently supported by any browser.

About conflicting CSS 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 as the text color.
- 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). Thus, if a text element has both an external style sheet and an inline CSS style affecting it, the inline style is the one that is applied.
- 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>
```

Shorthand CSS properties

The CSS specification allows for the creation of styles using an abbreviated syntax known as *shorthand CSS*. Shorthand CSS lets you specify the values of several properties using a single property tag. For example, the `font` property lets you set `font-style`, `font-variant`, `font-weight`, `font-size`, `line-height`, and `font-family` properties within a single line of syntax.

A key issue to note when using shorthand CSS, is that values omitted from a shorthand CSS property are assigned their default value. This may cause pages to be incorrectly displayed when two or more CSS styles are assigned to the same tag.

For example, the `H1` tag shown below uses longhand CSS syntax. Note that the `font-variant`, `font-stretch`, `font-size-adjust`, and `font-style` properties have been assigned their default values.

```
H1 {
  font-weight: bold;
  font-size: 16pt;
  line-height: 18pt;
  font-family: Arial;
  font-variant: normal;
  font-style: normal;
  font-stretch: normal;
  font-size-adjust: none
}
```

Rewritten as a single, shorthand property, the same tag appears as follows:

```
H1 { font: bold 16pt/18pt Arial }
```

When written using shorthand notation, omitted values are automatically assigned their default values. Thus, the shorthand example above omits the `font-variant`, `font-stretch`, `font-size-adjust`, and `font-style` tags.

If you have styles defined in more than one location (for example, both embedded in an HTML page and imported from an external style sheet) using both the short and long forms of CSS syntax, be aware that omitted properties may override (or *cascade*) properties that are explicitly set in another.

For this reason, Dreamweaver uses the long form of CSS notation by default. This prevents possible issues that can arise when a CSS style written using shorthand notation, overrides one written in the longhand form. If you open a web page that was coded using shorthand CSS notation in Dreamweaver, be aware that Dreamweaver will create any new CSS styles using the longhand form. You can specify how Dreamweaver creates and edits CSS styles by changing the CSS editing preferences in the CSS Styles category of the Preferences dialog box (Edit > Preferences in Windows; Dreamweaver > Preferences on Macintosh).

Note: The Relevant CSS and CSS Properties tabs only create CSS styles using longhand notation. If you create a page or CSS style sheet using either the Relevant CSS or CSS Properties tabs, be aware that hand coding shorthand CSS styles may result in the shorthand properties overriding those created using the longhand form. For this reason, use longhand CSS notation to create your CSS styles.

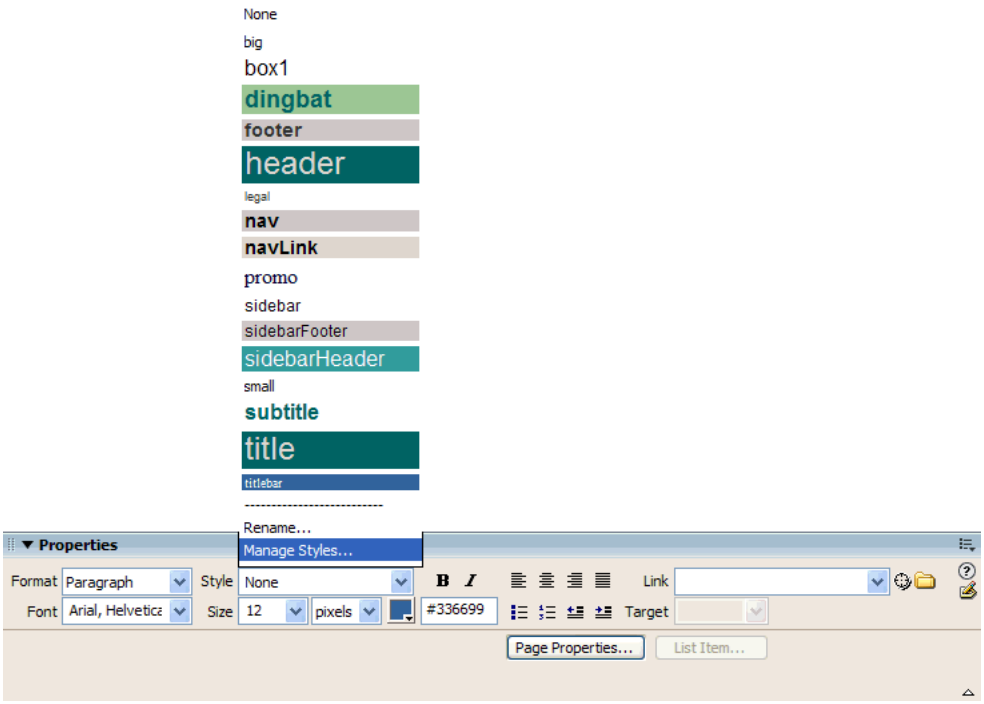
Related topics

- "About conflicting CSS styles" on page 276
- "The Relevant CSS tab" on page 278
- "The CSS Properties tab" on page 279

The Property inspector and text formatting

The text Property inspector lets you style and format the currently selected text. As you style and format text using the Property inspector, Dreamweaver keeps track of the formatting properties you assign to each text element, and assigns each a label using the naming convention: Style1, Style2, Style3, Style*n*. If you assign the same formatting attributes to two or more text elements, Dreamweaver labels those elements with the same title, eliminating redundant style names. The label Dreamweaver applies to a given body of text can then be applied using the Style pop-up menu, letting you build a library of styles within a page, and apply those same styles by simply selecting the text element in the page and selecting a style from the Styles pop-up menu. You can rename styles with more meaningful labels, such as Heading1, Heading2, Body, and TableBody.

The Style pop-up menu displays both the names of styles in your page, as well as a preview of the style's properties. The properties shown in the preview are font family, font size, font weight, text color, and background color.



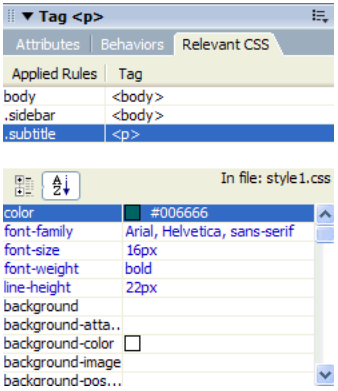
Related topics

- "Formatting paragraphs" on page 283
- "Setting and changing fonts and styles" on page 286
- "Renaming a style" on page 287
- "Using Cascading Style Sheets styles" on page 290

The Relevant CSS tab

The Relevant CSS tab lets you track the CSS rules affecting the currently selected text element, as well as modifying the rule's properties. Located in the Tag inspector (Window > Tag Inspector), the Relevant CSS tab consists of two sections. The upper portion of the Relevant CSS tab shows the rules affecting the current selection and the tags that they affect. The lower portion of the Relevant CSS tab displays the properties of the currently selected element in an editable grid. The properties are arranged alphabetically, with the set properties sorted to the top of the tab. Set properties are displayed in blue; properties irrelevant to the selection are displayed with a red strike through line. Hovering over a rule that is irrelevant displays a message explaining why the property is irrelevant. Typically a property is irrelevant because it's overridden or not an inherited property.

You can also use the lower panel of the Relevant CSS tab to modify the properties of a selection. The editable grid lets you change the values for any of the displayed properties. Any changes you make to a selection are applied immediately, letting you preview your work as you go.



Related topics

- "Using the CSS Styles panel" on page 291
- "Using the Relevant CSS tab" on page 292
- "Creating a new CSS style" on page 294
- "Changing attributes with the Tag inspector" on page 442

The CSS Properties tab

The CSS Properties tab displays the properties of the currently selected rule in the CSS Styles panel (Window > CSS Styles), Code view (View > Code), or by double-clicking a rule or property in the Relevant CSS tab (Window > Tag Inspector).

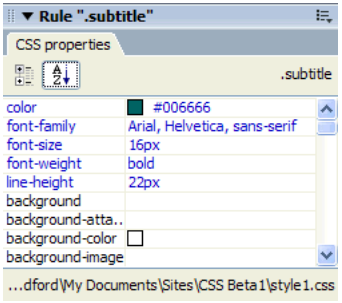
Note: The CSS Properties tab is only displayed when you select a rule in the CSS Styles panel, when you select a rule in Code view, or when you double-click a rule or property in the Relevant CSS tab. To return to the Relevant CSS tab, select a text element in an active HTML page.

The CSS Properties tab lets you easily see the a rule's property settings, as well as providing a quick way to modify any CSS style either embedded in the current document, or linked via an attached style sheet. The editable grid lets you change the values for any of the displayed properties. Any changes you make to a selection are applied immediately letting you preview your work as you go.

You can choose to display the properties either as an alphabetically arranged list, or as property categories by clicking the Show Category or Show List View buttons located in the upper left corner of the CSS Properties tab.

Category View divides the Dreamweaver supported CSS properties into eight categories: font, background, block, border, box, list, positioning, and extensions. Each category's properties are contained in a collapsible list that you expand or collapse by clicking the plus button next to its name. Properties within each category are sorted alphabetically, with set properties sorted to the top of the list displayed in blue text. Category view is the default view.

List View displays all of the Dreamweaver supported CSS properties in descending alphabetic order, with set properties sorted to the top of the list displayed in blue text (see the following illustration).



Related topics:

- "The Relevant CSS tab" on page 278
- "Using the CSS Styles panel" on page 291
- "Using the Relevant CSS tab" on page 292
- "Using the CSS Properties tab" on page 293
- "Changing attributes with the Tag inspector" on page 442

Inserting text

Dreamweaver enables you to easily insert text in a document by typing directly, copying and pasting, or importing. You can also insert extra space between characters and lines in your text.

Adding text to a document

To add text to a Dreamweaver document you can type text directly in the Dreamweaver document window or you can cut and paste. You can also import text from other documents (see "Importing tabular data documents" on page 281).

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 select Edit > Paste.

  Dreamweaver doesn't preserve text formatting applied in the other application, but it does preserve line breaks.

Importing tabular data documents

You can import tabular data into your document by first saving the files (such as Microsoft Excel files or database files) as delimited text files.

For information on importing and formatting table data, see "Importing and exporting tabular data" on page 166. For information about importing text from Microsoft Word HTML documents, see "Opening existing documents" on page 73.

You can also add text from MS Excel documents to a Dreamweaver document by copying and pasting the contents of the Excel file into a web page (see "Copying and pasting text from MS Office documents" on page 281).

To import tabular data:

1. Select File > Import > Import Tabular Data, or Insert > Table Objects > Import Tabular Data.

   The Import Table dialog box appears.
2. Browse for the file you want or enter its name in the text box.
3. 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.
4. Use the remaining options to format or define the table into which the data will be imported.
5. Click OK when you're done.

Copying and pasting text from MS Office documents

You can add the contents of a Word or Excel document to a new or existing web page.

Note: The file size, after Dreamweaver converts it to HTML, must be less than 300K.

When you add content to a page, Dreamweaver converts the content to HTML and copies it to your web page. You can edit the content in Dreamweaver; changes you make to the original file on your computer will not appear on your website.

Note: If you use Microsoft Office 97, you cannot add the contents of a Word or Excel document; you must insert a link to the document. For information, see "Inserting a link to a Word or Excel document" on page 282.

To add the contents of a Word or Excel document to a new or existing web page:

1. Open the web page that you want to copy the contents of the Word or Excel file to.
2. Do one of the following to select the file:
   - Drag the file from its current location to the page where you want the content to appear. If the Insert Microsoft Word or Excel Document dialog box appears, click Insert the contents of the document into this web page, then click OK.
   - Select File > Import > Word Document or File > Import > Excel Document. In the Open dialog box, browse to the file you want to add, then click Open.

   The contents of the Word or Excel document appears in your page.

Inserting a link to a Word or Excel document

You can insert a link to a Microsoft Word or Excel document in an existing page.

To create a link to a Word or Excel document:

1 Open the page where you want the link to appear.

2 Drag the file from its current location to your Dreamweaver page where you want the link to appear.

The Insert Microsoft Word or Excel Document dialog box appears.

3 Select Create a link, then click OK.

4 If the document you are creating a link to lives outside of your site's root folder, Dreamweaver prompts you to copy the document to the site root.

By copying the document to the site's root folder, you ensure that the document will be available when you publish the website.

5 A link to the Word or Excel document appears in the page.

The link text will be the name of the file you link to. To change link text after you create the link, see "Managing links" on page 327.

Formatting paragraphs and document structure

Dreamweaver supports all of the web standards used in page- and object-formatting. This section describes how to format paragraphs, as well as how to insert horizontal rules and dates.

Related topics

- "Setting page properties" on page 258
- "Aligning text" on page 283
- "Indenting text" on page 283
- "Adding paragraph spacing" on page 284
- "Using horizontal rules" on page 284
- "Inserting dates" on page 289

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.

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, select an option:
   - Select 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.
   - Select None to remove a paragraph format.

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, select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh); then in the General category, under Editing Options make sure Switch to Plain Paragraph after Heading is unchecked.

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 select Text > Align and select an alignment command.

To center elements:

1. Select the element (image, plug-in, table, or other page element) you want to center.
2. Select 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, select Text > Indent or Outdent, or select List > Indent or Outdent from the context menu.

Note: You can apply multiple indents to a paragraph. Each time you select this command, the text indents further from both sides of the document.

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 Text category of the Insert bar, select Character and click the Line Break icon.
- Select Insert > HTML > Special Characters > Line Break.

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:
   - Select Insert > HTML > Horizontal Rule.
   - In the HTML category of the Insert bar, click the Horizontal Rule button.

To modify a horizontal rule:

1. In the Document window, select the horizontal rule.
2. Select 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.

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
- Select 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 select 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 select 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.

Formatting text

You can apply text formatting to one letter or to entire site paragraphs and blocks of text.

Related topics

- "Modifying font combinations" on page 287
- "Changing the text color" on page 288
- "Inserting dates" on page 289
- "Inserting special characters" on page 289
- "Adding space between characters" on page 290
- "Creating a new CSS style" on page 294

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.

To set or change font characteristics:

1 Select the text. If no text is selected, the change applies to subsequent text you type.

2 Select from the following options:

- To change the font, select a font combination from the Property inspector or from the Text > Font submenu.

  Select 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 select a font style (Bold, Italic, Underline, and so on) from the Text > Style submenu.

  Note: When you use the Property inspector to apply bold or italic style, Dreamweaver automatically applies the `<strong>` or `<em>` 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).

- To change the font size, select a size (1 through 7) from the Property inspector or from the Text > Size submenu.

  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 "Using Cascading Style Sheets styles" on page 290.

- To increase or decrease the size of selected text, select 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 appear as 7. Dreamweaver does not display the `basefont` tag (which goes in the `head` section), although the font size should appear properly in a browser. To test this, compare text set at 3 and text set at +3.

Renaming a style

As you format text, Dreamweaver keeps track of the styles you create in each page, and builds a library of styles that you can reuse. This makes applying the same formatting to a block of text much simpler, as well as letting you create a more consistent look for your pages.

To rename a style:

1 Select Rename from the text Property inspector Style pop-up menu.

The Rename Style dialog box appears.

2 Select the style you want to rename from the Rename Style pop-up menu.

3 Enter a new name in the New Name text field.

4 Click OK.

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 Select 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 Do one of the following:

- 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 Select 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.

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 Do one of the following:

- Select a color from the palette by clicking the color picker in the Property inspector.
- Select 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 261).

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).

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:
   - Select Insert > Date.
   - In the Common category of the Insert bar, 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 Netscape Navigator and Internet Explorer) don't properly display many of the named entities.

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:
   - Select the name of the character from the Insert > HTML > Special Characters submenu.
   - In the Text category of the Insert bar, click the Characters button and select the character you want.

Tip: There are many other special characters available; to select one of them, select Insert › HTML › Special Characters › Other or select the HTML category of the Insert bar, click the Characters menu, and select Other Characters. Select a character from the Insert Other Character dialog box, and click OK.

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 insert a non-breaking space, do one of the following:

- In the HTML category of the Insert bar, click the Characters button and select Insert Non-breaking Space.
- Select Insert > HTML > Special Characters > Non-Breaking Space.
- Press Control+Shift+Spacebar (Windows) or Option+Spacebar (Macintosh).

To set a preference to add non-breaking spaces:

1. Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).
2. In the General category make sure Allow Multiple Consecutive Spaces is checked.

Using Cascading Style Sheets styles

By default, Dreamweaver uses Cascading Style Sheets (CSS) styles to format text. The styles that you apply to text using the Property inspector or menu commands create CSS rules which are embedded in the head of the current document. CSS styles provide you with greater flexibility and control over the appearance of your page, from the precise positioning of layout to specific fonts and text styles.

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 70.)

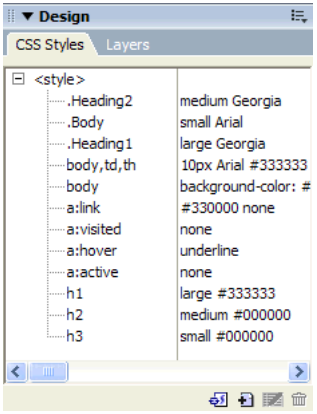
Using the CSS Styles panel

You use the CSS Styles panel to create, edit, and remove CSS styles, as well as to attach external style sheets to documents.

To open the CSS Styles panel:

- Do one of the following:
  - Select Window > CSS Styles.
  - Press Shift+F11.

  The CSS Styles panel lets you look at the style definition of the styles associated with the current document, as well as the hierarchy of the styles. The CSS Styles panel displays style definitions of custom (class) CSS styles, redefined HTML tags, and CSS selector styles.



  When you create or attach a CSS style sheet, the style's name and attributes appear in the CSS Styles panel. The CSS Styles panel lists all the selectors defined in all style tags and externally linked or imported style sheets. When you create a custom (class) style, it appears in the CSS Styles panel, the Style pop-up menu in the Property inspector, 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 "Exporting styles to create a CSS style sheet" on page 296.

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 294.

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 298.

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.

Related topics

- "Creating a new CSS style" on page 294
- "Applying a Class style" on page 295
- "Linking to or importing an external CSS style sheet" on page 296
- "Editing a CSS style" on page 298
- "Editing a CSS style sheet" on page 299

Using the Relevant CSS tab

Use the Relevant CSS tab to view and modify the properties of selected CSS elements in the current document. The Relevant CSS tab lets you see what rules are affecting the current selection, and what tags those rules relate to.

To open the Relevant CSS tab:

1 Select Window > Tag inspector.

The Tag inspector appears.

2 Click the Relevant CSS tab to display the properties for the current page.

3 Select a text element in the current page to display its properties.

The Relevant CSS tab displays the CSS rules for the current selection, the name and location of the file containing the CSS rules, and the rule's relevant properties.

To edit a CSS rule using the Relevant CSS tab, do one of the following:

- Double-click the rule in the Relevant CSS tab to both open the rule's code editing, and to display the CSS Property inspector.
- Edit the rule's properties using the controls provided in the lower half of the Relevant CSS tab window.

Related topics

- "The Relevant CSS tab" on page 278
- "Creating a new CSS style" on page 294
- "Editing a CSS style" on page 298

Using the CSS Properties tab

The CSS Properties tab lets you directly edit the properties and values of a CSS rule. By default, the CSS properties are organized into categories. You can modify the properties of each CSS rule by entering values in text fields and selecting values from pop-up menus associated with each property.

Note: While the CSS Properties tab displays the properties of read-only CSS style sheets, the properties contained in these style sheets remain uneditable. This allows you to view the properties of read-only or locked style sheets, while protecting the CSS values they define. Dreamweaver will try to check out any style sheet that is locked, and may prompt you to set your site preferences such that the file can be checked out.

Opening the CSS Properties tab:

1 Select Window > CSS Styles to display the CSS Styles panel.

2 Select Window > Tag inspector to display the Tag inspector panel.

3 Select a CSS style in one of the following locations:

- CSS Styles panel
- CSS style sheet opened in the document window
- HTML page in Code view with CSS embedded in the `head` of the page

4 When you select a CSS style in any of these locations, the CSS Properties tab is displayed in the Tag inspector panel, and the properties of that style are displayed.

You can select how you want to view the CSS properties by clicking either the Show Category View or Show List View button located in the upper left corner of the CSS Properties tab.

Category View divides the Dreamweaver supported CSS properties into eight categories: font, background, block, border, box, list, positioning, and extensions. Each category's properties are contained in a collapsible list that you expand or collapse by clicking the plus button next to its name. Properties within each category are sorted alphabetically, with set properties sorted to the top of the list displayed in blue text. Category view is the default view.

List View displays all of the Dreamweaver supported CSS properties in descending alphabetic order, with set properties sorted to the top of the list displayed in blue text.

Related topics

- "Using the Relevant CSS tab" on page 292
- "Editing a CSS style" on page 298

Creating a new CSS style

You can 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, select Manage Styles from the Style pop-up menu, and click New in the dialog box that is displayed.
- In the Relevant CSS tab (Select Window > Tag inspector, and click the Relevant CSS tab) right-click and select New Rule from the context menu.
- Select 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 text box, enter a name for the style.

  Note: 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 Tag, then in the Tag field, enter an HTML tag or select 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 text box, enter one or more HTML tags or select one from the pop-up menu. The selectors (known as pseudo-class 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, select New Style Sheet File.
- To embed the style in the current document, select This Document Only.

4 Click OK.

The Style Definition dialog box appears.

5 Select 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

Related topics

- "Using the CSS Styles panel" on page 291
- "Using the Relevant CSS tab" on page 292
- "Applying a Class style" on page 295
- "Editing a CSS style" on page 298

Applying a Class style

Class 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 class styles associated with the current document are displayed in the CSS Styles panel (with a period (.) prepended to their name) and in the Style pop-up menu 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 styles to the same text, the styles may conflict and produce unexpected results. For more information, see "About conflicting CSS styles" on page 276.

Tip: When previewing styles defined in an external CSS style sheet, be sure to save the style sheet to ensure that your changes are reflected when you preview the page in a browser.

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), right-click the name of the style you want to apply and select Apply from the context menu.
- In the text Property inspector, select the class style you want to apply from the Style pop-up menu.
- In the Document window, right-click (Windows) or Control-click (Macintosh) the selected text, and in the context menu, select 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:

1 Select the object or text you want to remove the style from.

2 Do one of the following:

- In the text Property inspector, select None from the Style pop-up menu.
- Right-click the applied rule you want to remove in the Relevant CSS tab and select Set Class > None from the context menu.

Related topics

- "The Relevant CSS tab" on page 278
- "Editing a CSS style" on page 298

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 Select File > Export > CSS Styles or select Text > CSS Styles > Export.

The Export Styles as CSS File dialog box appears.

2 Enter a name for your style sheet and click Save.

The style is saved as a CSS style sheet.

Related topics

- "Using the CSS Styles panel" on page 291
- "Editing a CSS style sheet" on page 299
- "Using Design-Time style sheets" on page 300

Linking to or importing an external CSS style sheet

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.

You can, of course, attach to your pages any style sheet that you create or copy into your site. In addition, Dreamweaver is shipped with prebuilt style sheets that can be automatically moved into your site and attached to your pages. For information about using the design style sheets that come with Dreamweaver, see "Creating a document based on a Dreamweaver design file" on page 70.

For information about applying a style, see "Applying a Class style" on page 295.

To link to or import an external CSS style sheet:

1 Open the CSS Styles panel by doing one of the following:
   - Select Window > CSS Styles.
   - Press Shift + F11.
2 In the CSS Styles panel, click the Attach Style Sheet button.

   The Attach External Style Sheet dialog box appears.
3 In the Attach External Style Sheet dialog box, do one of the following:
   - Click Browse to browse to an external CSS style sheet.
   - 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, select 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.
   - You cannot use a link tag to add a reference from one external style sheet to another. If you want to nest style sheets, you must use an import directive. Some but not all browsers also recognize the import directive within a page (rather than just within style sheets). There are subtle differences in how conflicting properties are resolved when overlapping rules exist within external style sheets that are linked versus imported to a page. If you want to import, rather than link to, an external style sheet, select Import.
5 Click the Preview button to verify that the style sheet applies the styles you want to the current page.

   If the styles applied are not what you expect them to be, click Cancel to remove the style sheet. The page will revert to its previous appearance.
6 Click OK.

   The name of the external CSS style sheet appears in the CSS Styles panel and custom (class) styles are preceded by the external style sheet identifier.

Related topics

- "Exporting styles to create a CSS style sheet" on page 296
- "Editing a CSS style sheet" on page 299

Using the sample Dreamweaver style sheets

Dreamweaver provides sample style sheets that you can apply to your pages or that you can use as starting points to develop your own styles.

To apply Dreamweaver style sheets:

1 Open the CSS Styles panel by doing one of the following:
- Select Window > CSS Styles.
- Press Shift+F11.

2 In the CSS Styles panel, click the Attach External Style Sheet button.

The Attach External Style Sheet dialog box appears.

3 In the Attach External Style Sheet dialog box, click Sample Style Sheets.

The Sample Style Sheets dialog box appears.

4 Select a style sheet from the list box.

As you select style sheets within the list box, the Preview pane displays the text and color formatting of the selected style sheet.

5 Click the Preview button to apply the style sheet and verify that it applies the styles you want to the current page.

If the styles applied are not what you expect them to be, select another style sheet from the list, and click Preview to see those styles.

6 By default, Dreamweaver saves the style sheet in a folder named CSS just below the root of the site you defined for your page. If that folder does not exist, Dreamweaver creates it. You can save the file to another location by clicking Browse and browsing to another folder.

7 When you find a style sheet whose formatting rules meet your design criteria, click OK.

Editing a CSS style

You can easily edit both internal and external styles you have applied to a document.

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 "Starting an external editor for media files" on page 359.

To edit a CSS style:

1 Select Window > Tag inspector.

2 In the Tag inspector, click the Relevant CSS tab to view the CSS styles in the current document.

3 Place the insertion point in the text whose CSS style you want to edit.

Each CSS style you select in the page displays in the CSS panel.

4 Click the style you want to edit to select it, then do one of the following:

- Double-click the selected style in the CSS Styles panel.
- Right-click (Windows) or Control-click (Macintosh), then select Edit Rule from the content menu.
- Click the Edit Style button located at the bottom of the CSS panel.
- Select the style in the CSS panel, and edit its properties in the CSS Property inspector.

The changes you make are immediately applied to the current document, allowing you to preview your changes as they are made. If you are editing styles stored in an external style sheet, be sure to save your changes to see your updates applied.

5 When you are finished editing the style, save your work.

If you modify CSS styles that are in a styles sheet used by more than one document, those changes will be reflected in those pages as well.

Related topics

- "Using the Relevant CSS tab" on page 292
- "Using the CSS Properties tab" on page 293

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 298), but sometimes you may 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, 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, select Edit Style Sheet from the context menu, then in the dialog box that appears, select the style sheet you want to edit and click Edit.

The styles for the selected CSS style sheet appear in the dialog box.

2 In the dialog box, select the style you want to edit, then click Edit.

The CSS Style Definition dialog box appears.

3 Modify the styles as desired, then click OK.

Related topics

- "Using the CSS Styles panel" on page 291
- "Editing a CSS style" on page 298

Updating CSS style sheets in a Contribute site

Contribute users can't make changes to a CSS style sheet. To change a style sheet for a Contribute site, use Dreamweaver.

The following are important factors to keep in mind when updating style sheets for a Contribute site:

- If you make changes to a style sheet while a Contribute user is editing a page that uses that style sheet, the user won't see the changes to the style sheet until they publish the page.
- If you delete a style from a style sheet, the style name is not deleted from pages that use that style sheet, but since the style no longer exists, it isn't displayed the way the Contribute user may expect. Thus, if a user tells you that nothing happens when they apply a particular style, the problem may be that the style has been deleted from the style sheet.

To edit a CSS style sheet in a Contribute site:

1. Edit the style sheet using the Dreamweaver style-sheet-editing tools. For more information, see "Using Cascading Style Sheets styles" on page 290.
2. Tell all of the Contribute users who are working on the site to publish pages that use that style sheet, then re-edit those pages to view the new style sheet.

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 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, and in the context menu select Design-time.
   - Select Text > CSS Styles > Design-time.
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 box.

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.

Related topics

- "Using the CSS Styles panel" on page 291
- "Exporting styles to create a CSS style sheet" on page 296
- "Linking to or importing an external CSS style sheet" on page 296
- "Editing a CSS style sheet" on page 299

Checking spelling

Use the Check Spelling command in the Text menu to check the spelling in the current1 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, select Edit > Preferences > General (Windows) or Dreamweaver > Preferences > General (Macintosh), 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 at www.macromedia.com/support/dreamweaver.

To check and correct spelling:

1 Select Text > Check Spelling or press Shift+F7.

When Dreamweaver encounters an unrecognized word the Check Spelling dialog box appears.

2 Select the appropriate option based on how you want the discrepancy handled.

Searching for and replacing text

You can use the Find and Replace command to search for text and for HTML tags and attributes in a document or a set of documents.

Note: To search for files in a site, use different commands: Locate in Local Site and Locate in Remote Site.

To search for text and HTML within documents:

1 Open the document to search in, or select documents or a folder in the Files panel.

2 Select Edit > Find and Replace.

The Find and Replace dialog box appears.

3 Specify which files to search in, then specify the kind of search you want to perform, and text or tags to search for. Optionally, specify replacement text as well. Then click one of the Find buttons or one of the Replace buttons.

For more information, click the Help button.

4 When you're done, click the Close button to close the dialog box.

To search again without displaying the Find and Replace dialog box:

- Press F3 (Windows) or Command+G (Macintosh).

CHAPTER 14
Inserting Images

In Macromedia Dreamweaver MX 2004, 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 start it to edit images while you work in Dreamweaver.

This chapter contains the following sections:

- "About images" on page 303
- "Inserting an image" on page 305
- "Resizing an image" on page 309
- "Cropping an image" on page 310
- "Optimizing an image using Fireworks" on page 310
- "Adjusting the brightness and contrast of an image" on page 311
- "Sharpening an image" on page 311
- "Creating a rollover image" on page 312
- "Using an external image editor" on page 313
- "Applying behaviors to images" on page 313

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 accessibility.

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, gray scale, and true-color images, and alpha channel support for transparency. PNG is the native file format of Macromedia Fireworks. 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 Dreamweaver.

Editing images in Dreamweaver

Dreamweaver provides basic image-editing features that let you modify images without having to use an external image-editing application such as Macromedia Fireworks. The Dreamweaver image-editing tools are designed to let you easily work with content designers responsible for creating image files for use on your website.

Note: You do not need to have Macromedia Fireworks installed on your computer to use the Dreamweaver image-editing features.

Dreamweaver has the following image-editing features:

Image resampling adds or subtracts pixels from a resized JPEG and GIF image files to match the appearance of the original image as closely as possible. Resampling an image reduces an images file size, resulting in improved download performance.

When you resize an image in Dreamweaver, you can resample it to accommodate its new dimensions. When a bitmap object is resampled, pixels are added to or removed from the image to make it larger or smaller. Resampling an image to a higher resolution typically causes little loss of quality. Resampling to a lower resolution, however, always causes data loss and usually a drop in quality.

Cropping lets you edit images by reducing the area of the image. Typically, you'll want to crop an image to place more emphasis on the subject of the image, and remove unwanted aspects surrounding the center of interest in the image.

Brightness/Contrast modifies the contrast or brightness of pixels in an image. This affects the highlights, shadows, and midtones of an image. You typically use Brightness/Contrast when correcting images that are too dark or too light.

Sharpening adjusts the focus of an image by increasing the contrast of edges found within the image. When you scan an image, or take a digital photo, the default action of most image capturing software is to soften the edges of objects in the image. This prevents extremely fine details from becoming lost in the pixels from which digital images are composed. However, to bring out the details in digital image files, it is often necessary to sharpen the image, thereby increasing edge contrast, and making the image appear sharper.

Note: Dreamweaver image-editing features apply only to JPEG and GIF image file formats. Other bitmap image file formats cannot be edited using these image-editing features.

Related topics

- “Resizing an image” on page 309
- “Cropping an image” on page 310
- “Adjusting the brightness and contrast of an image” on page 311
- “Sharpening an image” on page 311

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 insert images dynamically. Dynamic images are those images that change often. For example, advertising banner rotation systems need to randomly select a single banner from a list of potential banners, and then dynamically display the selected banner’s image when a page is requested. For more information, see “Making images dynamic” on page 548.

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).
- Select 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:

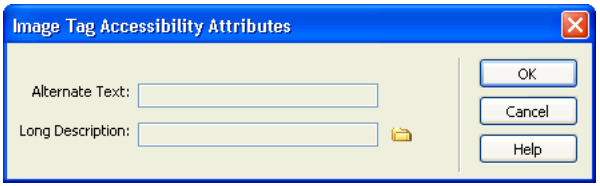
- Select File System to choose a graphic file.
- Select Data Source to choose a dynamic image source.

3 Browse to select 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. Click the Help button in the dialog box to learn about the dialog box options.

4 Click OK.

The Image Tag Accessibility Attributes dialog box appears if you have activated the dialog box in Preferences (see "Optimizing the workspace for accessible page design" on page 53).



5 Enter values in the Alternative Text and Long Description text boxes, then click OK.

Note: You can enter information in one or both text boxes depending on your needs.

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

The image appears in your document.

Note: If you click Cancel, the image appears in the document, but Dreamweaver does not associate accessibility tags or attributes with it.

6 In the Property inspector (Window > Properties), set properties for the image.

For more information, see "Setting image properties" in Dreamweaver Help.

To edit accessibility attributes for the image, see "Editing accessibility attributes for an image" on page 306.

Related topics

- "Setting a background image or background page color" on page 260
- "Using spacer images" on page 196

Editing accessibility attributes for an image

If you inserted accessibility attributes for an image (see "Inserting an image" on page 305), you can edit those values in the HTML code.

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 select Edit Tag Code.
- Edit the Alt value in the Property inspector.

Inserting an image placeholder

An image placeholder is a graphic you use until final artwork is ready to be added to a web page.

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 Common Insert bar, click the Image Placeholder icon.
- Select Insert > Image Placeholder.

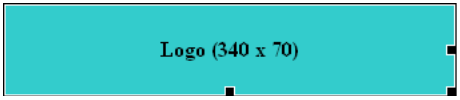
The Image Placeholder dialog box appears.

3 In the dialog box, select options for the image placeholder.

You can set the placeholder's size and color, as well as provide it with a text label. For more information, click the Help button in the dialog box.

4 Click OK.

The placeholder's color, size attributes, and label appear as follows:



When viewed in a browser, the label and size text do not appear.

Related topics

- "Resizing an image" on page 309
- "Using Fireworks to modify Dreamweaver image placeholders" on page 344.

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, 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, see "Using Fireworks to modify Dreamweaver image placeholders" on page 344.

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 text box.

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.

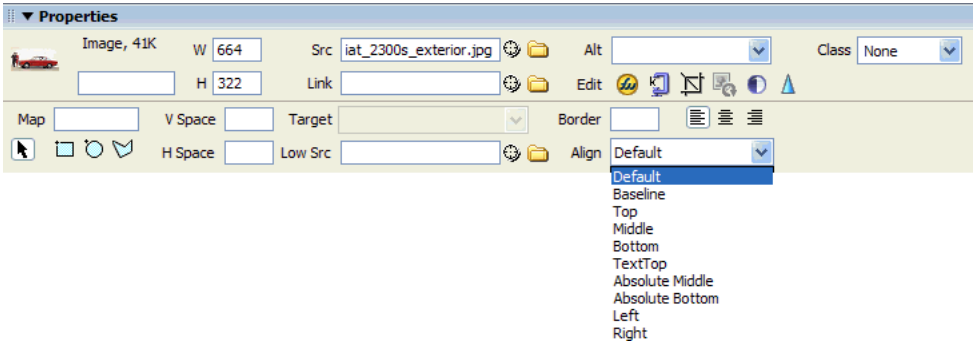
Aligning an image

You can align an image to text, another image, a plug-in, or other elements in the same line. You can also set the horizontal alignment of an image.

To align an image:

1 Select the image in Design view.

2 Set the alignment attributes of the image in the Property inspector.



You can set the alignment 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.

The following are the alignment options:

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.

Text Top 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 files, applets, and ActiveX controls in Dreamweaver.

Visually resizing an image in Dreamweaver helps you to see how the image affects the layout at different dimensions. Visually resizing an image does not scale the image file to the proportions that you specify. If you do visually resize an image in Dreamweaver, but don't use an image-editing application (such as Macromedia Fireworks) to scale the image file to the desired size, the user's browser has to scale the image when the page is loaded. This may cause both a delay in page download time and the improper display of the image in the user's browser. To reduce download time and to ensure that all instances of an image appear at the same size, use an image-editing application to scale images.

To visually resize an element:

1 Select the element (for example, an image or SWF file) 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 somewhere other than 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 text box, or click the Reset Size button.

To revert an image to its original size:

- Click the Reset size button in the image Property inspector.

To resample a resized image:

1 Resize the image as described above.

2 Click the Resample button in the image Property inspector.

Note: You can not resample image placeholders or elements other than bitmap images.

Related topics

- "Editing images in Dreamweaver" on page 304

Cropping an image

Dreamweaver lets you crop (or trim) bitmap file images.

Note: When you crop an image using Dreamweaver, the source image file is changed on disk. for this reason, you may want to keep a backup copy of the image file in the event you need to revert to the original image.

To crop an image file:

1 Open the page containing the image you want to crop, select the image, and do either of the following:

- Click the Crop Tool icon in the image Property inspector.
- Select Modify > Image > Crop.

Crop handles appear around the selected image.

2 Adjust the crop handles until the bounding box surrounds the area of the image that you want to keep.

3 Double-click inside the bounding box or press Enter to crop the selection.

A dialog box informs you that the image file you are cropping will be changed on disk. Click OK.

Every pixel in the selected bitmap outside the bounding box is removed, but other objects in the image remain.

4 Preview the image and ensure that it meets your expectations.

To undo the effects of the crop command:

- Select Edit > Undo Crop to revert to the original image.

  You can undo the effect of the Crop command (and revert to the original image file) up until the time that you quit Dreamweaver, or edit the file in an external image-editing application.

Related topics

- "Editing images in Dreamweaver" on page 304
- "Adjusting the brightness and contrast of an image" on page 311
- "Sharpening an image" on page 311

Optimizing an image using Fireworks

You can optimize images in your web pages from within Dreamweaver.

To optimize an image:

1 Open the page containing the image you want to optimize, select the image, and do either of the following:

- Click the Optimize in Fireworks button in the image Property inspector.
- Select Modify > Image > Optimize Image in Fireworks.

The Fireworks Optimize Image dialog box is displayed.

2 Click OK.

Related topics

- "Editing images in Dreamweaver" on page 304

- “Inserting an image” on page 305
- “Cropping an image” on page 310
- “Sharpening an image” on page 311

Adjusting the brightness and contrast of an image

Brightness/Contrast modifies the contrast or brightness of pixels in an image. This affects the highlights, shadows, and midtones of an image. You typically use Brightness/Contrast when correcting images that are too dark or too light.

To adjust the brightness and contrast of an image:

1 Open the page containing the image you want to adjust, select the image, and do either of the following:

- Click the Brightness/Contrast button in the image Property inspector
- Select Modify > Image > Brightness/Contrast

The Brightness/Contrast dialog box is displayed.

2 Drag the Brightness and Contrast sliders to adjust the settings.

Values range from -100 to 100.

3 Click OK.

Related topics

- “Editing images in Dreamweaver” on page 304
- “Inserting an image” on page 305
- “Cropping an image” on page 310
- “Sharpening an image” on page 311

Sharpening an image

Sharpening increases the contrast of pixels around the edges of objects to increase the image’s definition or sharpness.

To sharpen an image:

1 Open the page containing the image you want to sharpen, select the image, and do either of the following:

- Click the Sharpen button in the image Property inspector.
- Select Modify > Image > Sharpen.

The Sharpen dialog box is displayed.

2 You can specify the degree of sharpening Dreamweaver applies to the image by dragging the slider control, or entering a value between 0 and 10 in the text box.

As you adjust the sharpness of the image using the Sharpness dialog box, you can preview the change to the image.

3 Click OK.

4 Save your changes by selecting File > Save, or revert to the original image by selecting Edit > Undo Sharpen.

To undo the effects of the sharpen command:

- Select Edit > Undo Sharpen to revert to the original image.

  You can only undo the effect of the Sharpen command (and revert to the original image file) prior to saving the page containing the image. Once you save the page, the changes made to the image are permanently saved.

Related topics

- "Editing images in Dreamweaver" on page 304
- "Cropping an image" on page 310
- "Adjusting the brightness and contrast of an image" on page 311

Creating a rollover image

You can insert rollover images in your page. A *rollover* is an image that, when viewed in a browser, changes when the pointer moves across it.

Before you begin, obtain one or more pairs of images for the rollover. You create a rollover with two image files: 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.

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 396.

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.
   - Select Insert > Interactive Images > Rollover Image.

     The Insert Rollover Image dialog box appears.
3. Complete the dialog box.

   For more information, click the Help button in the dialog box.
4. Click OK.
5. Select File > Preview in Browser or press F12.

   You cannot see the effect of a rollover image in Design view.
6. In the browser, move the pointer over the original image.

   The display should switch to the rollover image.

Related topics

- "Inserting a navigation bar" on page 333

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 set up Fireworks as your primary external editor. For more information, see "Working with Fireworks" on page 342.

To start 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 select 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 start the primary image editor. If you haven't specified an image editor, Dreamweaver starts 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.

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.

Related topics

- "Specifying the editor to start from Dreamweaver" on page 360

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 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 387.)

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 396.)

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 397.)

You can also use behaviors to create more sophisticated navigation systems, such as a navigation bar or a jump menu. (See "Using navigation bars" on page 333 and "Inserting jump menus" on page 331.)

CHAPTER 15
Linking and Navigation

After you've set up a Macromedia Dreamweaver MX 2004 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 2004 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. For more information, see "Viewing a site map" on page 90.

There are several different ways of creating and managing links. Some web designers prefer to create links to nonexistent pages or files as they work, while others prefer to create all the files and pages first and then add 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 for broken, external, and orphaned links" on page 336.

This chapter contains the following sections:

- "Understanding document locations and paths" on page 316
- "Jump menus" on page 318
- "Navigation bars" on page 319
- "About image maps" on page 320
- "Creating links" on page 320
- "Managing links" on page 327
- "Inserting jump menus" on page 331
- "Using navigation bars" on page 333
- "Using image maps" on page 335
- "Attaching JavaScript behaviors to links" on page 336
- "Checking for broken, external, and orphaned links" on page 336
- "Fixing broken links" on page 338
- "Opening linked documents in Dreamweaver" on page 339

Understanding 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 website on naming and addressing at www.w3.org/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.

There are three types of link paths:

- Absolute paths (such as http://www.macromedia.com/support/dreamweaver/contents.html). For more information, see "Absolute paths" on page 316.
- Document-relative paths (such as dreamweaver/contents.html). For more information, see "Document-relative paths" on page 316.
- Site root-relative paths (such as /support/dreamweaver/contents.html). For more information, see "Site root-relative paths" on page 318.

Using Dreamweaver, you can easily select the type of document path to create for your links (see "Linking files and documents" on page 320).

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.

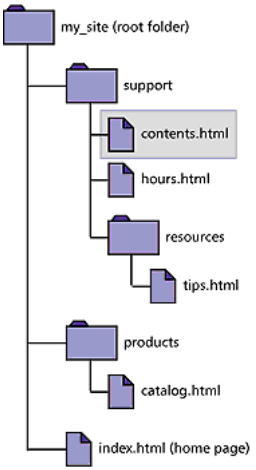
Note: When inserting images (not links): if you use an absolute path to an image that resides on a remote server and is not available on the local hard drive, you will not be able to view the image in your document window. Instead, you must preview the document in a browser to see it. If possible, use document or site root-relative paths for images. For more information, see "Inserting an image" on page 305.

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.

For example, suppose you have a site with the following structure:



You create links from contents.html to other files as follows:

- 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.

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 Files 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 Files panel, Dreamweaver updates all relevant links automatically.)

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:

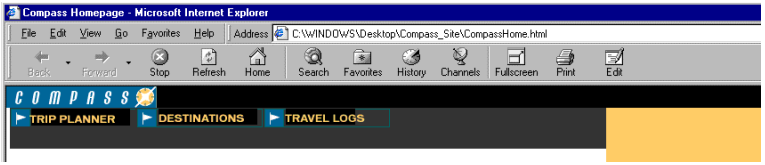
- (Optional) A menu selection prompt, such as a category description for the menu items, or instructions, such as "Choose one:"
- (Required) A list of linked menu items: a user selects an option and a linked document or file opens.
- (Optional) A Go button.

Related topics

- "Inserting jump menus" on page 331

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.



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.

Related topics

- "Using navigation bars" on page 333

About 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).

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 Netscape 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.

Related topics

- "Using image maps" on page 335

Creating links

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 320.)
- A named anchor link, which jumps to a specific location within a document. (See "Linking to a specific place in a document" on page 324.)
- An e-mail link, which creates a new blank e-mail message with the recipient's address already filled in. (See "Creating an e-mail link" on page 325.)
- 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 326.)

Note: Before creating links, make sure you understand how document-relative paths, site root-relative paths, and absolute paths work. (See "Understanding document locations and paths" on page 316.)

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 more information about using the site map to create links, see "Modifying links in the site map" on page 328.

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 new Dreamweaver site" on page 62). Dreamweaver uses this folder to determine the site root-relative paths to files.

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.

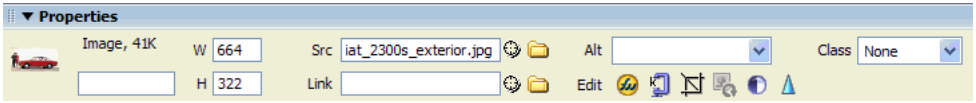
Linking to documents using the Property inspector

You can use the Property inspector's folder icon or Link text box to create links from an image, an object, or text to another document or file.

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 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 text box. 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 text box, 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.

For document-relative paths, omit the part of the absolute URL that is the same for both the current document and the linked document: If the file you're linking to is in the same folder as the current document, enter the filename; if it's in a subfolder, provide the name of the subfolder, then a forward slash (/), and then the filename; if it's in the parent folder, precede the filename with ../ (where ".." means "up one level in the folder hierarchy").

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 in 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 selecting Insert › Head Tag › Base and selecting the target information. For information about targeting frames, see "Controlling frame content with links" on page 213.

Linking documents using the Point-to-File icon

The Point-to-File icon lets you create links from an image, an object, or text to another document or file.

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 Files 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.

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 Files 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.

Linking documents using the site map

You can create links using the site map. The links you create are placed at the bottom of the selected HTML files, allowing you to quickly create links across a site.

To link documents using the site map and the Point-to-File icon:

1 Expand the Files panel, and then display both the Site Files and the Site Map views by holding down the site map icon and selecting 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.

To link documents in the site map, 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 Files 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 select Site > Link to Existing File (Windows) or Site > Site Map View > Link to Existing File (Macintosh), or select Link to Existing File from the context menu.
- Select an HTML page in the site map, and then select Site > Link to New File (Windows) or Site > Site Map View > Link to New File (Macintosh), or select Link to New File from the context menu.

Using the Hyperlink command

The Hyperlink command lets you create a text link to an image, an object, or to another document or file.

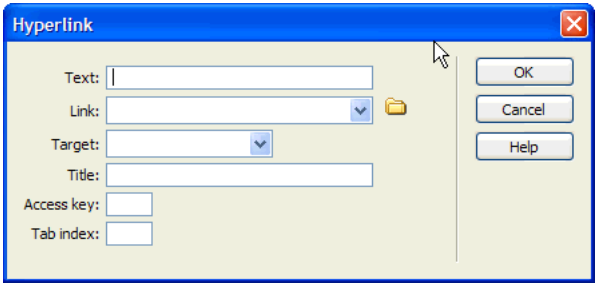
To add a hyperlink using the Hyperlink command:

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:

- Select Insert > Hyperlink.
- In the Common category of the Insert bar, 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.

Related topics

- "Modifying links in the site map" on page 328
- "Creating an e-mail link" on page 325
- "Creating null and script links" on page 326

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, you create a named anchor; then you create a link to the named anchor.

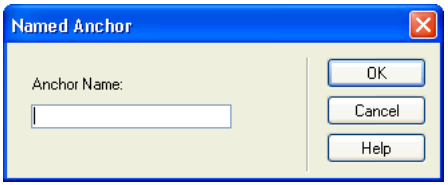
To create a named anchor:

1 In the Document window's Design view, place the insertion point where you want the named anchor.

2 Do one of the following:

- Select Insert > Named Anchor.
- Press Control+Alt+A (Windows) or Command+Option+A (Macintosh).
- In the Common category of the Insert bar, click the Named Anchor button.

The Named Anchor dialog box appears.



3 In the Anchor Name text box, type a name for the anchor, and click OK.

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, select 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: If you don't see the anchor, select View › Visual Aids › Invisible Elements to make it visible.

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.

Related topics

- "Linking files and documents" on page 320
- "Creating null and script links" on page 326

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 text box is automatically updated with the address specified in the e-mail link.

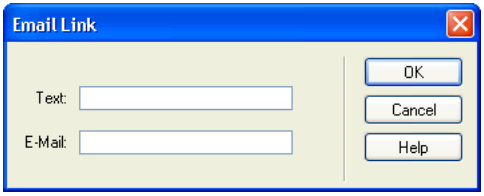
To create an e-mail link using the Insert E-mail Link command:

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:

- Select Insert > E-mail Link.
- In the Common category of the Insert bar, 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:jlydon@macromedia.com**.

Related topics

- "Linking to a specific place in a document" on page 324
- "Linking to a specific place in a document" on page 324

Creating null and script links

The most familiar kinds of links are those to documents and named anchors (see "Linking files and documents" on page 320 and "Linking to a specific place in a document" on page 324), 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 373.

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\"`).

Related topics

- "Linking to a specific place in a document" on page 324
- "Creating an e-mail link" on page 325

Managing links

To avoid broken links in your site, you can activate link management so that Dreamweaver updates links automatically when you make a change. You can also use a visual representation of your site to modify links or you can update all links to a particular file with one change.

Updating links automatically

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 make the updating process faster, Dreamweaver can create a cache file in which to store information about all the links in your local folder. The cache file updates invisibly as you add, change, or delete links to files on your local site.

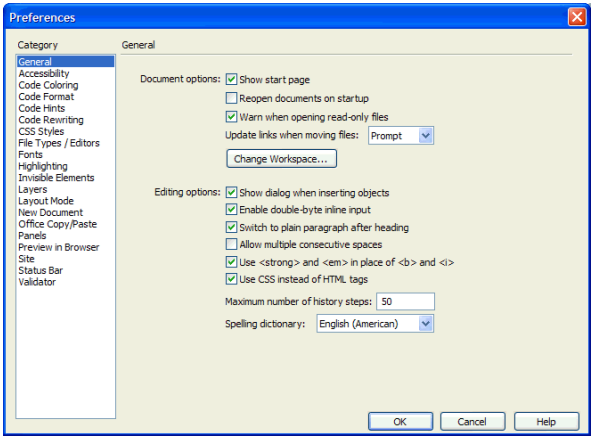
To turn on link management in Dreamweaver:

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

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 select 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 329.)

If you select 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 create a cache file for your site:

1 Select Site > Manage Sites.

The Edit Sites dialog box appears.

2 Select a site, and then click Edit.

The Site Definition dialog box appears.

3 Click the Advanced tab to display the Advanced category of the Site Definition dialog box.

4 Select Local Info from the category list on the left.

The Site Definition dialog box displays the Local Info options.

5 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 starting 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 recreate the cache for your site:

- In the Files panel, select Site > Advanced > Recreate Site Cache.

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 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:

- Select Site > Change Link (Windows) or Site > Site Map View > Change Link (Macintosh).

  Note: Use the Files panel's Site menu.

- Right-click (Windows) or Control-click (Macintosh), and select 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:

1 Select the page in the site map.

2 Do one of the following:

- Select Site > Remove Link (Windows) or Site > Site Map View > Remove Link (Macintosh).

  Note: Use the Files panel's Site menu.

- Right-click (Windows) or Control-click (Macintosh), and select 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:

1 Select a file in the site map.

2 Do one of the following:

- Select Site > Open Source of Link (Windows) or Site > Site Map View > Open Source of Link (Macintosh).
- Right-click (Windows) or Control-click (Macintosh), and select 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.

Related topics

- "Linking files and documents" on page 320
- "Working with a visual map of your site" on page 89

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.

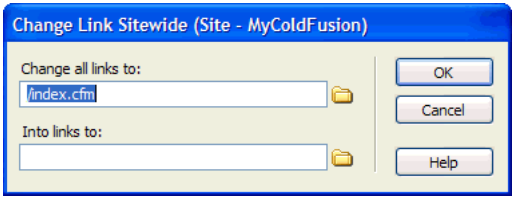
To change a link sitewide:

1 Select a file in the Local view of the Files panel.

Note: If you are changing an e-mail, FTP, null, or script link, you do not need to select a file.

2 Select 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.

Inserting jump menus

Jump menus let you associate URLs with options in a pop-up menu list. By choosing an item from the list, the user is redirected (or "jumps") to the specified URL. Jump menus are inserted within the Jump Menu form object.

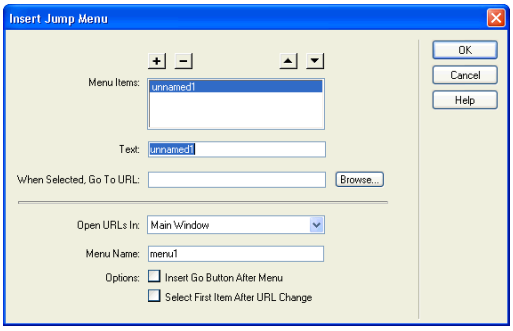
To insert a jump menu:

1 Open a document, and then place the insertion point in the Document window.

2 Do one of the following:

- Select Insert > Form > Jump Menu.
- In the Form category of the Insert bar, 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.

Related topics

- "Jump menus" on page 318
- "Troubleshooting jump menus" on page 332

Editing jump menu items

To edit jump menu items, 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 384).

To edit a jump menu item using the Property inspector:

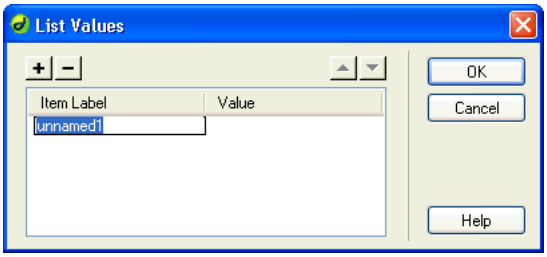
1 Select Window > Properties to open the Property inspector, if it isn't open.

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.

Related topics

- "Jump menus" on page 318
- "Inserting jump menus" on page 331

Troubleshooting jump menus

After a user selects a jump menu item, there is no way to reselect that menu item if the user navigates back to that page, or if the Open URL In text box 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.

Related topics

- "Jump menus" on page 318
- "Inserting jump menus" on page 331
- "Editing jump menu items" on page 332

Using navigation bars

A navigation bar consists of an image (or set of images) whose display changes based on the actions of a user.

Before using the Insert Navigation Bar command, you must 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.)

Once you create a navigation bar for a document, you can add or remove images them from the navigation bar using the Modify Navigation Bar command. 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.

Related topics

- "Navigation bars" on page 319

Inserting a navigation bar

When you insert a navigation bar, you name the navigation bar elements and select images to use for them.

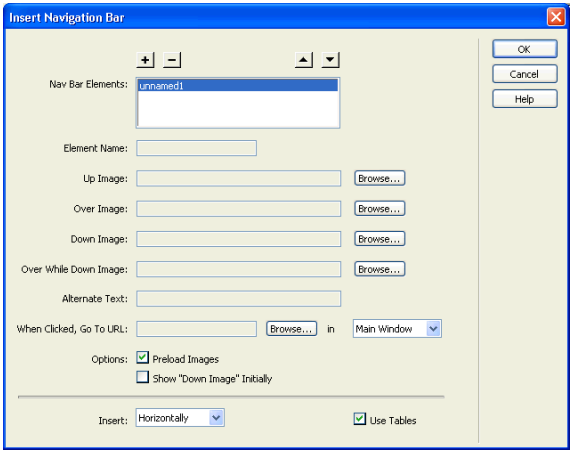
Tip: You can create a navigation bar, copy it to other pages in your site, use it with frames, and edit the page behaviors to show different states as pages are accessed.

To create a navigation bar:

1 Do one of the following:

- Select Insert > Image Objects > Navigation Bar.
- In the Common category of the Insert bar, click the Images menu and select 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.

Related topics

- “Navigation bars” on page 319

Modifying a navigation bar

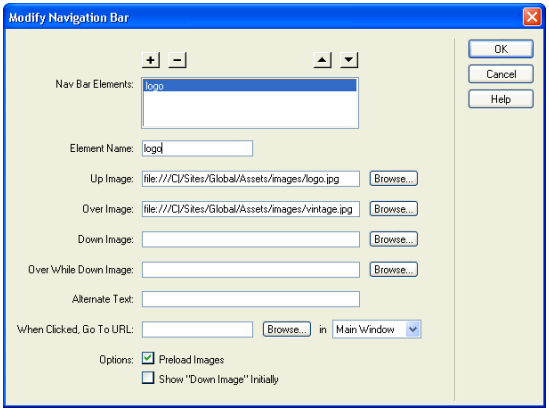
Once you create a navigation bar for a document, you can add images to or remove them from the navigation bar by using the Modify Navigation Bar command.

To modify a navigation bar:

1 Select the navigation bar in the active page.

2 Select Modify > Navigation Bar.

The Modify Navigation Bar dialog box appears.



3 In the Nav Bar Elements list, select the element you want to edit.

4 Make changes as necessary.

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

5 Click OK.

Related topics

- “Navigation bars” on page 319
- “Inserting a navigation bar” on page 333

Using 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).

Related topics

- "About image maps" on page 320

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 text box, enter 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.

Related topics

- "About image maps" on page 320

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.

Related topics

- "About image maps" on page 320
- "Inserting client-side image maps" on page 335

Attaching JavaScript behaviors to links

You can attach a behavior to any link in a document (see "Applying a behavior" on page 373). Consider using the following behaviors when you insert linked elements into documents:

Set Text of Status Bar determines the text of a message and displays 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 390.)

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 384.)

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 384.)

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 387.)

Checking for broken, external, and orphaned links

Use the Check Links feature to search for broken links and unreferenced—also known as orphaned—files (files that still exist in the site but no other files in the site link to) 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.

To check links within the current document:

1 Save the file to a location within your local Dreamweaver site.

2 Select File > Check Page > Check Links.

The Broken Links report appears in the Link Checker panel (in the Results panel group).

3 In the Link Checker panel, select External Links from the Show pop-up menu to view another report.

The External Links report appears in the Link Checker panel (in the Results panel group).

Tip: You can check for orphaned files when you check links across an entire site.

4 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 Files panel, select 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 select Check Links > Selected Files/Folders from the context menu.
- Select File > Check Page > Check Links.

The Broken Links report appears in the Link Checker panel (in the Results panel group).

4 In the Link Checker panel, select External Links from the Show pop-up menu to view another report.

The External Links report appears in the Link Checker panel (in the Results panel group).

Tip: 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 Files panel, select a site from the Current Sites pop-up menu.

2 Select Site > Check Links Sitewide.

The Broken Links report appears in the Link Checker panel (in the Results panel group).

3 In the Link Checker panel, select External Links or Orphaned Files from the Show pop-up menu to view another report.

A list of files that fit the report type you selected appears in the Link Checker panel.

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.

4 To save a report, click the Save Report button in the Link Checker panel.

Fixing broken links

After you run a links reports, 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 for broken, external, and orphaned links" on page 336).
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 "Checking in and checking out files" on page 97.

To fix links in the Property inspector:

1. Run a link check report (see "Checking for broken, external, and orphaned links" on page 336).
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, select 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 select 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.

Related topics

- "Previewing and testing pages in browsers" on page 270
- "Checking for broken, external, and orphaned links" on page 336
- "Fixing broken links" on page 338

CHAPTER 16
Working with Other Applications

Macromedia Fireworks MX 2004, Macromedia Flash MX 2004, and Macromedia Flash MX Professional 2004 are powerful web-development tools designed to create graphics and SWF movies for viewing in web pages. You can tightly integrate Macromedia Dreamweaver MX 2004 with these tools to simplify your web design workflow.

You can easily insert Fireworks images or tables and Flash content (SWF files) 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 it in a Dreamweaver document.

Note: To use Dreamweaver in conjunction with Fireworks and Flash, you must have all three applications installed on your computer.

This chapter contains the following sections:

- "About Fireworks and Flash integration" on page 341
- "Optimizing your work environment for Fireworks and Flash" on page 342
- "Working with Fireworks" on page 342
- "Working with Flash" on page 351

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).

Design Notes are small files that allow Dreamweaver to locate the appropriate source document for an exported image or movie file. 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). For information, see "Design Notes" on page 79.

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.

Optimizing your work environment for Fireworks and Flash

The key to developing a smoothly integrated workflow with Fireworks or Flash is to optimize your work environment.

To optimize your work environment for Fireworks and Flash:

1 Make sure Design Notes are enabled for your Dreamweaver site.

Design Notes are automatically enabled unless you alter your default site settings. For information about enabling Design Notes, see "Enabling and disabling Design Notes for a site" on page 108. For information on the role of Design Notes, see "About Fireworks and Flash integration" on page 341.

2 To easily start Fireworks from Dreamweaver, set Fireworks as the primary external image editor for Dreamweaver.

Set Fireworks as the primary editor for your graphic file types—GIF, PNG, and JPEG files. For more information, see "Using an external image editor" on page 313 or Setting external image editor preferences in Dreamweaver Help.

3 Save your Fireworks and Flash source and web-ready files in your defined Dreamweaver site folder.

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.

4 When exporting your Fireworks image 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 that Dreamweaver needs to work with Fireworks.

Working with Fireworks

Dreamweaver and Fireworks recognize and share many of the same file-editing procedures, 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.

Inserting a Fireworks image

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 (see "Using Fireworks to modify Dreamweaver image placeholders" on page 344).

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:

- Select 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.

Editing a Fireworks image or table from Dreamweaver

You can start Fireworks from Dreamweaver to edit images inserted in a Dreamweaver document. When you open and edit an image or an image slice that is part of a Fireworks table, Dreamweaver starts Fireworks, which opens the PNG file from which the image or table was exported.

When the image is part of a Fireworks table, you can open 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 start and edit a Fireworks image placed in Dreamweaver:

1 In Dreamweaver, select Window > Properties to open the Property inspector if it isn't already open.

2 Click the image or image slice to select it.

When you select an image that was exported from Fireworks, the Property inspector identifies the selection as a Fireworks image or table and displays the name of the PNG source file.

3 To start 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 select Edit With Fireworks from the context menu.

Fireworks starts 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.

Optimizing a Fireworks image from Dreamweaver

You can start 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 select Commands > Optimize Image in Fireworks.
2. If prompted, specify whether to start 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.

Using Fireworks to modify Dreamweaver image placeholders

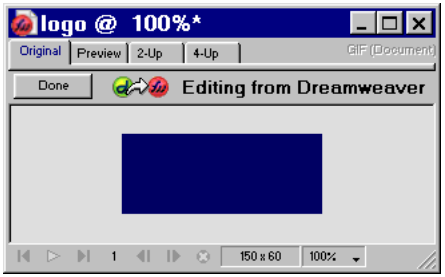
You can create a placeholder image in a Dreamweaver document and then start Fireworks to design a graphic image or Fireworks table to replace it. For information about inserting an image placeholder, see "Inserting an image placeholder" on page 307.

To create a new image from an image placeholder, you must have both Dreamweaver MX and Fireworks MX installed on your system.

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 start Fireworks for editing:
   - In the Property inspector, click Create.
   - Press Control (Windows) or Command (Macintosh) then double-click the image placeholder.
   - Right-click (Windows) or Control-click (Macintosh) the image placeholder, then select Create Image in Fireworks.

Fireworks starts, in Editing from Dreamweaver mode.



4 Use Fireworks options to design the image.

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 cut out a slice in Fireworks in the new image, it will delete the link in the Dreamweaver document when you replace the image placeholder.

The following 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.

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 text box, select 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 text box 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, JPEG, or, in the case of sliced images, as HTML and images.

8 In the Export dialog box, for Save In, select the Dreamweaver local site folder.

9 The Name text box 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 to Dreamweaver. In the Dreamweaver document, the exported file or Fireworks table replaces the image placeholder.

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.

To edit the background images in an image-based pop-up menu, select the image you want to update in Dreamweaver, then click Edit. For information about editing a Fireworks image, see "Editing a Fireworks image or table from Dreamweaver" on page 343.

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 edit the pop-up menu in Fireworks rather than in Dreamweaver.

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. The Show Pop-Up Menu behavior 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 the pop-up menu options, see "Show Pop-Up Menu" on page 393.
3. Make the changes you want to make to the pop-up menu.
4. When you finish modifying the pop-up menu, click OK.

Specifying launch-and-edit preferences for Fireworks source files

When using Fireworks to edit images, the images you place in your web pages are normally exported by Fireworks from a source PNG file. When you open ("launch") an image file in Dreamweaver to edit it, Fireworks automatically opens the source PNG file, prompting you to locate the PNG file if it cannot be found. If you prefer, you can set preferences in Fireworks to have Dreamweaver open the inserted image, or you can have Fireworks give you the option to use the inserted image file or the Fireworks source file every time you open an image in Dreamweaver.

Note: Dreamweaver recognizes the Fireworks launch-and-edit preferences only in certain cases. Specifically, you must be opening and optimizing an image that is not part of a Fireworks table and contains a correct Design Notes path to a source PNG file.

To specify launch-and-edit preferences for Fireworks:

1. In Fireworks, select Edit > Preferences or Fireworks > Preferences (Macintosh) and then click the Launch and Edit tab (Windows) or select 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 opens 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 opens 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 open 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 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 and 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:
   - Select 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 select 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, select 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 select 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, select 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, select HTML and Images.
4. In the HTML pop-up menu, select 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 select 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. Select File > Save.
3. In Fireworks, select 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

You can automatically generate a website that showcases a photo album of images located in a given folder. Dreamweaver uses Fireworks to create a thumbnail image 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 create a 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 have any of the following extensions: .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, select Commands > Create Web Photo Album.

2 In the Photo Album Title text box, 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 boxes.

3 Select the folder containing source images by clicking the Browse button next to the Source Images Folder text box. Then select (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 box.

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:

- Select 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 Select 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 Select 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 Select 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.

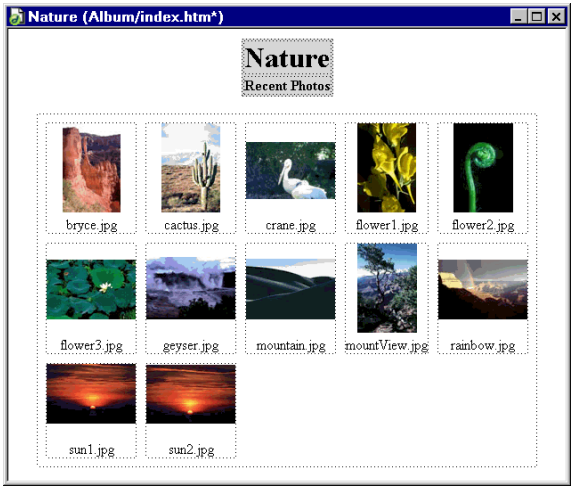
If you select this option, the thumbnails link to the navigation pages. If you don't select this option, the thumbnails link directly to the large-size images.

9 Click OK to create the HTML and image files for the web photo album.

Fireworks starts (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 box 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 Flash

You can use Dreamweaver to set playback and display options for a Flash file in a web page, or update the links in the movie. If Macromedia Flash MX 2004 is installed, you can also select a SWF file in a Dreamweaver document and start Flash to edit it.

Editing Flash content in Dreamweaver

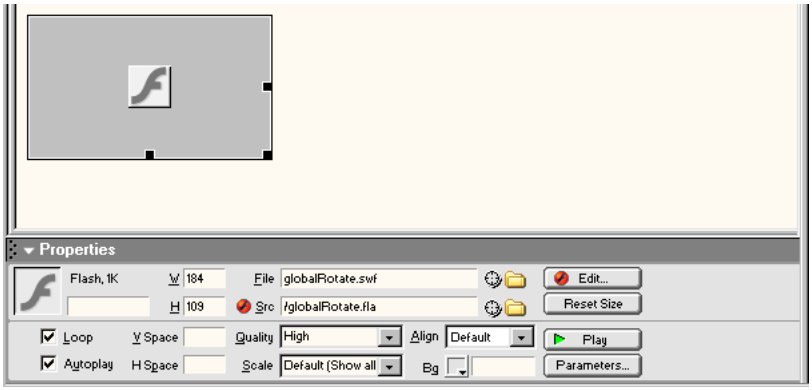
If Macromedia Flash MX is installed, you can select a SWF file in a Dreamweaver document and open Flash to edit it. Flash does not directly edit the SWF file; it edits the source document (FLA file) and re-exports the SWF file.

To open and edit Flash content inserted from Dreamweaver:

1 In Dreamweaver, select 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 SWF file placeholder to select it; then in the Property inspector click Edit.



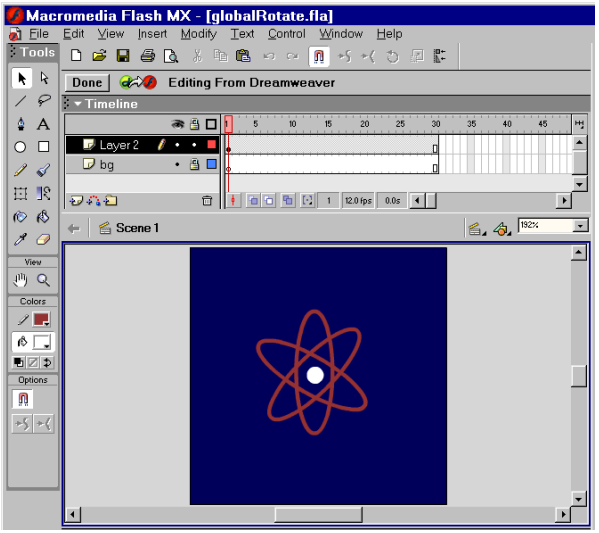
Note: If you do not have Flash MX, the Edit button is disabled.

- 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 select Edit With Flash from the context menu.

Dreamweaver starts Flash, and Flash attempts to locate the Flash authoring file (FLA) for the selected SWF file. 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 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 within Dreamweaver.



4 When you are finished making edits, click Done.

Flash updates the Flash authoring document (FLA file), re-exports the movie file (SWF file), closes, and then returns the focus to the Dreamweaver document.

5 To view the updated SWF in the document, click Play in the Property inspector or press F12 to preview your page in a browser window.

Updating links in a SWF file

You can use Dreamweaver to update a link in a Flash file (SWF file), then update the change in the Flash authoring document (FLA file).

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. 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 94. For information about Site Map view, see "Viewing a site map" on page 90.

2 Open Site Map view.

3 To show dependent files, do one of the following:

- Select View > Show Dependent Files.
- Select View > Layout to open the Site Definition dialog box, then select the Display Dependent Files option.

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 (Windows) or Control-click (Macintosh) the link, then select Change Link, then in the dialog box that appears in the URL text box, type the new URL path.
- To update all instances of the link, select Site > Change Link Sitewide, then in the dialog box that appears, in the Change All Links To text box, browse to or type the path of the link you are changing and in the Into Links To text box, browse to or type the path of the new URL.

5 Click OK.

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.

CHAPTER 17
Adding Audio, Video, and Interactive Elements

Macromedia Dreamweaver MX 2004 lets you add sound and movies to your website quickly and easily. You can attach Design Notes to these objects, which let you communicate with your team. You can also insert Macromedia Flash MX 2004 button and text objects from within Dreamweaver itself.

This chapter contains the following sections:

- "About media files" on page 356
- "Inserting and editing media objects" on page 358
- "Starting an external editor for media files" on page 359
- "Using Design Notes with media objects" on page 360
- "Inserting and modifying a Flash button object" on page 361
- "Inserting a Flash text object" on page 363
- "Inserting Flash content" on page 363
- "Downloading and installing Flash elements" on page 364
- "Inserting Flash elements" on page 364
- "Editing Flash element attributes" on page 365
- "Inserting Shockwave movies" on page 365
- "Adding video" on page 365
- "Adding sound to a page" on page 366
- "Inserting Netscape Navigator plug-in content" on page 367
- "Inserting an ActiveX control" on page 368
- "Inserting a Java applet" on page 368
- "Using behaviors to control media" on page 369

About media files

You can incorporate the following media files into your Dreamweaver pages: Flash and Shockwave movies, QuickTime, AVI, Java applets, Active X controls, and audio files of various formats.

Flash file types

Dreamweaver comes with Flash objects you can use whether you have Flash installed on your computer or not. If you have Flash, see "Working with Flash" on page 351 for information about using these applications in an integrated manner.

Before you use the Flash commands available in Dreamweaver, you should know about the 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 SWF 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 Flash text objects. For more information, see "Inserting and modifying a Flash button object" on page 361, "Inserting a Flash text object" on page 363, and "Inserting Flash content" on page 363.

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 (www.macromedia.com/exchange/dreamweaver/) 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 www.macromedia.com/go/flash_buttons.

The Flash element (.swc) is a Flash SWF file that lets you create Rich Internet applications by incorporating them in a web page. Flash elements have customizable parameters that you can modify to perform different application functions. For more information, see "Inserting Flash elements" on page 364 and "Editing Flash element attributes" on page 365.

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. MP3 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 an entire song could still take quite a while to download over a typical dial-up (telephone line) modem 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. Visitors must download and install the RealPlayer helper application or plug-in to play these files.

.qt, .qtm, .mov or QuickTime is both an audio and video format developed by Apple Computer. QuickTime is included with Apple Macintosh operating systems, and is used by most Macintosh applications that use audio, video, or animation. PCs can also play files in QuickTime format, but require a special QuickTime driver. QuickTime supports most encoding formats, including Cinepak, JPEG, and MPEG.

Note: In addition to the more common formats listed above, their are many different audio and video file formats available for use on the web. If you encounter a media file format that you are unfamiliar with, locate the creator of the format for information on how best to use and deploy it.

Inserting and editing media objects

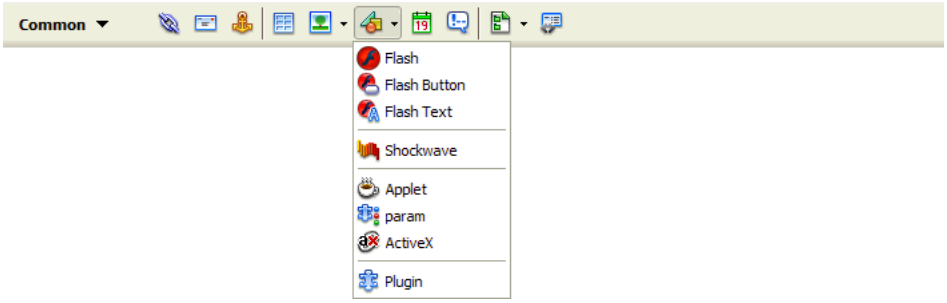
You can insert a Flash movie or object, QuickTime or Shockwave movie, Java applet, ActiveX control, or other audio or video objects in a Dreamweaver document.

To insert a media object in a page:

1 Place the insertion point in the Document window where you want to insert the object.

2 Insert the object by doing one of the following:

- In the Common category of the Insert bar, click the Media button and select the button for the type of object you want to insert.



- Select the appropriate object from the Insert > Media submenu.
- If the object you want to insert is not a Flash, Shockwave, Applet, or ActiveX object, use the Netscape Navigator plug-in button (the puzzle piece icon on the Insert bar) to insert it. For more information, see "Inserting Netscape Navigator plug-in content" on page 367.

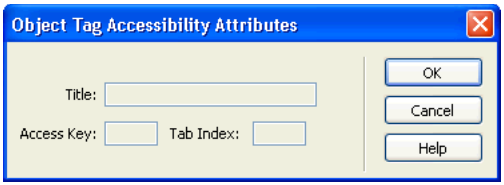
In most cases, a dialog box appears letting you select a source file and specify certain parameters for the media object.

Tip: To prevent such dialog boxes from appearing, select Edit › Preferences › General (Windows) or Dreamweaver› Preferences › General (Macintosh) 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 either click the Shockwave button in Media pop-up menu of the Common Insert bar, or select Insert › Media › Shockwave.)

3 Complete the Select File or Insert Flash dialog box.

4 Click OK.

The Object Tag Accessibility Attributes dialog box appears if you have activated the dialog box in Preferences (see "Optimizing the workspace for accessible page design" on page 53).



5 If the Object Tag Accessibility Attributes dialog box appears, complete the dialog box and click OK.

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

The media object appears in the document.

Note: If you click Cancel, a media object placeholder appears in the document, but Dreamweaver does not associate accessibility tags or attributes with it.

To specify a source file, or to set dimensions and other parameters and attributes, use the Property inspector for each object.

Editing accessibility attributes for media objects

If you inserted accessibility attributes with a media object (see "Inserting and editing media objects" on page 358), you can edit those values in the HTML code.

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 select Edit Tag Code.

Starting an external editor for media files

You can start an external editor from Dreamweaver to edit most media files. You can also specify the editor you want Dreamweaver to start to edit the file.

Starting an external editor from Dreamweaver

You can start an external editor from Dreamweaver to edit most media files.

To start an external editor from Dreamweaver:

1 Make sure the media file type is associated to an editor on your system.

To find out what editor is associated with the file type, select Edit > Preferences in Dreamweaver and select File Types/Editors from the Category list. Click the file's extension in the Extensions column to view the associated editor or editors in the Editors column. You can change the editor associated to a file type. For more information, see "Specifying the editor to start from Dreamweaver" on page 360.

2 Double-click the media file in the Site panel to open it in the external editor.

The editor that starts when you double-click the file in the Site panel is called the primary editor. If you double-click an image file, for example, Dreamweaver opens the file in the primary external image editor such as Macromedia Fireworks.

3 If you don't want to use the primary external editor to edit the file, you can use another editor on your system to edit the file by doing one of the following:

- In the Site panel, right-click (Windows) or Control-click (Macintosh) the filename and select Open With from the context menu.
- In Design view, right-click (Windows) or Control-click (Macintosh) the media element within the current page, and select Edit With from the context menu.

Specifying the editor to start from Dreamweaver

You can specify the editor you want Dreamweaver to use for editing a file type, and add or delete file types that Dreamweaver recognizes.

To explicitly specify which external editors should be started for a given file type:

1 Select 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.

2 Select the file type extension in the Extensions list and do one of the following:

- To associate a new editor with the file type, click the Plus (+) button above the Editors list and complete the dialog box that appears.

  For example, select the application icon for Acrobat to associate it with the file type.
- To make an editor the primary editor for a file type (that is, the editor that opens when you double-click the file type in the Site panel), select the editor in the Editors list and click Make Primary.
- To dissociate an editor from a file type, select the editor in the Editors list and click the Minus (-) button above the Editors list.

To add a new file type and associated editor:

1 Click the Plus (+) button above the Extensions list and 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 **.xml .xsl** if you wanted to associate them with an XML editor installed on your system.

2 Select an editor for the file type by clicking the Plus (+) button above the Editors list and completing the dialog box that appears.

To remove a file type:

- Select the file type in the Extensions list and click the Minus (-) button above the Extensions list.

  Note: You can't undo after removing a file type, so be sure that you want to remove it.

Using Design Notes with media objects

As with other objects in Dreamweaver, you can add Design Notes to a media object. Design Notes are notes associated with a particular file, that are 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 more information on working with Design Notes, see "Storing file information in Design Notes" on page 108.

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 for a site" on page 108).

2 Select 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.

Inserting and modifying a Flash button object

You can create, insert, and modify Flash buttons in your documents while working in Dreamweaver; Macromedia Flash is not required. The Flash button object is an updateable button based on a Flash template. You can customize a Flash button object, adding text, background color, and links to other files.

Creating and inserting a Flash button

You can create and insert Flash buttons in your documents while working in Design view or in Code view.

Note: You must save your document before inserting a Flash button or text object.

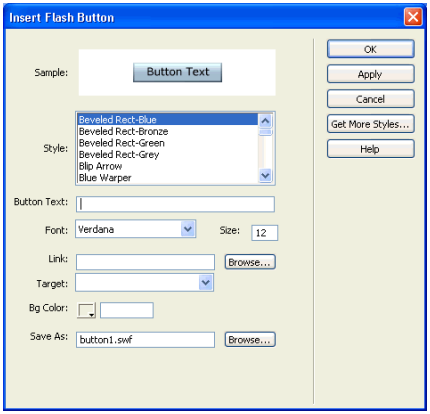
To insert a Flash button object:

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 Common category of the Insert bar, select Media and click the Flash Button icon.
- Select Insert > Media > Flash Button.

The Insert Flash Button dialog box appears.



3 Complete the Insert Flash Button dialog box, and click Apply or OK to insert the Flash button in the Document window.

Tip: To preview the button in Design view, click Apply. The dialog box remains open, and you can preview the button in your document.

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 background 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 select Edit from the context menu.
4. In the Insert Flash Button dialog box, make your edits in the text boxes described in "Inserting and modifying a Flash button object" on page 361.

   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 in the Property inspector (see "Resizing an image" on page 309).

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.

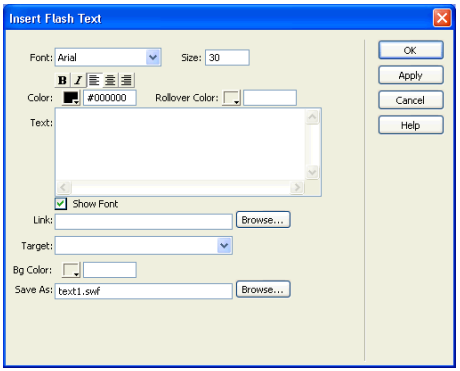
To insert a Flash text object:

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 Common category of the Insert bar, select Media and click the Flash Text icon.
- Select Insert > Media > Flash Text.

The Insert Flash Text dialog box appears.



3 Complete the Insert Flash Text dialog box, and 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 362).

Inserting Flash content

You can use Dreamweaver to insert Flash content in your pages.

To insert a SWF file (Flash content):

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 Common category of the Insert bar, select Media and click the Insert Flash icon.
- Select Insert > Media > Flash.

2 In the dialog box that appears, select a Flash 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.

Downloading and installing Flash elements

To use Flash elements in your web pages, you must first add the elements to Dreamweaver using the Extension Manager. The Extension Manager is a separate application that lets you install and manage extensions in Macromedia applications. Start the Extension Manager from Dreamweaver by selecting Commands > Manage Extensions.

To find the latest Flash elements for Dreamweaver, use the Macromedia Exchange website at www.macromedia.com/exchange/dreamweaver/. Once there, you can log in and download Flash elements and other Dreamweaver 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 new Flash elements or other Dreamweaver extensions.

For information about using the Extension Manager to install Flash elements (as well as other Dreamweaver extensions), see "Adding extensions to Dreamweaver" on page 57.

Inserting Flash elements

Using Dreamweaver you can insert Flash elements into your documents. Flash elements let you quickly and easily build Rich Internet Applications using prebuilt elements. For more information about Flash elements, and how they can be used in your web pages, see "Flash file types" on page 356.

To insert a Flash element:

1 In the Document window, place the insertion point where you want to insert a Flash element and do one of the following:

- In the Flash element category of the Insert bar, click the icon of the Flash element you want to insert.
- Select Insert > Flash element > *Flash element name*.

The Save Flash element dialog box appears.

2 Enter a filename for the Flash element and save it to an appropriate location within your site.

3 Click OK.

The Flash element placeholder appears in the document. You can modify the properties of the Flash element using the Tag and Property inspectors.

4 Select File > Preview in Browser to preview the Flash element.

Editing Flash element attributes

You edit Flash element attributes using the Property and Tag inspectors.

To edit Flash element attributes:

1 Do one of the following in the Document window:
- In Design view, select the Flash element.
- In Code view, click anywhere in a Flash component's name or in its contents.

2 If it is not already displayed, open the Tag inspector (Windows > Tag inspector).

3 Edit the Flash element attributes using the Tag inspector and the Property inspector.

4 Press Enter (Windows) or Return (Macintosh), or click elsewhere in the Tag inspector, to update the tag in your document.

For more information, click the Help button in the Tag inspector panel.

Inserting Shockwave movies

You can use Dreamweaver to insert Shockwave movies into your documents. 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.

To insert a Shockwave movie:

1 In the Document window, place the insertion point where you want to insert a Shockwave movie and do one of the following:
- In the Common category of the Insert bar, click the Media button and select the Shockwave icon from the menu.

- Select 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 text 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.

To include a short video clip in your page that the user can download:

1 Place the clip in your site folder.

These clips are often in the AVI or MPEG file format.

2 Link to the clip or embed it in your page.

To link to the clip, enter text for the link such as "Download Clip", select the text, and click the folder icon in the Property inspector. Browse to the video file and select it.

For information on how to include a streaming video in your page, see "Inserting Netscape Navigator plug-in content" on page 367.

Note: The user must download a helper application to view common streaming formats like Real Media, QuickTime, and Windows Media.

Adding sound to a page

You can add sound to a web page. There are several different types of sound files and formats, including .wav, .midi, and .mp3. For more information, see "Audio file formats" on page 357. 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 and then embed the SWF file to improve consistency.

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.

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 text box.

Embedding a sound file

Embedding audio incorporates the sound 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 to control the volume, the way the player looks on the page, or the beginning and ending points of the sound file.

Tip: When incorporating sound files in your web pages, carefully consider their appropriate use in your web site, and how visitors to your site use these media resources. Always provide a control to either turn off or mute the playing of the sound, in the event that visitors don't want to listen to the audio content.

To embed an audio file:

1 In Design view, place the insertion point where you want to embed the file and do one of the following:

- In the Common category of the Insert bar, click the Media button and select the plug-in icon.
- Select Insert > Media > Plugin.

For more information about the Plugin object, see "Inserting Netscape Navigator plug-in content" on page 367.

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 text box.

3 Enter the width and height by entering the values in the appropriate text boxes 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.

Inserting Netscape Navigator plug-in content

You can create content such as a QuickTime movie for a Netscape Navigator plug-in, and then use Dreamweaver to insert that content into an HTML document. Typical plug-ins include RealPlayer and QuickTime, while some content files include MP3s and QuickTime movies.

Inserting plug-in content in your page

Your page can include content that will run in a plug-in in the user's browser.

To insert Netscape 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 Common category of the Insert bar, select Media and then click the plug-in icon.
- Select Insert > Media > Plugin.

2 In the dialog box that appears, select a content file for a Netscape Navigator plug-in.

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

Playing plug-ins in the Document window

You can preview movies and animations that rely on Netscape Navigator plug-ins directly in the Design view of 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.

Note: You cannot preview movies or animations that rely on ActiveX controls.

To play movies in Design view, the proper plug-ins must be installed on your computer.

To play plug-in content in the Document window:

1 Insert one or more media elements by selecting Insert > Media > Shockwave, Insert > Media > Flash, or Insert > Media > Plugin.

2 Do one of the following:

- Select one of the media elements you have inserted, and select View > Plugins > Play or click the Play button in the Property inspector.
- Select 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 select View > Plugins >Stop, or click the Stop button in the Property inspector.

  You can also select View > Plugins > Stop All to stop all plug-in content from playing.

Troubleshooting Netscape 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. Some plug-ins require an additional 2 to 5 MB of memory to run.

Inserting an ActiveX control

You can insert an ActiveX control in your page. 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.

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 Common category of the Insert bar, click the drop down arrow on the Media button and select the ActiveX icon.
- In the Common category of the Insert bar, click the drop down arrow on the Media button and select the ActiveX icon. With the ActiveX icon displayed in the Insert bar, you can drag the icon to the Document window
- Select Insert > Media > ActiveX.

An icon marks where the ActiveX control will appear on the page in Internet Explorer.

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

Inserting a Java applet

You can insert a Java applet into an HTML document using Dreamweaver. Java is a programming language that allows the development of lightweight applications (*applets*) that can be embedded in web pages.

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 Common category of the Insert bar, click the drop down arrow on the Media button and select the Applet icon.
- Select Insert > Media > Applet.

2 In the dialog box that appears, select a file containing a Java applet.

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

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 380).

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 386).

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 379.

CHAPTER 18
Using JavaScript Behaviors

Macromedia Dreamweaver MX 2004 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 pre-written JavaScript code that performs a specific task, such as opening a browser window, showing or hiding a layer, playing a sound, or stopping a Macromedia Shockwave movie. The actions provided with 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 2004 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 376.) You can write your own behavior actions if you are proficient in JavaScript. For more information on writing behavior actions, see Extending Dreamweaver Help (Help > 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 the following sections:

- "Using the Behaviors panel" on page 372
- "About events" on page 372
- "Applying a behavior" on page 373
- "Attaching a behavior to text" on page 374
- "Changing a behavior" on page 375
- "Updating a behavior" on page 375
- "Creating new actions" on page 375
- "Downloading and installing third-party behaviors" on page 376
- "Using the behavior actions that come with Dreamweaver" on page 376

Using the Behaviors panel

You 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:

- Select 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, select 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 Behavior 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 www.macromedia.com/support/dreamweaver/.

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. 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. Events may be disabled (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 select one of them, Dreamweaver wraps an `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.

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 the 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 select 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 Netscape Navigator 4.0 or any 3.0 browser.

Note: You can't attach a behavior to plain text. For more information, see "Attaching a behavior to text" on page 374.

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 375.

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. Select Window > Behaviors to open the Behaviors panel.
3. Click the Plus (+) button and select 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 Control Shockwave or Flash action is dimmed if the document contains no Shockwave or Macromedia Flash SWF files. If no events are available for the selected object, all actions are dimmed.

   When you select 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 376.
5. The default event to trigger the action appears in the Events column. If this is not the trigger event you want, select 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.)

Attaching a behavior to 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, but 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 text box. Be sure to include both the colon and the semicolon.

   Note: You can instead use a number sign (#) in the Link text box 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. Select an action from the Actions pop-up menu, enter parameters for the action, and select an event to trigger the action. For detailed information, see "Applying a behavior" on page 373.

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 selecting 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 detailed information, see Chapter 13, "Inserting and Formatting Text," on page 273.

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 Select 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 Do one of the following:

- 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. Alternately, you can select the action and cut and paste it into the desired location among the other actions.
- 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 in this section), all other occurrences of that behavior in that page are also updated. Behaviors created in Dreamweaver 3 work without modification in Dreamweaver 4.

Note: As stated previously, when you update an occurrence of a behavior on a page, all other occurrences of that behavior in that page are also updated. However, you must update behaviors for each page in your website.

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 (www.macromedia.com/exchange/dreamweaver/).

To download and install new behaviors from the Macromedia Exchange website:

1 Open the Behaviors panel and select 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 detailed information, see "Adding extensions to Dreamweaver" on page 57.

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 Netscape 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 select 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, select 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 change are determined by the browser; many more properties can be changed by this behavior in Internet Explorer 4.0 than in Internet Explorer 3.0 or Netscape 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 select Change Property from the Actions pop-up menu.
3. From the Type of Object pop-up menu, select 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. Select an object from the Named Object pop-up menu.
5. Select a property from the Property pop-up menu, or enter the name of the property in the text box.

   To see the properties that can be changed in each browser, select 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 text box, 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, select 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 Netscape 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 `<a href="javascript:;">`) 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 select 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, select 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, select 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 Select an option from the Other Browsers pop-up menu to specify what to do if the browser is neither Netscape 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 Netscape Navigator and Internet Explorer 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 boxes 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, select 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 select 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 Internet Explorer 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 select Check Plugin from the Actions pop-up menu.

3 Select a plug-in from the Plugin pop-up menu, or click Enter and type the exact name of the plug-in in the adjacent text box.

You must use the exact name of the plug-in as specified in bold on the About Plug-ins page in Netscape Navigator. (In Windows, select Navigator's Help > About Plug-ins command; on the Macintosh, select About Plug-ins from the Apple menu.)

4 In the If Found, Go To URL text box, 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 text box, 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 text box. 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 deselected.

This option applies only to Internet Explorer; Netscape Navigator can always detect plug-ins.

7 Click OK.

8 Check that the default event is the one you want.

If it isn't, select 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 SWF files. To use the Control Shockwave or Flash action:

1. Select Insert > Media > Shockwave or Insert > Media > Flash to insert a Shockwave or Flash SWF file, respectively.
2. Select Window > Properties and enter a name for the movie in the upper left text box (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 SWF file. 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 select Control Shockwave or Flash from the Actions pop-up menu.

   A parameters dialog box appears.
6. Select a movie from the Movie pop-up menu.

   Dreamweaver automatically lists the names of all Shockwave and Flash SWF files 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. Select whether 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, select 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. Select 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 select 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 Internet Explorer 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 Select 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, select constrained movement.

7 For constrained movement, enter values (in pixels) in the Up, Down, Left, and Right text boxes.

Values are relative to the starting position of the layer. To constrain movement within a rectangular region, enter positive values in all four text boxes. 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 text boxes.

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 text boxes. Values are relative to the top left corner of the browser window. Click Get Current Position to automatically fill the text boxes with the current position of the layer.

9 Enter a value (in pixels) in the Snap if Within text box 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, select 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 Select 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 select 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 text box 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 box. For more information, see "Gathering information about the draggable layer" on page 382.

13 Enter JavaScript code or a function name (for example, **evaluateLayerPos**()) in the second Call JavaScript text box 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, select 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 Netscape 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 values of `MM_UPDOWN` or `MM_LEFTRIGHT` in a form field, you could use those values in a variety of other ways. For example, 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.

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 Select Edit > Find.

3 Select HTML Source from the Find What pop-up menu.

4 Type (**!curDrag**), including the parentheses, in the adjacent text box.

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.

To use the Go To URL action:

1 Select an object and open the Behaviors panel.

2 Click the Plus (+) button and select Go to URL from the Actions pop-up menu.

3 Select 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 text box.

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, select 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 "Jump menus" on page 318.

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. For more information, see "Inserting HTML form menus" on page 614.

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 and 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; selecting an item from a jump menu generally causes a URL to load without any need for further user action. But if the visitor selects 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-select 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 select Jump Menu Go from the Actions pop-up menu.
3. In the Choose Jump Menu pop-up menu, select 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 opened 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 select 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 text box.

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, select 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 predict reliably 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 select 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 text box.
4. Click OK.
5. Check that the default event is the one you want.

   If it isn't, select 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 detailed information, see "Open Browser Window" on page 384.

To use the Popup Message action:

1. Select an object and open the Behaviors panel.
2. Click the Plus (+) button and select Popup Message from the Actions pop-up menu.
3. Enter your message in the Message text box.
4. Click OK.
5. Check that the default event is the one you want.

   If it isn't, select 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 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 select 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 text box.
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 select 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, select 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 333.)

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, select an image state.
   - Select Down Image if you want to change the display of another image after a user has clicked the selected image.
   - Select 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.

   For information about image states, see "Using navigation bars" on page 333.
5. In the Also Set Image list, select another image on the page to set.
6. Click Browse to select the image file to be displayed, or type the path of the image file in the To Image File text box.
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 box, 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:

- Select Modify > Frameset > Split Frame Left, Right, Up, or Down.

  For more information, see "Creating frames and framesets" on page 204.

To use the Set Text of Frame action:

1. Select an object and open the Behaviors panel.
2. Click the Plus (+) button and select Set Text > Set Text of Frame from the Actions pop-up menu.
3. In the Set Text of Frame dialog box, select 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 text box and click OK.
6. Check that the default event is the one you want. If it isn't, select 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 text box 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. Select Insert > Layer.

   For more information, see "Inserting a layer" on page 150.
2. In the Property inspector, type a name for the layer.

To use the Set Text of Layer action:

1. Select an object and open the Behaviors panel.
2. Click the Plus (+) button and select 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 select the target layer.
4. Enter a message in the New HTML text box, then click OK.
5. Check that the default event is the one you want. If it isn't, select 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. 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 select 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 text box.

   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, select 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. Select Insert > Form Objects > Text Field.

   If Dreamweaver prompts you to add a form tag, click Yes. For more information, see "Creating Forms" on page 607.
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 select Set Text > Set Text of Text Field from the Actions pop-up menu.

3 In the Set Text of Text Field dialog box, select the target text field from the Text Field pop-up menu.

4 Enter text in the New Text text box, then click OK.

5 Check that the default event is the one you want. If it isn't, select 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 Select 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 select 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 Internet Explorer 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, select 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:

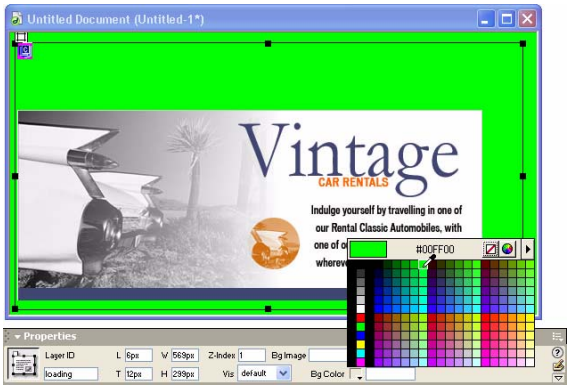
1 Click the Draw Layer button in the Common category of the Insert bar and draw a large layer in the Document window's Design view.

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 text box 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, select 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. Select `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, see Chapter 16, "Working with Other Applications," on page 341.

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 select 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 indention, 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 text box, select the default text (New Item), then enter 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 text box, 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 select 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, and 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, select 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 select a font that site visitors 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:

- Select one of the preset options.
- Set customized position coordinates by typing a number in the X text box to set the horizontal coordinate and 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 393.

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. Select 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 select 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, select 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 leave the Restore option selected while attaching Swap Image, you should never need to select the Swap Image Restore action manually.

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 Select Insert > Form or click the Form button in the Insert bar to insert a form.

2 Select 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 select 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 select Window > Behaviors.

4 Select 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 Select 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.

PART V
Working with Page Code

Use the advanced coding tools in Macromedia Dreamweaver MX 2004 to create or modify pages.

This part contains the following chapters:

Chapter 19, “Setting Up Your Coding Environment,” on page 401

Chapter 20, “Coding in Dreamweaver,” on page 415

Chapter 21, “Optimizing and Debugging Your Code,” on page 433

Chapter 22, “Editing Code in Design View,” on page 441

CHAPTER 19
Setting Up Your Coding Environment

You can adapt the coding environment in Macromedia Dreamweaver MX 2004 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 401
- "Using the coder-oriented workspace (Windows only)" on page 402
- "Setting viewing preferences" on page 403
- "Setting coding preferences" on page 403
- "Customizing keyboard shortcuts" on page 405
- "Opening files in Code view by default" on page 405
- "Setting Validator preferences" on page 405
- "Managing tag libraries" on page 406
- "Importing custom tags into Dreamweaver" on page 410
- "Using an external HTML editor with Dreamweaver" on page 412

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. The Code inspector works just like Code view; you can think of it as a detachable Code view for the current document.

This section contains instructions for changing the way you view your code.

To view code in the Document window:

- Select View > Code.

To code and visually edit a page in the Document window at the same time:

1 Select 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, select 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.

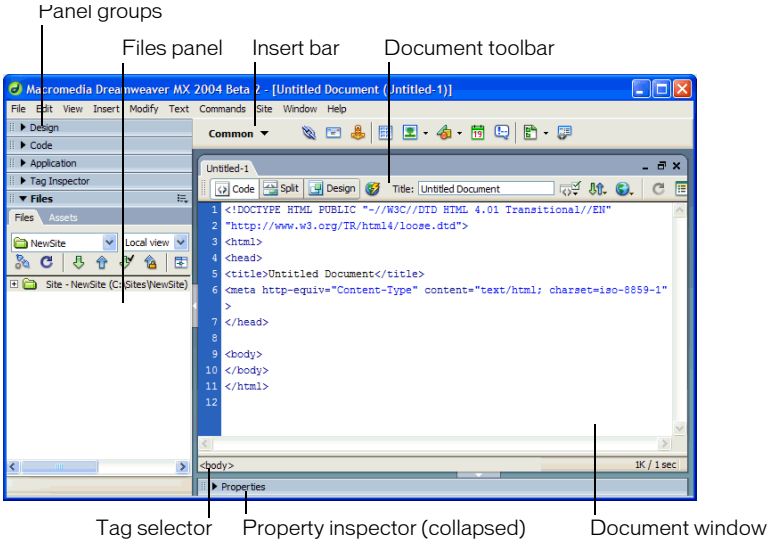
Code view is updated automatically when you make changes in Design view. However, after making changes in Code view, you must manually update the document in Design view by clicking in Design view or pressing F5.

To view code in a separate window:

- Select Window > Others > Code Inspector.

Using the coder-oriented workspace (Windows only)

In Windows, you can use a workspace that looks similar to the workspace in Macromedia HomeSite, with the panel groups docked on the left side of the main window instead of 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. For information on using this option, see "Choosing the workspace layout (Windows only)" on page 55.



Related topics

- "Opening files in Code view by default" on page 405.

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 View a document in Code view or the Code inspector.

2 Click the View Options menu in the toolbar at the top of Code view or the Code inspector.

A pop-up menu appears with a check mark beside each option that is already set.

3 To enable or disable options, select them from the menu.

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

Setting coding preferences

You can customize the Dreamweaver coding environment to meet your specific needs, by setting preferences for code formatting, rewriting, and coloring, among others.

Note: To set advanced preferences, use the Tag Library editor (see "Managing tag libraries" on page 406).

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 403.

To set code formatting preferences:

1 Select Edit > Preferences.

2 Select Code Format from the Category list on the left.

The Code Format preferences appear.

3 Adjust any of the settings in the dialog box.

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

Applying new formatting preferences to existing documents

The code formatting options that you specify in Code Format preferences are automatically applied only to new documents that you subsequently create with Dreamweaver. However, you can apply new formatting preferences to existing documents.

To apply new formatting preferences to an existing document:

1 Open the document in Dreamweaver.

2 Select Commands > Apply Source Formatting.

Setting code hints preferences

Code hints let you quickly insert tag names, attributes, and values as you type code. For more information, see "Using code hints" on page 423.

Tip: Even if code hints are disabled, you can display a pop-up hint in Code view by pressing Control+Spacebar.

To set code hints preferences:

1. Select Edit > Preferences.
2. Select Code Hints from the Category list on the left.

   The code hints preferences appear.
3. Adjust any of the settings in the dialog box.

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

Related topics

- "Using code hints" on page 423

Setting code rewriting preferences

When you open a document, Dreamweaver fixes (or *rewrites*) certain kinds of technically illegal code, depending on which code rewriting preferences you specify. 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.

To set code rewriting preferences:

1. Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).
2. Select Code Rewriting from the Category list on the left.

   The Code Rewriting preferences appear.
3. Adjust any of the settings in the dialog box.

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

Related topics

- "Importing a Microsoft Word HTML file" on page 74

Setting code coloring preferences

Use the code coloring preferences to specify colors for general categories of tags and code elements, such as form-related tags or JavaScript identifiers. To set color preferences for a specific tag, edit the tag definition in the Tag Library editor. For more information, see "Editing libraries, tags, and attributes" on page 408.

To set code coloring preferences:

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

2 Select Code Coloring from the Category list on the left.

The Code Coloring preferences appear.

3 Adjust any of the settings in the dialog box.

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

Related topics

- "Customizing code coloring preferences for a template" on page 228

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 "Customizing keyboard shortcuts" on page 57.

Related topics

"Working with code snippets" on page 424

Opening files in Code view by default

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 specify which file types open in Code view.

To set the default view for non-HTML files:

1 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

2 Select File Types/Editors from the Category list on the left.

The File Types/Editors preferences appear.

3 In the Open in Code View text box, add the filename extension of the file type you want to open automatically in Code view.

Type a space between filename extensions. You can add as many extensions 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 437). 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 Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh).

2 Select Validator from the Category list on the left.

The Validator preferences appear.

3 Select tag libraries to validate against, and set options for those libraries.

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

Managing tag libraries

A tag library, in Dreamweaver, is a collection of tags of a particular type, along with information about how Dreamweaver should format the tags. Tag libraries provide the information about tags that Dreamweaver uses for code hints, target browser checks, the Tag Chooser, and other coding capabilities. Using the Tag Library editor, you can add and delete tag libraries, tags, and attributes; set properties for a tag library; and edit tags and attributes.

See also "Importing custom tags into Dreamweaver" on page 410.

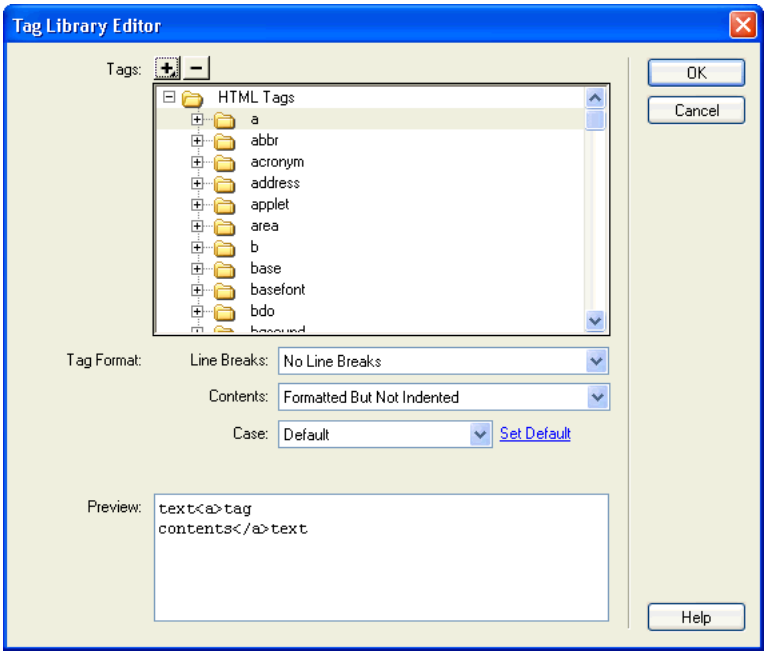
Opening and closing the Tag Library editor

To manage tag libraries, use the Tag Library editor.

To open the Tag Library editor:

- Select Edit > Tag Libraries.

The Tag Library editor appears. (The contents of this dialog box change depending on the selected tag.)



To close the Tag Library editor and save changes:

- Click OK.

To close the Tag Library editor without saving changes:

- Click Cancel.

  Note: When you click Cancel, all changes you've made in the Tag Library editor are discarded. If you deleted a tag or tag library, it's restored.

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 410.

To add a tag library:

1. In the Tag Library editor (Edit > Tag Libraries), click the Plus (+) button and select New Tag Library.

   The New Tag Library dialog box appears.
2. In the Library Name text box, type a name (for example, **Miscellaneous Tags**).
3. Click OK.

To add tags to a tag library:

1. In the Tag Library editor (Edit > Tag Libraries), click the Plus (+) button and select New Tags.

   The New Tags dialog box appears.
2. Adjust any of the settings in the dialog box.

   For more information, click the Help button in the dialog box.
3. 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 select New Attributes.

   The New Attributes dialog box appears.
2. Adjust any of the settings in the dialog box.

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

Related topics

- "Managing tag libraries" on page 406
- "Deleting libraries, tags, and attributes" on page 409

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 attribute 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 (not a tag) in the Tags list.

Note: The properties for tag libraries appear only when a tag library is selected. Tag libraries are represented by the top-level folders in the Tags list; for example, the HTML Tags folder represents a tag library, while the abbr folder within the HTML Tags folder represents a tag.

2 In the Used In list, select every document type that should use the tag library.

The document types you select here determine which document types provide code hints for the given tag library. For example, if the HTML option is not selected for a given tag library, code hints for that tag library don't appear in HTML files.

3 If the tags in the tag library need a prefix, then enter the prefix in the Tag Prefix text box.

Note: A prefix is used to identify a tag in the code as part of a particular tag library. Some tag libraries don't use prefixes.

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), expand a tag library in the Tags list 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, formatting, and indentation rules to the content.

Case specifies the case for a specific tag. Select from Default, Lowercase, Uppercase, or Mixed Case. If you select Mixed Case, the Tag Name Mixed Case dialog box appears. Type the tag with the case Dreamweaver should use when inserting it (for example, **getProperty**) and click OK.

Set Default sets the default case for all tags. In the Default Tag Case dialog box that appears, select <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), expand a tag library in the Tags box, expand a tag, and select a tag attribute.

2 In the Attribute Case pop-up menu, select Default, Lowercase, Uppercase, or Mixed Case.

If you select Mixed Case, the Attribute Name Mixed Case dialog box appears. Type the attribute with the case Dreamweaver should use when inserting it (for example, **onClick**) and click OK.

Click the Set Default link to set the default case for all attribute names.

3 In the Attribute Type pop-up menu, select the type of the attribute.

If you select Enumerated, type every allowed value for the attribute in the Values text box. Separate the values with commas, but no spaces. For example, the enumerated values of the `showborder` attribute of the `cfchart` tag are listed as `yes,no`.

Related topics

- "Managing tag libraries" on page 406
- "Adding libraries, tags, and attributes" on page 407

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 box.

2 Click the Minus (–) button.

3 If you are asked to confirm the deletion, click OK to permanently delete the item.

The item is removed from the Tags box.

4 Click OK to close the Tag Library editor and complete the deletion.

Related topics

- "Managing tag libraries" on page 406
- "Adding libraries, tags, and attributes" on page 407
- "Editing libraries, tags, and attributes" on page 408

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 an imported custom tag in Code view, a code hints menu appears, listing the tag's attributes and letting you select one.

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 select DTD Schema > Import XML DTD or Schema File.
3. Enter the filename or URL of the DTD or schema file.
4. Enter the prefix to be used with the tags.

   Note: A prefix is used to identify a tag in the code as part of a particular tag library. Some tag libraries don't use prefixes.

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 461). 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 do one of the following:
   - To import all the ASP.NET custom tags from the application server, select ASP.NET > Import All ASP.NET Custom Tags.
   - To import only certain custom tags from the application server, select 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 select JSP > Import From File (*.tld, *.jar, *.zip).
4. Enter a filename, a URI, and a prefix.
   For more information, click the Help button in the dialog box.
5. Click OK.

Importing JSP tags from a server (web.xml)

You can import a JSP tag library into Dreamweaver from a web.xml file on a JSP 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 select JSP > Import from Server (web.xml).
   The Import from Server dialog box appears.
4. Enter a filename and a URI.
   For more information, click the Help button in the dialog box.
5. 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 select JSP > Import JRun Server Tags from Folder.
4. Enter a folder name, a URI, and a prefix.
   For more information, click the Help button in the dialog box.
5. Click OK.

Using an external HTML editor with Dreamweaver

You can start 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, WordPad, TextPad, TextEdit, SimpleText, vi, or emacs.

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 applications track the current selection; for example, if you select text in Dreamweaver and switch to BBEdit, the same element is selected in BBEdit.

You can open other external editors (besides HomeSite or BBEdit) from Dreamweaver, but the document is not kept in sync in both applications as it is with HomeSite or BBEdit. When you are finished making changes in an external editor other than HomeSite or BBEdit, you must manually refresh the document in Dreamweaver.

Using HomeSite (Windows only)

You don't need to enable integration for HomeSite; it's integrated automatically when both applications are installed.

To use HomeSite:

1. Select 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 not to use BBEdit. Selections are not tracked between Dreamweaver and BBEdit if BBEdit integration is disabled. However, editing in Dreamweaver may be faster if BBEdit integration is disabled.

To use BBEdit with Dreamweaver:

1. Select 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. Select Edit > Preferences or Dreamweaver > Preferences (Mac OS X), and select File Types/Editors.
2. Deselect Enable BBEdit Integration and click OK.

Setting file type and external editor preferences

You can specify which external application to use for editing each of a variety of filename extensions.

To select an external HTML editor:

1. Select Edit > Preferences.
2. Select File Types/Editors from the Category list on the left.

   The File Types/Editors preferences appear.
3. Adjust any of the settings in the dialog box.

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

To start an external HTML editor:

- Select Edit > Edit with [editor name].

Related topics

- "Opening files in Code view by default" on page 405
- "Starting an external editor for media files" on page 359
- "Using an integrated HTML editor" on page 412

CHAPTER 20
Coding in Dreamweaver

Macromedia Dreamweaver MX 2004 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.

Note: For information on migrating from Macromedia HomeSite to Dreamweaver, visit the Dreamweaver Support Center at www.macromedia.com/go/migrate_from_homesite.

.This chapter contains the following topics:

- "About coding in Dreamweaver" on page 416
- "Writing and editing code" on page 423
- "Searching and replacing tags and attributes" on page 428
- "Making quick changes to a code selection" on page 430
- "Using language-reference material" on page 431
- "Printing your code" on page 432

Related topics

- "Setting Up Your Coding Environment" on page 401
- "Optimizing and Debugging Your Code" on page 433
- "Editing Code in Design View" on page 441

About coding in Dreamweaver

The coding environment in Dreamweaver allows you to write, edit, and test code (in a variety of languages) in your pages. Dreamweaver doesn't change your hand-written code unless you enable specific options to rewrite certain kinds of invalid code.

Related topics

- "Viewing your code" on page 401

Supported languages

In addition to text-editing capabilities, Dreamweaver provides various features, such as code hints, to help you code in certain languages. These languages include:

- HTML
- XHTML
- CSS
- JavaScript
- ColdFusion Markup Language (CFML)
- Visual Basic (for ASP and ASP.NET)
- C# (for ASP.NET)
- JSP
- PHP

Certain other languages, such as Perl, are not supported by the language-specific coding features in Dreamweaver; you can create and edit Perl files using Dreamweaver, but code hints (for example) don't apply to that language.

Related topics

- "Automatic code modification in Dreamweaver" on page 417
- "Using code hints" on page 423

Fixing invalid markup

If your document contains invalid code, Dreamweaver displays that code in Design view and optionally highlights it in Code view. If you select the code in either view, the Property inspector displays information about why it's invalid and how to fix it.

You can specify preferences in Dreamweaver for automatically rewriting various kinds of invalid code when you open a document.

Related topics

- "Setting coding preferences" on page 403
- "Setting viewing preferences" on page 403

Automatic code modification in Dreamweaver

You can set options that instruct Dreamweaver to automatically clean up your hand-written code according to criteria that you specify. However, Dreamweaver never rewrites your code unless the code rewriting options are enabled or you perform an action that changes the code. For example, Dreamweaver does not alter your white space or change the case of attributes unless you use the Apply Source Formatting command.

A few of these code rewriting options are enabled by default. For information on how to disable them, or enable others, see "Setting code rewriting preferences" on page 404.

The Roundtrip HTML capabilities in Dreamweaver let 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. These capabilities include the following:

- Dreamweaver lets you start a third-party text editor to edit the current document. For more information, see "Using an external HTML editor with Dreamweaver" on page 412.
- By default, Dreamweaver does not make changes in code created or edited in other HTML editors, even if the code is invalid, unless you enable code-rewriting options.
- 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><em>text</MyNewTag></em>`), Dreamweaver marks it as an error but doesn't rewrite the code.
- Optionally, you can set Dreamweaver to highlight invalid HTML in Code view (in yellow). When you select a highlighted section, the Property inspector displays information on how to correct the error.

Related topics

- "Setting coding preferences" on page 403
- "Setting viewing preferences" on page 403
- "About server behavior code" on page 423

About the XHTML code generated by Dreamweaver

Dreamweaver generates new XHTML code and cleans up existing XHTML code in a way that meets most of the XHTML requirements. Dreamweaver also provides you with the tools that you need to meet the few XHTML requirements that remain.

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	Actions Dreamweaver performs to meet this requirement
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: `<?xml version="1.0" encoding="iso-8859-1"?>`

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:
	`<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">`
	Or, if the XHTML document has a frameset:
	`<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">`

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: `<html xmlns="http://www.w3.org/1999/xhtml">`
A standard document must have the `head`, `title`, and `body` structural elements. A frameset document must have the `head`, `title`, and `frameset` structural elements.	In a standard document, includes the `head`, `title`, and `body` elements. In a frameset document, includes the `head`, `title`, and `frameset` elements.

All elements in the document must nest properly:
`<p>This is a <i>bad example.</p></i>`
`<p>This is a <i>good example.</i></p>`
	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 `/>`. For example, ` ` is not valid; the correct form is ` </br>` or ` `. 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 `/>` (for example, ` `, not ` `).	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, `<td nowrap>` is not valid; the correct form is `<td nowrap="nowrap">`. 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. *Note:* If an HTML browser does not support HTML 4, it might fail to interpret these Boolean attributes when they appear in their full form.

XHTML requirement	Actions Dreamweaver performs to meet this requirement
All attribute values must be surrounded by quotation marks.	Places quotation marks around 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, `<a name="intro">Introduction</a>` is not valid; the correct form is `<a id="intro">Introduction</a>` or `<a id="section1" name="intro"> Introduction</a>`.	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.
For attributes with values of an enumerated type, the values must be lowercase. 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`.	Forces enumerated type values to be lowercase in the code that it generates, and when cleaning up XHTML.
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.)	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.
All `img` and `area` elements must have an `alt` attribute.	Sets these attributes in the code that it generates and, when cleaning up XHTML, reports missing `alt` attributes.

Server-side includes

A server-side include is a file that the server incorporates into your document when a browser requests your document from the server.

When a visitor's browser requests the document containing the include instruction, your server processes the include instruction and creates a new document in which the include instruction is replaced by the contents of the included file. The server then sends this new document to the visitor's 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 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. Note that 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. Select the one that is appropriate for the type of web server you use:

- If your server is an Apache web server, select 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), select File. (Virtual works with IIS only in certain specific circumstances.)

  Note: 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, so pages that the server parses are served a little more slowly than other pages; therefore, some system administrators won't provide the option of parsing all files.

Related topics

- "Inserting a server-side include" on page 448
- "Editing the contents of a server-side include" on page 448

Regular expressions

Regular expressions are patterns that describe character combinations in text. Use them in your code searches to help describe concepts such as "lines that begin with 'var'" and "attribute values that contain a number." For more information on searching, see "Searching and replacing tags and attributes" on page 428.

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
^	Beginning of input or line.	`^T` matches "T" in "This good earth" but not in "Uncle Tom's Cabin"
$	End of input or line.	`h$` matches "h" in "teach" but not in "teacher"
*	The preceding character 0 or more times.	`um*` matches "um" in "rum", "umm" in "yummy", and "u" in "huge"
+	The preceding character 1 or more times.	`um+` matches "um" in "rum" and "umm" in "yummy" but nothing in "huge"
?	The preceding character at most once (that is, indicates that the preceding character is optional).	`st?on` matches "son" in "Johnson" and "ston" in "Johnston" but nothing in "Appleton" or "tension"
.	Any single character except newline.	`.an` matches "ran" and "can" in the phrase "bran muffins can be tasty"
x|y	Either x or y.	`FF0000|0000FF` matches "FF0000" in `bgcolor="#FF0000"` and "0000FF" in `font color="#0000FF"`
{n}	Exactly n occurrences of the preceding character.	`o{2}` matches "oo" in "loom" and the first two o's in "moooooo" but nothing in "money"
{n,m}	At least n, and at most m, occurrences of the preceding character.	`F{2,4}` matches "FF" in "#FF0000" and the first four F's in #FFFFFF
[abc]	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]).	`[e-g]` matches "e" in "bed", "f" in "folly", and "g" in "guard"
[^abc]	Any character not enclosed in the brackets. Specify a range of characters with a hyphen (for example, [^a-f] is equivalent to [^abcdef]).	`[^aeiou]` initially matches "r" in "orange", "b" in "book", and "k" in "eek!"
\b	A word boundary (such as a space or carriage return).	`\bb` matches "b" in "book" but nothing in "goober" or "snob"
\B	Anything other than a word boundary.	`\Bb` matches "b" in "goober" but nothing in "book"
\d	Any digit character. Equivalent to [0-9].	`\d` matches "3" in "C3PO" and "2" in "apartment 2G"
\D	Any nondigit character. Equivalent to [^0-9].	`\D` matches "S" in "900S" and "Q" in "Q45"

Character	Matches	Example
\f	Form feed.	
\n	Line feed.	
\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 Design view, not as line breaks.	

Use parentheses to set off groupings within the regular expression to be referred to later. Then use $1, $2, $3, and so on in the Replace With field to refer to the first, second, third, and later parenthetical groupings.

Note: In the Search For text box, to refer to a parenthetical grouping earlier in the regular expression, use \1, \2, \3, and so on instead of $1, $2, $3.

For example, searching for (\d+)\/(\d+)\/(\d+) and replacing it with $2/$1/$3 swaps the day and month in a date separated by slashes, thereby converting between American-style dates and European-style dates.

Related topics

- "Searching for tags, attributes, or text contained in specific tags" on page 428
- "Saving search patterns" on page 429

About server behavior code

When you develop a dynamic page and select a server behavior from the Server Behaviors panel, Dreamweaver inserts one or more code blocks into your page to make the server behavior work.

If you manually change the code within a code block, you can no longer use panels such as the Bindings and Server Behaviors panels to edit 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 code block's code in any way, 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.

Using code hints

The code hints feature helps you insert and edit code quickly and without mistakes. When you type certain characters in Code view, such as the first letters of a tag or attribute or CSS property name, 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.

Code hints are available for a variety of kinds of code. An appropriate list of items appears when you type a particular character that indicates the beginning of a piece of code; for example, to display a code hints list of HTML tag names, type a start bracket (<).

Tip: For best results, especially when using code hints for functions and objects, set the Delay option in the Code Hints preferences dialog box to a delay of 0 seconds. For more information, see "Setting code hints preferences" on page 404.

The code hints list disappears when you press Backspace (Windows) or Delete (Macintosh).

To display a code hints menu if it doesn't appear automatically:

- Press Control+Spacebar.

To insert markup or other code in Code view using code hints:

1. Type the beginning of a piece of code. For example, to insert a tag, type a start bracket (<); to insert an attribute, place the insertion point immediately after a tag name and press Spacebar.

   A list of items (such as tag names or attribute names) appears.

   Tip: To close the list at any time, press Escape.

2. Scroll through the list using the scroll bar or the Up Arrow and Down Arrow keys.
3. To insert an item from the list, double-click it, or select it and press Enter (Windows) or Return (Macintosh).

   Tip: If a recently created CSS style doesn't appear in a code hints list of CSS styles, select Refresh Style List from the code hints list. If Design view is showing, some invalid code might appear in Design view temporarily after you select Refresh Style List; to remove that invalid code from Design view, after you finish inserting the style, press F5 to refresh Design view again.

To edit a tag using code hints, do either of the following:

- To replace an attribute with a different attribute, first delete the attribute and its value, then add a new attribute and its value as described in the previous procedure.
- To change a value, first delete the value, then add a new value as described in the previous procedure.

Related topics

- "Setting code hints preferences" on page 404.

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 predefined snippets that you can use as a starting point.

This section describes how to insert, create, edit or delete code snippets. It also describes how to manage your code snippets and share them with other members of your team.

To insert a code snippet:

1. Place the insertion point where you want to insert the code snippet, or select code to wrap a snippet around.
2. In the Snippets panel (Window > Snippets), double-click the snippet.

   You can also right-click (Windows) or Control-click (Macintosh) the snippet, then select 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 a code snippet:

- In the Snippets panel, select a snippet and click the Edit Snippet button at the bottom of the panel.

To delete a code snippet:

- In the Snippets panel, select a snippet and click the Remove button at the bottom of the panel.

To create code snippet folders and manage code snippets:

1. In the Snippets panel, click the New Snippet Folder button at the bottom of the panel.
2. Drag snippets to the new folder or other folders, as desired.

To add or edit a keyboard shortcut for a snippet:

1. In the Snippets panel, right-click (Windows) or Control-click (Macintosh) and select Edit Keyboard Shortcuts.
2. The Keyboard Shortcuts Editor appears.
3. In the Commands pop-up menu, select Snippets.

4 A list of snippets appears.

5 Select a snippet and assign a keyboard shortcut to it. For more information, see "Customizing keyboard shortcuts" on page 57.

To share a snippet with other members of your team:

1 Find the file corresponding to the snippet that you want to share in the Configuration/Snippets folder in the Dreamweaver MX application folder.

2 Copy the snippet file to a shared folder on your computer or a network computer.

3 Have the other members of the team copy the snippet file to their Configuration/Snippets folder.

Inserting code quickly with the Insert bar

You can use the Insert bar to quickly add code to your page.

To insert code quickly:

1 Position the insertion point in the code.

2 Select an appropriate category in the Insert bar.

3 Click a button in the Insert bar, or select an item from a pop-up menu 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 button does, point to the button with the mouse pointer and wait for a tooltip to appear. The number and type of buttons available in the Insert bar varies depending on the current document type. 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.

Related topics

- "The Insert bar" on page 37

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 Macromedia ColdFusion and ASP.NET tag libraries). For more information on the tag libraries, see "Managing tag libraries" on page 406.

To insert a tag using the Tag Chooser:

1 Position the insertion point in the code, then right-click (Windows) or Control-click (Macintosh) and select 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.

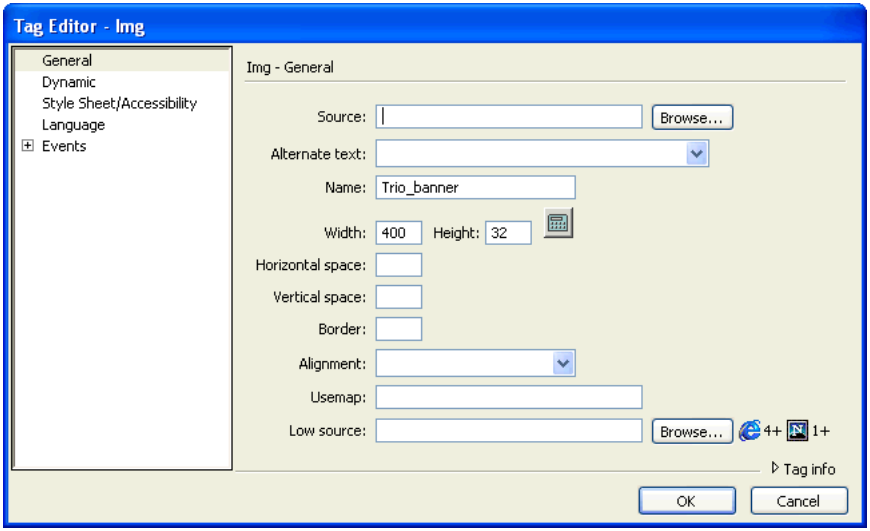
2 Select and insert a tag.

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

3 To close the Tag Chooser, click the Close button.

Editing tags with Tag editors

Tag editors let you view, specify, and edit the attributes of a tag.



To edit a tag with a Tag editor:

1 Right-click (Windows) or Control-click (Macintosh) a tag in Code view or an object in Design view, and select Edit Tag from the pop-up menu.

2 Specify or edit attributes for the tag and click OK.

Tip: To get more information about the tag within the Tag editor, click Tag Info.

Indenting code blocks

As you write and edit code in Code view or the Code inspector, you can change the indentation level of a selected block or line of code, shifting it right or left by one tab.

To indent the selected block of code, do one of the following:

- Press Tab.
- Select Edit > Indent Code.

To unindent the selected block of code, do one of the following:

- Press Shift+Tab.
- Select Edit > Outdent Code.

Related topics

- "Making quick changes to a code selection" on page 430

Copying and pasting code

You can copy and paste code from another application or from Code view itself. You can copy and paste the code as text (so that it appears on the page when viewed in a browser) or as code (so that browsers interpret it as, for example, HTML tags).

To copy and paste code as text:

1. Copy the code from Dreamweaver or another application.
2. Place the insertion point in Design view and select Edit > Paste.

   The code appears in Design view. All angle brackets are encoded as HTML entities, so that browsers interpret the code as text rather than as tags.

To copy and paste code as code:

1. Copy the code from Dreamweaver or another application.

   To copy from Design view, select Edit > Copy as HTML.
2. Do one of the following:
   - In Code view, position the insertion point and select Edit > Paste. To update the display in Design view, press F5.
   - In Design view, position the insertion point, and select Edit > Paste as HTML.

Inserting HTML comments

A comment is descriptive text that you insert in HTML code to explain the code or provide other information. The text of the comment appears only in Code view and is not displayed in a browser.

To insert a comment at the insertion point:

- Select Insert > Comment.

  In Code view, Dreamweaver inserts a comment tag and places the insertion point in the middle of the tag. Type your comment.

  In Design view, Dreamweaver displays the Comment dialog box. Type your comment and click OK.

To display comment markers in Design view:

- Select View > Visual Aids > Invisible Elements.

  Make sure that the Comments option is selected in the Invisible Elements preferences or the comment marker will not appear.

To edit an existing comment, do one of the following:

- In Design view, select the Comment marker and edit the comment's text in the Property inspector.
- In Code view, find the comment and edit its text.

Viewing JavaScript and VBScript functions in your document

You can view a list of all of the JavaScript or VBScript functions in your document, and you can jump to any function.

To view the names of the script functions in a document:

1. View the document in Code view.
2. Right-click (Windows) or Control-click (Macintosh) anywhere in Code view. In the context menu, select the Functions submenu.

   Note: The Functions submenu does not appear in Design view.

   If your code contains JavaScript or VBScript functions, they appear in the submenu.

   Tip: To see the functions listed in alphabetical order, Control-right-click (Windows) or Option-Control-click (Macintosh) in Code view, then select the Functions submenu.

To view a function in your code:

- Select a function name from the Functions submenu.

  Code view displays the function, with the function name highlighted.

Searching and replacing tags and attributes

You can use Dreamweaver to search for and replace tags and attributes in your code.

Related topics

- "Regular expressions" on page 420
- "Searching for and replacing text" on page 301

Searching for tags, attributes, or text contained in specific tags

You can search for specific tags, attributes, and attribute values. For example, you can search for all `img` tags that have no `alt` attribute.

You can also 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` contained in a `title` tag to find all the untitled pages on your site.

To search for tags, attributes, or text in code:

1. Open the document to search in, or select documents or a folder in the Files panel.
2. Select Edit > Find and Replace.

   The Find and Replace dialog box appears.
3. Specify which files to search in, then specify the kind of search you want to perform, and text or tags to search for. Optionally, specify replacement text as well. Then click one of the Find buttons or one of the Replace buttons.

   For more information, click the Help button.
4. When you're done, click the Close button to dismiss the dialog box.

To search again without displaying the Find and Replace dialog box:

- Press F3 (Windows) or Command+G (Macintosh).

Related topics

- "Regular expressions" on page 420

Saving search patterns

You can save a search pattern and reuse it later.

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, attributes, or text contained in specific tags" on page 428 for information about setting additional search parameters.
2. Click the Save Query button (the disk icon).
3. In the dialog box that appears, navigate to the folder where you want to save queries. Then type a filename to identify the query by 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.

   Note: Saved queries have the filename extension .dwr. Some saved queries from older versions of Dreamweaver may have the extension .dwq.

To recall a search pattern:

1. Select Edit > Find and Replace.
2. Click the Load Query button (the folder icon).
3. Navigate to the folder where your queries are saved. Then select a query file and click Open.
4. Click Find Next, Find All, Replace, or Replace All to initiate the search.

Related topics

- "Searching for tags, attributes, or text contained in specific tags" on page 428
- "Regular expressions" on page 420

Making quick changes to a code selection

You can select code and then make quick changes to it using a context menu.

To make quick changes to selected code:

1. In Code view, select some code and right-click (Windows) or Control-click (Macintosh). In the context menu, select the Selection submenu.
2. Select one of the following options from the submenu:

   Comment Out Lines adds comment tags around selected lines. If a line is partially selected, the entire line is commented out. You can use this tool to comment out potentially incorrect code when debugging a page.

   Uncomment Lines removes comment marks from the beginning and end of any lines within the commented out selection. It works on entire lines only, not on comments within lines.

   Convert Tabs to Spaces converts each tab in the selection to a number of spaces equal to the Tab Size value set in Code Format preferences. For more information, see "Setting code formatting preferences" on page 403.

   Convert Spaces to Tabs converts runs of spaces in the selection to tabs. Each run of spaces that has a number of spaces equal to the tab size is converted to one tab.

   Indent indents the selection, shifting it to the right. For more information about indenting and outdenting, see "Indenting code blocks" on page 426.

   Outdent shifts the selection to the left.

   Remove All Tags removes all the tags in the selection.

   Convert Lines to Table wraps the selection in a `table` tag with no attributes.

   Add Line Breaks adds a `br` tag at the end of each line of the selection.

   Convert to Uppercase converts all letters in the selection (including tag and attribute names and values) to uppercase.

   Convert to Lowercase converts all letters in the selection (including tag and attribute names and values) to lowercase.

   Convert Tags to Uppercase converts all tag and attribute names and attribute values in the selection to uppercase.

   Convert Tags to Lowercase converts all tag and attribute names and attribute values in the selection to lowercase.

Using language-reference material

The Reference panel provides you with a quick reference tool for markup languages, programming languages, and CSS styles; it provides information on the specific tags, objects, or styles that you are working with in Code view (or the Code inspector).

To open the Reference panel:

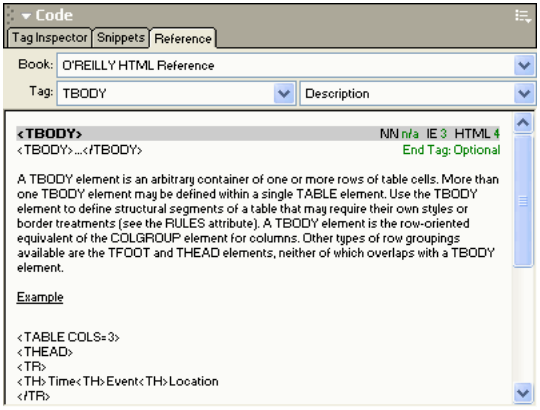
1 Do one of the following in Code view:

- Right-click (Windows) or Control-click (Macintosh) a tag, attribute, or keyword and select Reference from the context menu.
- Place the insertion point in a tag, attribute, or keyword and press F1.

The Reference panel opens and displays information about the tag, attribute, or keyword you clicked.

2 To adjust the text size in the Reference panel, select Large Font, Medium Font, or Small Font from the options menu (the small arrow at the top right of the panel).

3 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, select a different book from the Book pop-up menu.



4 The Tag, Object, Style, or CFML pop-up menu (depending on which book you have selected) displays the tag, object, style, or function that you selected in Code view (or the Code inspector). To view information about another item, select a new one from the menu.

5 To view information about an attribute of the selected item, select the attribute from the pop-up menu next to the Tag, Object, or Style pop-up menu.

This menu contains the list of attributes for the item you select. The default selection is Description, which displays a description of the chosen item.

Printing your code

You can print your code to edit it offline, archive it, or distribute it.

To print code:

1. Open a page in Code view.
2. Select File > Print Code.
3. Specify printing options, then click OK (Windows) or Print (Macintosh).

CHAPTER 21
Optimizing and Debugging Your Code

You can use Macromedia Dreamweaver MX 2004 to optimize and debug your code. For example, you can validate your tags, make your document XHTML-compliant, or debug your ColdFusion code.

This chapter covers the following topics:

- "Cleaning up your code" on page 433
- "Verifying that tags and braces are balanced" on page 434
- "Checking for browser compatibility" on page 434
- "Validating your tags" on page 437
- "Making pages XHTML-compliant" on page 438
- "Using the ColdFusion debugger (Windows only)" on page 439

Cleaning up your code

Dreamweaver can automatically remove empty tags, combine nested `font` tags, and otherwise improve messy or unreadable HTML or XHTML code.

For information on how to clean up HTML generated from a Microsoft Word document, see "Opening existing documents" on page 73.

To clean up the code:

1 Open a document:

- If the document is in HTML, select Commands > Clean Up HTML.
- If the document is in XHTML, select Commands > Clean Up XHTML.

  For an XHTML document, the Clean Up XHTML command fixes XHTML syntax errors, sets the case of tag attributes to lowercase, and adds or reports the missing required attributes for a tag in addition to performing the HTML cleanup operations.

2 In the dialog box that appears, select any of the options.

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

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.

Related topics

- "Setting code formatting preferences" on page 403
- "Setting code coloring preferences" on page 405

Verifying that tags and braces are balanced

You can check to make sure the tags, parentheses (( )), braces ({ }), and square brackets ([ ]) in your page are balanced. Balanced means that every opening tag, parenthesis, brace, or bracket has a corresponding closing one, and vice versa.

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 Select Edit > Select Parent Tag.

The enclosing matching tags (and their contents) are selected in your code. If you keep selecting Edit > Select Parent Tag, and your tags are balanced, eventually Dreamweaver will select the outermost `html` and `/html` tags.

To check for balanced parentheses, braces, or square brackets:

1 Open the document in Code view.

2 Place the insertion point in the code you want to check.

3 Select Edit > Balance Braces.

All of the code between the enclosing parentheses, braces, or square brackets is selected. Choosing Edit > Balance Braces again selects all of the code inside the parentheses, braces, or square brackets that enclose the new selection.

Checking for browser compatibility

Dreamweaver enables you to create web pages with elements that are supported by all major graphical browsers (such as images and text), as well as with elements that are supported only by newer browsers (such as CSS styles).

You should determine who the audience for your site is likely to be, and what browsers they're likely to use. If most of your users will be using Netscape 4 (as is still true in some academic intranet environments, for example), you should avoid using tags that aren't supported in that browser.

The Check Target Browsers feature tests the code in your documents to see if any of the tags, attributes, CSS properties, or CSS values are unsupported by your target browsers. The check does not alter the document in any way.

The target browser check gives information about three levels of potential problems: errors, warnings, and informational messages. The following descriptions explain the differences between those levels:

- An error indicates code that may cause a serious visible problem in a particular browser, such as causing parts of a page to disappear. (In some cases, code with an unknown effect is also marked as an error.)
- A warning indicates a piece of code that won't be displayed correctly in a particular browser, but that won't cause any serious display problems.

- An informational message indicates code that isn't supported in a particular browser, but that has no visible effect; for example, the `img` tag's `galleryimg` attribute isn't supported in some browsers, but those browsers ignore that attribute, so it has no visible effect.

By default, Dreamweaver automatically performs a target browser check whenever you open a document. You can also run a target browser check manually on a document, on a folder, or on an entire site.

Note: The target browser check is not updated continually. After you make changes to code, run the target browser check manually to confirm that you've eliminated code that doesn't work with your target browsers.

The target browser check does not check scripts in your site. Also, the target browser check is not a syntax validator; it detects only markup that isn't supported in your target browsers.

Note: 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 version 3.0 and later, for Microsoft Internet Explorer version 3.0 and later, and Opera version 2.1 and later. For information on how to modify the existing profiles or to create new ones, see the "Customizing Dreamweaver" article at www.macromedia.com/support/dreamweaver/.

To select browsers for Dreamweaver to check against:

1. Select Settings from the Target Browser Check menu in the Document toolbar.

   The Check Target Browsers dialog box appears.
2. Select the checkbox next to each browser you want to check.
3. For each selected browser, select a minimum version to check against from the corresponding pop-up menu.

   For example, to verify that all of the document's markup is supported in Microsoft Internet Explorer 3.0 and later and Netscape Navigator 4.0 and later, select the checkboxes next to those browser names, and select 3.0 from the Microsoft Internet Explorer pop-up menu and 4.0 from the Netscape Navigator pop-up menu.

To view the results of an automatic target browser check, do one of the following:

- Open a file and view the file in Code view (View > Code or View > Code and Design).
- After making a change in Code view, click the Refresh button in the Property inspector or press F5.

A wavy red underline appears under the name of every item that's considered an error in one of your target browsers. (Warnings and informational messages are not marked in Code view; to view warnings and informational messages, view the report for the entire document.) If Dreamweaver finds no unsupported markup, then nothing is underlined, and the Target Browser Check menu icon in the Document toolbar changes to indicate that there are no errors.

Tip: To easily use various commands related to target browser checks, select a command from the context menu by right-clicking (Windows) or Control-clicking (Macintosh) any red-underlined item.

To see which browsers don't support a specific red-underlined item:

- Position the pointer to point to the red-underlined text.

A tooltip appears, indicating which browsers don't support that item.

To view the Check Target Browser report for the entire document:

- In the Target Browser Check menu in the Document toolbar, select Show All Errors.

The Results panel group appears, with the Target Browser Check panel selected. Errors are marked with a red exclamation mark icon, warnings are marked with a yellow exclamation mark icon, and informational messages are marked with a word-balloon icon.

To view a long error message in the Target Browser Check panel:

1. Open the Target Browser Check panel.
2. Select an error message that's too long to read in the panel.
3. Click the More Info button.

A Description dialog box appears, displaying the full text of the selected error message.

To disable the automatic target browser check:

- Deselect Auto-Check On Open from the Target Browser Check menu in the Document toolbar.

To jump to the next or previous error in the code:

- Select Next Error or Previous Error from the Target Browser Check menu in the Document toolbar.

To jump to a specific error from the Target Browser Check panel:

- Double-click the error message.

The unsupported markup is selected in Code view.

To run a target browser check manually on the current file:

- Select File > Check Page > Check Target Browsers.

The report appears in the Target Browser Check panel (in the Results panel group).

To run a target browser check manually on a site or a set of selected files:

1. In the Local view of the Files panel, select a set of files, or select the folder that contains the entire site.
2. Select File > Check Page > Check Target Browsers.

   The report appears in the Target Browser Check panel (in the Results panel group).
3. To cancel the report while it's running, click the Stop button in the Target Browser Check panel.

To switch between viewing the current document report and viewing the full site report:

- In the Target Browser Check panel, select Current Document or Site Report from the pop-up menu.

To save a target browser check report:

- Click the Save Report button in the Target Browser Check panel.

  Note: The target browser check report is not saved automatically; if you want to keep a copy for future reference, save it.

To view a target browser check report in a browser:

- Click the Browse Report button in the Target Browser Check panel.

To fix an error:

- Remove the unsupported code, or change it to other code that's supported by your target browsers.

To specify that Dreamweaver should ignore a particular type of error:

- Right-click (Windows) or Control-click (Macintosh) red-underlined code, and select Ignore Error from the context menu.

That type of error is changed to a warning, and Dreamweaver stops underlining that type of error in all documents.

Validating your tags

You can use Dreamweaver to find out if your code has tag or syntax errors. Dreamweaver can validate documents in many languages, including HTML, XHTML, ColdFusion Markup Language (CFML), JavaServer Pages (JSP), Wireless Markup Language (WML), and XML.

Note: For information on how to validate your document for accessibility, see "Testing your site" on page 111.

You can validate the current document or a selected tag.

To validate your tags:

1. Do one of the following:
   - For an XML (or XHTML) file, select File > Check Page > Validate as XML.
   - Otherwise, select File > Check Page > Validate Markup.

   The Validation tab of the Results panel either displays a "No errors or warnings" message or lists the syntax errors it found.
2. Double-click an error message to highlight the error in the document.
3. To save the report as an XML file, click the Save Report button.
4. To view the report in your primary browser (which lets you print the report), click the Browse Report button.

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 for, and the types of errors that the Validator should report. For more information, see "Setting Validator preferences" on page 405.

Related topics

"Checking for browser compatibility" on page 434

Making pages XHTML-compliant

When you create a new page, you can make it XHTML-compliant. You can also make an existing HTML document XHTML-compliant.

To create a new, XHTML-compliant document:

1 Select File > New.

The New Document dialog box appears.

2 Select a document type.

3 Select the Make Document XHTML Compliant option.

Note: Not all document types can be made XHTML-compliant.

4 Click OK.

To create XHTML-compliant documents by default:

1 Select Edit > Preferences or Dreamweaver > Preferences (Mac OS X) and select the New Document category.

2 In the New Document category, select 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, then do one of the following:
  - For a document without frames, select File > Convert > XHTML.
  - For a document with frames, select a frame and select 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, a document based on an XHTML template will always be in XHTML, and a document based on a non-XHTML-compliant HTML template will always be HTML and can't be converted to XHTML or any other language.

Related topics

- "About the XHTML code generated by Dreamweaver" on page 417

Using the ColdFusion debugger (Windows only)

If you're a ColdFusion developer using Macromedia ColdFusion MX as your Dreamweaver testing server, you can view ColdFusion debugging information 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 errors appears at the bottom of the page.

Before you begin debugging, make sure debugging settings are enabled in ColdFusion Administrator. For more information, see the ColdFusion documentation in Dreamweaver Help (Help > Using ColdFusion). Also, make sure your Dreamweaver testing server is running ColdFusion MX. For more information on the Dreamweaver testing server, see "Specifying where dynamic pages can be processed" on page 461.

Tip: 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, select 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.

To debug a ColdFusion page:

1 Open the ColdFusion page in Dreamweaver.

2 Click the Server Debug icon on the Document toolbar.

Dreamweaver requests the page 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 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).

CHAPTER 22
Editing Code in Design View

Macromedia Dreamweaver MX 2004 lets you visually create and edit web pages without worrying about the underlying source code, but there are times when you might need to edit the code for greater control or to troubleshoot your web page. Dreamweaver lets you edit some code while working in Design view.

This chapter is designed for people who prefer to work in Design view, but who also want quick access to the code.

The chapter contains the following topics:

- "Editing code with the Property inspector" on page 441
- "Changing attributes with the Tag inspector" on page 442
- "Editing code with the Quick Tag Editor" on page 443
- "Editing code with the tag selector" on page 446
- "Editing scripts" on page 446
- "Working with server-side includes" on page 448
- "Using JavaScript behaviors" on page 449
- "Viewing and editing head content" on page 449

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. The properties shown in the Property inspector generally correspond to attributes of tags; changing a property in the Property inspector generally has the same effect as changing the corresponding attribute in Code view.

Note: The Tag inspector and the Property inspector both allow you to view and edit a tag's attributes. The Tag inspector allows you to view and edit every attribute associated with a given tag. The Property inspector shows only the most common attributes, but provides a richer set of controls for changing those attributes' values, and allows you to edit certain objects (such as table columns) that don't correspond to specific tags.

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.

Related topics

- "Using the Property inspector" on page 45

Changing attributes with the Tag inspector

You can use the Tag inspector to edit tags and objects using a property sheet similar to the ones found in other integrated development environments (IDEs).

Note: The Tag inspector and the Property inspector both allow you to view and edit a tag's attributes. The Tag inspector enables you to view and edit every attribute associated with a given tag. The Property inspector shows only the most common attributes, but provides a richer set of controls for changing those attributes' values, and allows you to edit certain objects (such as table columns) that don't correspond to specific tags.

To edit attributes using the Tag inspector:

1 Do one of the following in the Document window:

- In Code view (or the Code inspector), click anywhere in a tag's name or in its contents.
- In Design view, select an object, or select a tag in the Tag Selector.

2 Open the Tag inspector (Window > Tag Inspector), and select the Attributes tab.

The selection's attributes and their current values appear in the Tag inspector.

3 Do any of the following in the Tag inspector:

- To view the attributes organized by category, click the Show Category View button.

- To view the attributes in an alphabetical list, click the Show List View button.
- To change an attribute value, select the value and edit it (see the procedure below).
- To add a value for an attribute with no value, click in the attribute-value column to the right of the attribute and add a value (see the procedure below).
- To delete an attribute value, select the value and press Backspace (Windows) or Delete (Macintosh).
- To change the name of an attribute, select the attribute name and edit it.

  Note: If you change the name of a standard attribute and then add a value for that attribute, the attribute and its new value move to the appropriate category.

- To add a new attribute not already listed, click in the empty space below the last listed attribute name and type a new attribute name.

4 Press Enter (Windows) or Return (Macintosh), or click elsewhere in the Tag inspector, to update the tag in your document.

To edit or add an attribute value, do one of the following:

- Type a new value for the attribute in the attribute-value column, to the right of the attribute name.
- If the attribute takes pre-defined values, select a value from the pop-up menu (or the color picker) to the right of the attribute-value column.
- If the attribute takes a URL value, click the Browse button or use the Point-to-File icon to select a file, or type the URL in the text box.
- If the attribute takes a value from a source of dynamic content (such as a database), click the Dynamic Data button to the right of the attribute-value column. Then select a source.

For information on defining sources of dynamic content, see "Defining Sources of Dynamic Content" on page 525.

Related topics

- "Editing code with the Property inspector" on page 441
- "Using JavaScript Behaviors" on page 371
- "Using Cascading Style Sheets styles" on page 290

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). For more information, see Dreamweaver Help.

Inserting an HTML tag with the Quick Tag Editor

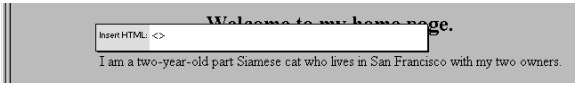
You can use the Quick Tag Editor to insert an HTML tag in your document.

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, along with a matching closing tag if applicable.

4 Press Escape to exit without making any changes.

Related topics

- "Applying HTML to a selection with the Quick Tag Editor" on page 444
- "Using the hints menu" on page 445

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 446.

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.

Related topics

- "Inserting an HTML tag with the Quick Tag Editor" on page 443
- "Using the hints menu" on page 445

Applying HTML to a selection with the Quick Tag Editor

You can use the Quick Tag Editor to wrap the current selection 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.

Note: If you select text or an object that includes an opening or closing HTML tag, the Quick Tag Editor opens in Edit Tag mode instead of Wrap 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 **strong**, and press Enter (Windows) or Return (Macintosh).

The tag is inserted at the beginning of the current selection, and a matching closing tag is inserted at the end.

4 To exit without making any changes, instead of pressing Enter, press Escape.

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.

You can also disable the hints menu or adjust the delay before the menu pops up in the Quick Tag Editor.

To see a hints menu listing valid attributes for a tag, pause briefly 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.

Similarly, to see a hints menu listing valid tag names, pause briefly while entering or editing a tag name in the Quick Tag Editor.

Note: The Quick Tag Editor code hints preferences are controlled by a separate dialog box from the Code view code hints preferences. For more information, see "Using code hints" on page 423.

To use a hints menu:

1. Do one of the following:
   - Begin to type a tag or attribute name. The selection in the Code Hints menu jumps to the first item that starts with the letters you typed.
   - Use the Up and Down Arrow keys to select an item.
   - Use the scroll bar to find an item.
2. Press Enter to insert the selected item, or double-click an item to insert it.
3. To dismiss the hints menu without inserting an item, press Escape or simply continue typing.

To disable the hints menu or change the delay before it appears:

1. Select Edit > Preferences (Windows) or Dreamweaver > Preferences (Macintosh) 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 appears, adjust the Delay slider.

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

Related topics

- "Editing an HTML tag with the Quick Tag Editor" on page 444

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:

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 444.

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 corresponding to 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.

Writing a client-side script in Design view

You can write a JavaScript or VBScript for your page without leaving Design view.

Before starting, select View > Visual Aids > Invisible Elements to ensure that script markers will appear on the page.

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, select View > Visual Aids > Invisible Elements to ensure that script markers will appear on the page.

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 and 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, select View > Visual Aids > Invisible Elements to ensure that script markers will appear on the page.

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 opens in Code view, where you can make your edits.

   Note: If there is code between the script tags, the Script Properties dialog box opens even if there is also a link to an external script file.
3. Edit the script and click OK.

   For more information, click the Help button in the 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. For more information, see "Server-side includes" on page 419.

You can use Dreamweaver to insert server-side includes in your pages, edit the includes, or preview pages containing includes.

Inserting a server-side include

You can use Dreamweaver to insert server-side includes in your page.

To insert a server-side include:

1. Select 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.

Related topics

- "Server-side includes" on page 419

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.
3. Upload the include file to the remote site if necessary.

Related topics

- "Server-side includes" on page 419

Using JavaScript behaviors

The Behaviors tab of the Tag inspector allows you to easily attach JavaScript (client-side) behaviors to page elements. For more information, see "Using the behavior actions that come with Dreamweaver" on page 376.

Viewing and editing head content

You can view the elements in the `head` section of a document using the View menu, the Document window's Code view, or the Code inspector.

To view elements in the head section of a document:

- Select 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.
   - Select 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. Select 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 Dreamweaver Help.

PART VI
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 23, "Setting Up a Web Application," on page 453

Chapter 24, "Database Connections for ColdFusion Developers," on page 463

Chapter 25, "Database Connections for ASP.NET Developers," on page 465

Chapter 26, "Database Connections for ASP Developers," on page 469

Chapter 27, "Database Connections for JSP Developers," on page 479

Chapter 28, "Database Connections for PHP Developers," on page 487

Chapter 29, "Troubleshooting Database Connections," on page 489

CHAPTER 23
Setting Up a Web Application

This chapter describes how to configure your system to build web applications in Macromedia Dreamweaver MX 2004.

What you need to build web applications

To build web applications in Dreamweaver, you need the following software:

- A web server (see "Setting up a web server" on page 454)
- An application server that works with your web server (see "Setting up an application server" on page 454)

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 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 461.

Several web hosting companies offer plans that let you use their software to test and deploy web applications. In some cases, 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 2000 or XP computer) so that other developers on your team can work on a project.

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 Microsoft Internet Information Server (IIS), Netscape Enterprise Server, Sun ONE 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. Windows users who want to develop ColdFusion web applications can use the web server included in the developer edition of the ColdFusion MX application server, which you can install and use for free. For more information, see "Installing a ColdFusion application server" on page 455.

Other Windows users can run a web server on their local computer by installing IIS; Windows 98 users can install Microsoft Personal Web Server (PWS), a scaled-down version of IIS. One of these web servers may already be installed on your system. 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 "Installing a Web Server" in Getting Started Help.

ASP.NET pages only work with one web server: Microsoft IIS 5 or higher. 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 host ASP.NET applications. Windows 98 is not supported. However, you can develop (as opposed to host) ASP.NET applications on any computer running Dreamweaver, including the Macintosh.

If you're a Macintosh user, you can use a web hosting service or install the required software on a remote computer. You can also develop PHP sites locally using the Apache web server and PHP application server installed with your operating system. For more information, see "Installing a PHP application server" on page 457.

For information on installing and configuring other web servers, see the server vendor's documentation or your system administrator.

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 dynamic pages.

Choosing your application server

The application server you choose 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 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 "Choosing a server technology" in Getting Started Help. 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 6 with .NET Framework
ASP	Microsoft IIS or PWS Sun ONE Active Server Pages
JSP	Macromedia JRun Sun ONE Application Server 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.

Windows users can download and install a fully functional, developer edition of ColdFusion MX from the Macromedia website at www.macromedia.com/go/coldfusion/. A copy of ColdFusion MX Server Developer Edition is also available on the Dreamweaver CD (Windows version only).

Note: The Developer Edition is for non-commercial use for developing and testing web applications. It is not licensed for deployment. It supports requests from the local host and one remote IP address and includes every feature of the Enterprise Edition except Macromedia ClusterCATS and 10K Verity document search. After 30 days, it will only support requests from one IP address, but you can still use it for development and testing as long as you want. The software does not expire.

During installation, you can configure ColdFusion to use the web server built into ColdFusion or another web server installed on your system. Generally, it's best to match your development environment to your production environment as closely as possible. Therefore, if you have an existing web server such as Microsoft IIS on your development computer, you may want to select it instead of the built-in ColdFusion web server.

Note: If you select the ColdFusion web server, you can still manually configure ColdFusion later to work with IIS. For more information, see the Macromedia website at www.macromedia.com/go/coldfusion_iis4/.

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. You can also install ColdFusion on a Mac OS X computer running a J2EE server such as JRun or Tomcat. For more information, see the Macromedia website at www.macromedia.com/go/coldfusion_macintosh/.

After installing the application server, create a root folder for your web application. See "Creating a root folder for the application" on page 458.

Installing an ASP.NET application server

To run 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.

It is highly recommended that you also install the Microsoft Data Access Components (MDAC) 2.7 package after installing the .NET Framework. You can download and install the MDAC 2.7 package for free from the Microsoft website at www.microsoft.com/data/download.htm.

Macintosh or Windows 98 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 458.

Installing an ASP application server

To run 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 2000 and Windows XP Professional
- Microsoft PWS, a scaled-down version of IIS that can run on Windows 98 computers
- Sun ONE Active Server Pages, versions of which run on Windows NT or 2000, Linux, or Solaris. For more information, see the Sun website at wwws.sun.com/software/chilisoft/index.html

Windows users can install and run IIS or PWS on their local computer. For instructions, see "Installing a Web Server" in Getting Started Help.

Macintosh users can use a web hosting service with an ASP plan or install IIS 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 458.

Installing a JSP application server

To run JSP pages, you need an application server that supports JavaServer Pages. Here are some popular choices:

- Macromedia JRun for Windows, Mac OS X, Linux, Solaris, or UNIX. You can download a trial version of JRun from the Macromedia website at www.macromedia.com/go/jrun/.
- Tomcat for Windows and UNIX (including Mac OS X). You can download a copy of Tomcat from the Jakarta Project website at http://jakarta.apache.org/tomcat/. For more information on installing Tomcat on a Macintosh, see the Apple website at http://developer.apple.com/internet/java/tomcat1.html.
- IBM WebSphere for various operating systems. You can download a trial version from the IBM website at www7b.software.ibm.com/wsdd/downloads/WASsupport.html#download.

After installing a JSP application server, create a root folder for your web application. See "Creating a root folder for the application" on page 458.

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 Sun ONE servers, and almost all web servers that support the CGI interface.

You can download the PHP application server from the PHP website at 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 www.php.net/download-docs.php.

If you're a Macintosh user, you can use the PHP application server installed with your operating system. For more information, see the following websites:

- www.macromedia.com/go/php_macintosh/
- http://developer.apple.com/internet/macosx/php.html

After installing the PHP application server, create a root folder for your web application.

Creating a root folder for the application

After signing up with a web hosting company or setting up the server software yourself, 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 ColdFusion MX, any file in the \CFusionMX\wwwroot folder or any of its subfolders can be served to a web browser.

The following are the default root folders of selected web servers:

Web server	Default root folder
ColdFusion MX	\CFusionMX\wwwroot
IIS or PWS	\Inetpub\wwwroot
Apache (Windows)	\apache\htdocs
Apache (Macintosh)	Users:MyUserName:Sites
Jakarta Tomcat (Windows)	\jakarta-tomcat-4.x.x\webapps\ROOT\

To test the web server, place a test HTML page in the default root folder and attempt to open it by entering the page's URL in a browser. The URL comprises the domain name, such as www.mysite.com, and the filename of the HTML page, as follows:

www.mysite.com/testpage.htm

If the web server is running on your local computer, you can use **localhost** instead of a domain name. Using the previous example, enter the following localhost URL depending on your web server:

Web server	Localhost URL
ColdFusion MX	http://localhost:8500/testpage.htm
IIS or PWS	http://localhost/testpage.htm
Apache (Windows)	http://localhost:80/testpage.htm
Apache (Macintosh)	http://localhost/~MyUserName/testpage.htm (where MyUserName is your Macintosh user name)
Jakarta Tomcat (Windows)	http://localhost:8080/testpage.htm

Note: By default the ColdFusion MX web server runs on port 8500 and the Jarkarta Tomcat web server runs on port 8080.

If the page doesn't open as expected, check for the following errors:

- The web server is not started. Consult the web server's documentation for starting instructions.
- The file does not have an .htm or .html extension.
- You entered the page's file path (for example, c:\CFusionMX\wwwroot\testpage.htm), not its URL (for example, http://localhost:8500/testpage.htm), in the browser's Address text box.
- The URL contains a typing mistake. Check for errors and make sure the filename is not followed by a slash, such as http://localhost:8080/testpage.htm/.

After creating a root folder for your application, define a Dreamweaver site to manage your files.

Defining a Dreamweaver site

After configuring your system to develop web applications, define a Dreamweaver site to manage your files.

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 454.
- An application server is installed and running on the system running your web server. See "Setting up an application server" on page 454.
- 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 458.

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 459).
2. Define a folder located on the computer running your web server as a Dreamweaver remote folder (see "Defining a remote folder" on page 460).
3. Specify where Dreamweaver should send dynamic pages to be processed while you work (see "Specifying where dynamic pages can be processed" on page 461).

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. 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, select Site > Manage Sites, click the New button on the Manage Sites dialog box, and select Site from the context menu.

   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 text box, enter a descriptive name for your Dreamweaver site.
5. In the Local Root Folder text 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 458).

You don't need to define a remote folder if the folder you defined in "Defining a local folder" on page 459 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 selecting Site > Manage Sites, selecting your site from the list in the Manage Sites dialog box, and clicking Edit.

   The Site Definition dialog box appears.

2. If the wizard is showing, click Advanced and select Remote Info from the Category list.

   The Remote Info dialog box appears.

3. In the Access pop-up menu, select 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 selecting SourceSafe Database. (SourceSafe is used by developers for file version control.) If you select this option, you need to define a separate folder. For instructions, see "Specifying where dynamic pages can be processed" on page 461.

4. After selecting 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 458) because the server probably also runs an application server that can handle the dynamic pages.

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 selecting Site > Manage Sites, selecting your site from the list in the Manage Sites dialog box, and clicking Edit.

The Site Definition dialog box appears.

2 If the wizard is showing, click Advanced and 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.

After defining a Dreamweaver site, you can connect to a database.

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 747.

Dreamweaver handles database connections differently depending on your choice of server technology. See the following chapters:

- Chapter 24, "Database Connections for ColdFusion Developers," on page 463
- Chapter 25, "Database Connections for ASP.NET Developers," on page 465
- Chapter 26, "Database Connections for ASP Developers," on page 469
- Chapter 27, "Database Connections for JSP Developers," on page 479
- Chapter 28, "Database Connections for PHP Developers," on page 487

CHAPTER 24
Database Connections for ColdFusion Developers

You can connect to databases when developing Macromedia ColdFusion applications with Macromedia Dreamweaver MX 2004.

The chapter assumes you have set up a ColdFusion web application (see Chapter 23, "Setting Up a Web Application," on page 453). It also assumes a database is set up on your local computer or on a system to which you have network or FTP access.

To connect to the sample database provided by Dreamweaver, see "Setup for Sample ColdFusion Site" in Getting Started Help. To learn more about databases and database connections, see Appendix A, "Beginner's Guide to Databases," on page 741.

Connecting to a database

When developing a ColdFusion web application in Dreamweaver, you connect to a database by selecting a ColdFusion data source defined in ColdFusion Administrator, the server's management console.

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 461.

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 464.

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 464.

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 4 or 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 4 or 5. For more information, see "Using a DSN" on page 751.

To create a ColdFusion data source:

1 Open any ColdFusion page in Dreamweaver.

2 In the Databases panel (Window > Databases) in Dreamweaver, click the Modify Data Sources icon on the panel toolbar.

ColdFusion Administrator opens in a browser.

3 Log in to ColdFusion Administrator and create the data source.

For instructions, see the ColdFusion documentation from within Dreamweaver (Help > Using ColdFusion).

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 should appear in the panel.

If the data sources do not appear, make sure you complete the checklist in the panel. Also 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 461.

Related topics

- Chapter 29, "Troubleshooting Database Connections," on page 489

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 on the panel toolbar. ColdFusion Administrator opens in a browser. Log in and modify the data source. For instructions, see Using ColdFusion Help.

To avoid getting errors after deleting or renaming a ColdFusion data source, you should 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.

CHAPTER 25
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 2004. This chapter describes how to create the database connection.

Note: If you're developing Microsoft Active Server Pages (ASP) applications, see Chapter 26, "Database Connections for ASP Developers," on page 469.

The chapter assumes you have set up an ASP.NET application (see Chapter 23, "Setting Up a Web Application," on page 453). It also assumes a database is set up on your local computer or on a system to which you have network or FTP access.

To connect to the sample database provided by Dreamweaver, see "Setup for Sample ASP.NET Site" in Getting Started Help. To learn more about databases and database connections, see Appendix A, "Beginner's Guide to Databases," on page 741.

Connecting to a database

This section describes how to connect to a database when developing an ASP.NET application in Dreamweaver.

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 465.

Once the database provider is installed, you can use it to connect to the database. For instructions, see the following sections:

- "Creating an ASP.NET database connection in Dreamweaver" on page 466
- "Creating a connection using Data Link Properties" on page 466

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 "Communicating with the database" on page 747.

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 can obtain an OLE DB provider for Microsoft Access by downloading and installing the Microsoft Data Access Components (MDAC) 2.7 package from the Microsoft website at www.microsoft.com/data/download.htm.

You can download OLE DB providers for Oracle 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 an ASP.NET 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 466.

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.

Related topics

- Chapter 29, "Troubleshooting Database Connections," on page 489

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 465), 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 461).

6 Click OK.

The new connection appears in the Databases panel.

Related topics

- Chapter 29, "Troubleshooting Database Connections," on page 489

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 an ASP.NET database connection in Dreamweaver" on page 466). 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: 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 an include file in the Connections subfolder in the site's local root folder. You can edit or delete the connection information in the file manually or as follows.

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 select 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 DataSet that uses the old connection name by double-clicking the recordset in the Bindings panel and selecting 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 select 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 DataSet that uses the old connection by double-clicking the name of the DataSet in the Bindings panel and selecting a new connection in the DataSet dialog box.

CHAPTER 26
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 2004. This chapter describes how to create the database connection.

Note: If you're developing ASP.NET applications, see Chapter 25, "Database Connections for ASP.NET Developers," on page 465.

The chapter assumes you have set up an ASP application (see Chapter 23, "Setting Up a Web Application," on page 453). It also assumes a database is set up on your local computer or on a system to which you have network or FTP access.

To connect to the sample database provided by Dreamweaver, see "Setup for Sample ASP Site" in Getting Started Help. To learn more about databases and database connections, see Appendix A, "Beginner's Guide to Databases," on page 741.

About database connections in ASP

An ASP application must connect to a database through an ODBC driver or an OLE DB provider. Creating an OLE DB connection can improve the speed of your connection.

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 "Communicating with the database" on page 747. 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 Jet Provider for Access (OLE DB)
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 myConnection, 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 471.

A connection string is a hand-coded expression that identifies the database and lists the information needed to connect to it. The following is an 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 474
- "Connecting to a database on an ISP" on page 475

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.

OLE DB connections

You can use an OLE DB provider to communicate with your database. Creating a direct database-specific OLE DB connection can improve the speed of your connection by eliminating the ODBC layer between your web application and the database.

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 2.7 packages on the Windows computer running IIS or PWS (if applicable). You can download the MDAC packages for free from the Microsoft website at www.microsoft.com/data/download.htm.

Note: Make sure you install MDAC 2.5 before installing MDAC 2.7.

You can download OLE DB providers for Oracle databases from the Oracle website at http://otn.oracle.com/software/tech/windows/ole_db/content.html (registration is required).

In Dreamweaver, you create an OLE DB connection by including a `Provider` parameter in a connection string (see "Creating a DSN-less connection" on page 474). 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.

Related topics

- "Creating a DSN-less connection" on page 474

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 "Using a DSN" on page 751.

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 "OLE DB connections" on page 470.

You can define the DSN on a local or remote Windows computer.

Related topics

- "Creating a DSN-less connection" on page 474

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 the following articles on the Microsoft website:

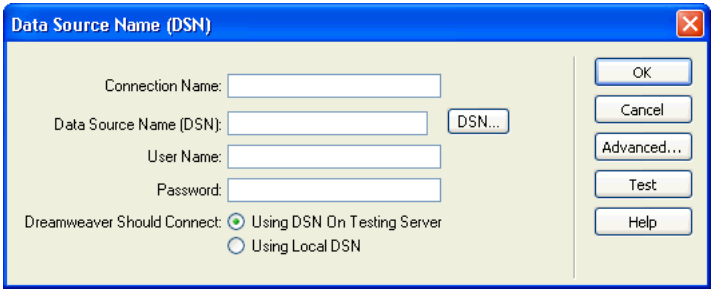
- If the computer runs Windows 98, see Microsoft Knowledge Base Article 300595 at http://support.microsoft.com/default.aspx?scid=kb;en-us;300595
- If the computer runs Windows 2000, see Microsoft Knowledge Base Article 300596 at http://support.microsoft.com/default.aspx?scid=kb;en-us;300596
- If the computer runs Windows XP, see Microsoft Knowledge Base Article 305599 at http://support.microsoft.com/default.aspx?scid=kb;en-us;305599

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 471.

7 If required, complete the User Name and Password text 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 506.

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 461).

10 Click OK.

The new connection appears in the Databases panel.

Related topics

- "Creating a DSN-less connection" on page 474
- Chapter 29, "Troubleshooting Database Connections," on page 489

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 the following articles on the Microsoft website:

- If the remote computer runs Windows 98, see Microsoft Knowledge Base Article 300595 at http://support.microsoft.com/default.aspx?scid=kb;en-us;300595
- If the remote computer runs Windows 2000, see Microsoft Knowledge Base Article 300596 at http://support.microsoft.com/default.aspx?scid=kb;en-us;300596
- If the remote computer runs Windows XP, see Microsoft Knowledge Base Article 305599 at http://support.microsoft.com/default.aspx?scid=kb;en-us;305599

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 select from the DSNs defined on it.

7 If required, complete the User Name and Password text 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 506.

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 461).

10 Click OK.

The new connection appears in the Databases panel.

Related topics

- "Creating a connection using a local DSN" on page 471
- Chapter 29, "Troubleshooting Database Connections," on page 489

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 this kind of connection. A connection string combines all the information your web application needs to connect to a database.

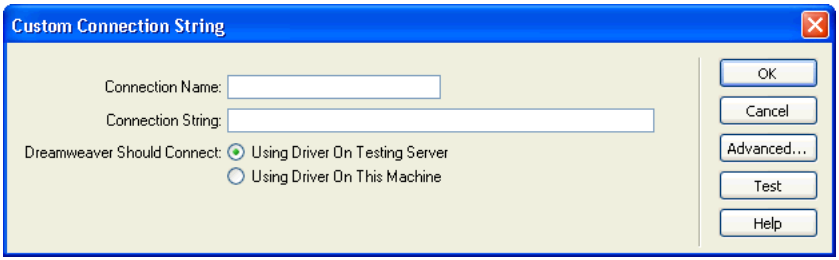
To create a DSN-less connection:

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.



3 Complete the dialog box and click OK.

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

Related topics

- "OLE DB connections" on page 470
- "Using a virtual path to connect to a database" on page 476
- Chapter 29, "Troubleshooting Database Connections," on page 489

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.

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.

Related topics

- "Using a virtual path to connect to a database" on page 476

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.

Related topics

- "Understanding physical and virtual paths" on page 475

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.

Related topics

- “Understanding physical and virtual paths” on page 475
- “Finding a file’s physical path with the virtual path” on page 475
- “Creating a DSN-less connection” on page 474

Editing or deleting a database connection

When you create a database connection, Dreamweaver stores the connection information in an include file in the Connections subfolder in the site's local root folder. You can edit or delete the connection information in the file manually or as follows.

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 select 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 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 select 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 27
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 2004. This chapter describes how to create the connection.

The chapter assumes you have set up a JSP application (see Chapter 23, "Setting Up a Web Application," on page 453). 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.

To connect to the sample database provided by Dreamweaver, see "Setup for Sample JSP Site" in Getting Started Help. To learn more about databases and database connections, see Appendix A, "Beginner's Guide to Databases," on page 741.

About database connections in JSP

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.

JSP connections

A JSP application must connect to a database through a JDBC driver. For more information on JDBC and the role of database drivers, see "Communicating with the database" on page 747.

You must specify certain parameter values to connect through your JDBC driver. For more information, see "JDBC connection parameters" on page 479. For the parameter values specific to your driver, see the driver vendor's documentation or consult your system administrator.

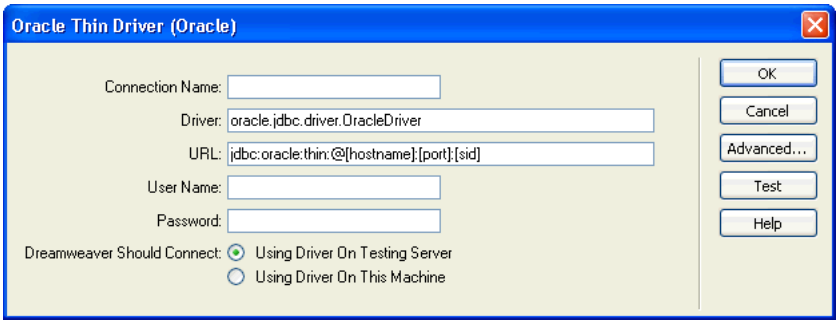
You can also use an ODBC driver (and so a Windows DSN) if you have a JDBC-ODBC Bridge driver. For more information, see "Connecting through an ODBC driver" on page 482.

JDBC connection parameters

When you create a JSP database connection in Dreamweaver, you usually need to enter JDBC connection parameters (see "Creating a database connection for JSP" on page 481). 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.

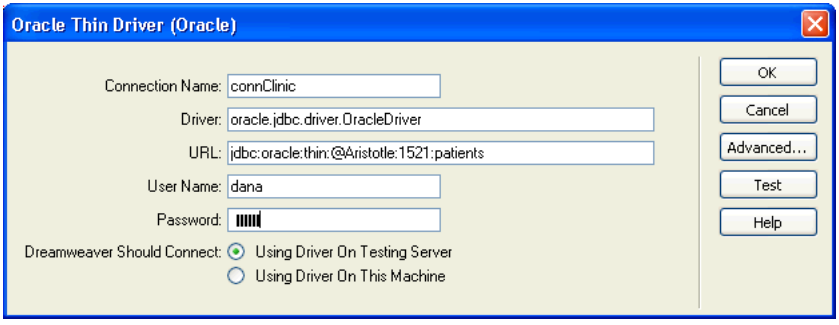
This section shows how to define connection parameters in Dreamweaver using the Oracle Thin JDBC driver as an example. For the connection parameters of other drivers, consult the driver vendor's documentation.

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 select 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 to a database

This section describes how to connect to a database when developing a JSP application in Dreamweaver.

The section assumes a JSP application server is running on a local or remote computer. For more information, see "Setting Up a Web Application" on page 453.

Before you can connect to a database, you must obtain a JDBC driver for your database. For more information, see "Obtaining a JDBC driver for your database" on page 481.

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 481.

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 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.

Related topics

- "Installing the Sun JDBC-ODBC Bridge driver" on page 483
- "Communicating with the database" on page 747

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 select your driver from the pop-up menu.
   If your driver is not listed, select Custom JDBC Connection.
   A connection dialog box appears.
3. Enter the connection parameters in the connection dialog box.
   For more information, see "JDBC connection parameters" on page 479.

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 461).
6 Click OK.

The new connection appears in the Databases panel.

Related topics

- Chapter 29, "Troubleshooting Database Connections," on page 489

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. The bridge driver acts as an interpreter between your JSP application, which "speaks" JDBC, and your ODBC driver, which "speaks" ODBC. This channel of communication lets your JSP application communicate with the database.

This kind of connection offers two advantages for Windows users. 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 483.
- 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 483.

If you meet these requirements, you can connect through an ODBC driver. For instructions, see "Creating an ODBC connection" on page 484.

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 an ODBC driver is installed, see "Viewing the ODBC drivers installed on a Windows system" on page 749. If an appropriate driver is not installed, you can download and install the Microsoft Data Access Components (MDAC) 2.5 and 2.7 packages on the computer running the JSP application server. You can download MDAC for free from the Microsoft website at www.microsoft.com/data/download.htm. These packages contain the latest ODBC drivers from Microsoft.

Note: Install the MDAC 2.5 package before installing MDAC 2.7.

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 any of the following folders: jdk1.2, jdk1.3, or j2sdk1.4.

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 (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 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 the following articles on the Microsoft website:

- Windows 98 users, see Microsoft Knowledge Base Article 300595 at http://support.microsoft.com/default.aspx?scid=kb;en-us;300595
- Windows 2000 users, see Microsoft Knowledge Base Article 300596 at http://support.microsoft.com/default.aspx?scid=kb;en-us;300596
- Windows XP users, see Microsoft Knowledge Base Article 305599 at http://support.microsoft.com/default.aspx?scid=kb;en-us;305599

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 select 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 461).

9 Click OK.

The new connection appears in the Databases panel.

Related topics

- Chapter 29, "Troubleshooting Database Connections," on page 489

Editing or deleting a database connection

When you create a database connection, Dreamweaver stores the connection information in an include file in the Connections subfolder in the site's local root folder. You can edit or delete the connection information in the file manually or as follows.

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 select 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, you should 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 select 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 28
Database Connections for PHP Developers

To use a database with a PHP application, you need to create a database connection in Macromedia Dreamweaver MX 2004. 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 www.mysql.com/downloads/mysql.html.

The chapter assumes you have set up a PHP application (see Chapter 23, "Setting Up a Web Application," on page 453). 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.

To connect to the sample database provided by Dreamweaver, see "Setup for Sample PHP Site" in Getting Started Help. To learn more about databases and database connections, see Appendix A, "Beginner's Guide to Databases," on page 741.

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 select 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 an include file in the Connections subfolder in the site's local root folder. You can edit or delete the connection information in the file manually or as follows.

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 select 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 select 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 selecting a new connection in the Recordset dialog box.

CHAPTER 29
Troubleshooting Database Connections

This chapter describes some common problems you might encounter after creating a database connection, and describes ways to fix them.

Troubleshooting permissions problems

One of the most common problems is insufficient folder or file permissions. If your database is located on a Windows 2000 or Windows XP computer and you receive an error message when you try to view a dynamic page in a web browser or in Live Data mode, the error may be due to a permissions problem.

Here are possible error messages pointing to a permissions problem:

- Microsoft OLE DB Provider for ODBC Drivers error '80004005' [Microsoft][ODBC Microsoft Access 97 Driver] Couldn't use '(unknown)'; file already in use.
- Microsoft OLE DB Provider for ODBC Drivers (0x80004005) [Microsoft][ODBC Microsoft Access Driver] The Microsoft Jet database engine cannot open the file '(unknown)'. It is already opened exclusively by another user, or you need permission to view its data.
- Microsoft OLE DB Provider for ODBC Drivers error '80004005' [Microsoft][ODBC Microsoft Access 97 Driver] Couldn't lock file.

The Windows account attempting to access the database doesn't have sufficient permissions. The account might be either the anonymous Windows account (by default, IUSR_*computername*) or a specific user account, if the page has been secured for authenticated access.

You must change the permissions to give the IUSR_*computername* account the correct permissions so the web server can access the database file. In addition, the folder containing the database file must also have certain permissions set to write to that database.

If the page is meant to be accessed anonymously, give the IUSR_*computername* account full control to the folder and database file, as described in the procedure below.

Additionally, if the path to the database is being referenced using UNC (\\Server\Share), make sure the Share Permissions give the IUSR_*computername* account full access. This step applies even if the share is on the local web server.

If you copy the database from another location, it might not automatically inherit the permissions from its destination folder. You might have to change the permissions for the database.

To check or change the database file permissions (Windows XP):

1 Make sure you have administrator privileges on the computer.

2 In Windows Explorer, locate the database file or the folder containing the database, right-click the file or folder, and select Properties.

3 Select the Security tab.

Note: This step applies only if you have an NTFS file system. If you have an FAT file system, the dialog box won't have a Security tab.

4 If the IUSR_*computername* account is not listed in the Group or User Names list, click the Add button to add it.

5 In the Select Users or Groups dialog box, click the Advanced button.

The dialog box changes to show more options.

6 Click the Locations button and select the computer's name.

7 Click the Find Now button.

A list of account names associated with the computer appears.

8 Select the IUSR_*computername* account and click OK; then click OK again to clear the dialog box.

9 To assign the IUSR account full permissions, select the Full Control checkbox and click OK.

To check or change the database file permissions (Windows 2000):

1 Make sure you have administrator privileges on the computer.

2 In Windows Explorer, locate the database file or the folder containing the database, right-click the file or folder, and select Properties.

3 Select the Security tab.

Note: This step applies only if you have an NTFS file system. If you have an FAT file system, the dialog box won't have a Security tab.

4 If the IUSR_*computername* account is not listed among the Windows accounts in the File Permissions dialog box, click the Add button to add it.

5 In the Select Users, Computers, or Groups dialog box, select the computer name from the Look In pop-up menu.

A list of account names associated with the computer appears.

6 Select the IUSR_*computername* account and click Add.

7 To assign the IUSR account full permissions, select Full Control from the Type of Access pop-up menu and click OK.

For added security, permissions can be set so that Read permission is turned off for the web folder containing the database. Browsing the folder won't be permitted, but web pages will still be able to access the database.

For more information about the IUSR account and web server permissions, see the following TechNotes on the Macromedia Support Center:

- Understanding anonymous authentication and the IUSR account at www.macromedia.com/go/authentication
- Setting IIS web server permissions at www.macromedia.com/go/server_permissions

Troubleshooting Microsoft error messages

This section describes some common Microsoft error messages and ways to troubleshoot them. These errors can occur if you use Internet Information Server (IIS) with a Microsoft database system such as Access or SQL Server. These errors normally occur when you request a dynamic page from the server.

Note: Macromedia does not provide technical support for third-party software such as Microsoft Windows, and IIS. If this section does not fix your problem, please contact Microsoft technical support or visit the Microsoft support website at http://support.microsoft.com/.

For more information on 80004005 errors, see "INFO: Troubleshooting Guide for 80004005 Errors in Active Server Pages and Microsoft Data Access Components (Q306518)," on the Microsoft website at http://support.microsoft.com/default.aspx?scid=kb;en-us;Q306518.

80004005–Data source name not found and no default driver specified

This error occurs when you attempt to view a dynamic page in a web browser or in Live Data mode. The error message may vary depending on your database and web server. Other variations of the error message include:

- 80004005—Driver's SQLSetConnectAttr failed
- 80004005—General error Unable to open registry key 'DriverId'

Here are possible causes and solutions:

- The page can't find the DSN. Make sure a DSN has been created on both the web server and on the local machine. For more information, see "Using a DSN" on page 751.
- The DSN might have been set up as a user DSN, not as a system DSN. Delete the user DSN and create a system DSN to replace it.

  Note: If you don't delete the user DSN, the duplicate DSN names produce a new ODBC error.

- If you use Microsoft Access, the database file (.mdb) might be locked. The lock might be due to a DSN with a different name accessing the database. In Windows Explorer, search for the lock file (.ldb) in the folder containing the database file (.mdb) and delete the .ldb file. If another DSN is pointing to the same database file, you may want to delete the DSN to prevent the error in the future. Be sure to reboot the computer after making these changes.

80004005–Couldn't use '(unknown)'; file already in use

This error occurs when you use a Microsoft Access database and attempt to view a dynamic page in a web browser or in Live Data mode. Another variation of this error message is "80004005—Microsoft Jet database engine cannot open the file (unknown)."

The probable cause is a permissions problem. For more information, see "Troubleshooting permissions problems" on page 489. Here are some specific causes and solutions:

- The account being used by Internet Information Server (usually IUSR) might not have the correct Windows permissions for a file-based database or for the folder containing the file. Check the permissions on the IIS account (IUSR) in the user manager.
- You might not have permission to create or destroy temporary files. Check the permissions on the file and the folder. Make sure you have permission to create or destroy any temporary files. Temporary files are usually created in the same folder as the database, but the file may also be created in other folders such as /Winnt.

- In Windows 2000, the time-out value may need to be changed for the Access database DSN. To change the time-out value, select Start > Settings > Control Panel > Administrative Tools > Data Sources (ODBC). Click the System tab, highlight the correct DSN, and click the Configure button. Click the Options button and change the Page Timeout value to **5000**.

If you still have problems, see the following Microsoft Knowledge Base articles:

- PRB: 80004005 "Couldn't Use '(unknown)'; File Already in Use" at http://support.microsoft.com/default.aspx?scid=kb;en-us;Q174943.
- PRB: Microsoft Access Database Connectivity Fails in Active Server Pages at http://support.microsoft.com/default.aspx?scid=kb;en-us;Q253604.
- PRB: Error "Cannot Open File Unknown" Using Access at http://support.microsoft.com/default.aspx?scid=kb;en-us;Q166029.

80004005–Logon Failed()

This error occurs when you use Microsoft SQL Server and attempt to view a dynamic page in a web browser or in Live Data mode.

This error is generated by SQL Server if it doesn't accept or recognize the logon account or password being submitted (if you're using standard security), or if a Windows account does not map to a SQL account (if you're using integrated security).

Here are possible solutions:

- If you use standard security, the account name and password might be incorrect. Try the system Admin account and password (UID= "sa" and no password), which must be defined in the connection string line. (DSNs do not store user names and passwords.)
- If you use integrated security, check the Windows account calling the page and find its mapped SQL account (if any).
- SQL Server does not allow an underscore in SQL account names. If someone manually maps the Windows IUSR_machinename account to a SQL account of the same name, it will fail. Map any account that uses an underscore to an account name on SQL that does not use an underscore.

80004005–Operation must use an updateable query

This error occurs when an event is updating a recordset or inserting data in a recordset.

Here are possible causes and solutions:

- The permissions set on the folder containing the database are too restrictive. IUSR privileges must be set to read/write. (See "Troubleshooting permissions problems" on page 489.)
- The permissions on the database file itself does not have full read/write privileges in effect. (See "Troubleshooting permissions problems" on page 489.)
- The database might be located outside the Inetpub/wwwroot directory. Though you can view and search the data, you might not be able to update it unless the database is located in the wwwroot directory.
- The recordset is based on a non-updateable query. Joins are good examples of non-updateable queries within a database. Restructure your queries so they are updateable.

For more information on this error, see "PRB: ASP 'Error The Query Is Not Updateable' When You Update Table Record," in the Microsoft Knowledge Base at http://support.microsoft.com/default.aspx?scid=kb;en-us;Q174640.

80040e07–Data type mismatch in criteria expression

This error occurs when the server tries to process a page containing an Insert Record or Update Record server behavior, and the server behavior attempts to set the value of a Date/Time column in a Microsoft Access database to an empty string ("").

Microsoft Access has strong data typing; it imposes a rigorous set of rules on given column values. The empty string value in the SQL query cannot be stored in an Access Date/Time column. Currently, the only known workaround is to avoid inserting or updating Date/Time columns in Access with empty strings ("") or with any other value that does not correspond to the range of values specified for the data type.

80040e10–Too few parameters

This error occurs when a column specified in your SQL query does not exist in the database table. Check the column names in your database table against the SQL query. The cause of this error is often a typographical error.

80040e10–COUNT field incorrect

This error occurs when you preview a page containing an Insert Record server behavior in a web browser and try to use it to insert a record in a Microsoft Access database.

You might be trying to insert a record into a database field that has a question mark (?) in its field name. The question mark is a special character for some database engines, including Microsoft Access, and should not be used for database table names or field names.

Open your database system and delete the question mark (?) from the field names, and update the server behaviors on your page that refer to this field.

80040e14–Syntax error in INSERT INTO statement

This error occurs when the server tries to process a page containing an Insert Record server behavior.

This error typically results from one or more of the following problems with the name of a field, object, or variable in the database:

- Using a reserved word as a name. Most databases have a set of reserved words. For example, "date" is a reserved word and cannot be used for column names in a database.
- Using special characters in the name. Examples of special characters include:

  . / * : ! # & - ?
- Using a space in the name.

The error can also occur when an input mask is defined for an object in the database, and the inserted data does not conform to the mask.

To fix the problem, avoid using reserved words such as "date", "name", "select", "where," and "level" when specifying column names in your database. Also, eliminate spaces and special characters.

See the following web pages for lists of reserved words for common database systems:

- Microsoft Access at http://support.microsoft.com/default.aspx?scid=kb;en-us;Q209187
- Microsoft SQL Server at http://msdn.microsoft.com/library/default.asp?url=/library/en-us/tsqlref/ts_ra-rz_9oj7.asp
- Oracle at http://technet.oracle.com/doc/server.804/a58234/vol2_wor.htm#421705 (registration required)
- MySQL at www.mysql.com/doc/R/e/Reserved_words.html

80040e21–ODBC error on Insert or Update

This error occurs when the server tries to process a page containing an Update Record or Insert Record server behavior. The database cannot handle the update or insert operation the server behavior is trying to perform.

Here are possible causes and solutions:

- The server behavior is trying to update a database table's auto-number field or to insert a record into an auto-number field. Because auto-number fields are populated automatically by the database system, any attempt to externally populate them with a value fails.
- The data the server behavior is updating or inserting is the wrong type for the database field, such as inserting a date into a Boolean (yes/no) field, inserting a string into a numeric field, or inserting an improperly formatted string into Date/Time field.

800a0bcd–Either BOF or EOF is true

This error occurs when you attempt to view a dynamic page in a web browser or in Live Data mode.

The problem occurs when the page tries to display data from an empty recordset. To solve the problem, apply the Show Region server behavior to the dynamic content to be displayed on the page, as follows:

1. Highlight the dynamic content on the page.
2. In the Server Behaviors panel, click the Plus (+) button and select Show Region > Show Region If Recordset Is Not Empty.
3. Select the recordset supplying the dynamic content and click OK.
4. Repeat steps 1 to 3 for each element of dynamic content on the page.

PART VII
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 30, “Optimizing the Workspace for Visual Development,” on page 497

Chapter 31, “The Workflow for Dynamic Page Design,” on page 507

Chapter 32, “Obtaining Data for Your Page,” on page 515

Chapter 33, “Defining Sources of Dynamic Content,” on page 525

Chapter 34, “Adding Dynamic Content to Web Pages,” on page 545

Chapter 35, “Displaying Database Records,” on page 555

Chapter 36, “Using ColdFusion Components,” on page 573

Chapter 37, “Using Web Services,” on page 579

Chapter 38, “Adding Custom Server Behaviors,” on page 589

Chapter 39, “Creating Forms,” on page 607

CHAPTER 30
Optimizing the Workspace for Visual Development

You can optimize the Macromedia Dreamweaver MX 2004 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 web-application development panels" on page 497
- "Viewing your database within Dreamweaver" on page 499
- "Viewing live data in Design view" on page 500
- "Working in Design view without live data" on page 505
- "Previewing dynamic pages in a browser" on page 505
- "Restricting database information displayed in Dreamweaver" on page 506

Displaying web-application development panels

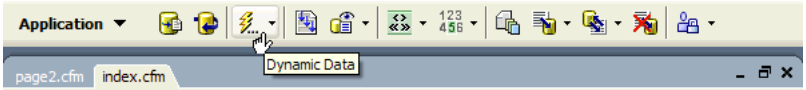
Click the Application tab in the Insert bar to display a set of buttons that let you add dynamic content and server behaviors to your page, as follows.

The number and type of buttons that appear vary depending upon the document type opened in the Document window. The Insert bar includes buttons to add the following items to the page:

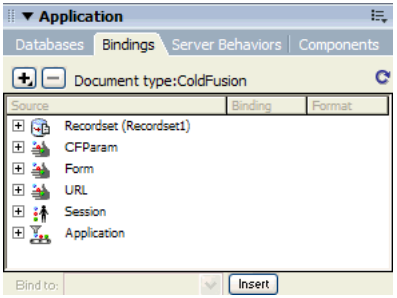
- Recordsets
- Dynamic text or tables
- Forms to insert or update records in a database
- Record navigation bars

If you switch to Code view (View > Code), additional panels might appear in their own Insert bar category, allowing you to insert code in the page. For example, if you view a ColdFusion page in Code view, a CFML panel becomes available in the CFML category of the Insert bar:

To find out what each button on the Insert bar does, move your mouse over an icon. A tooltip appears:

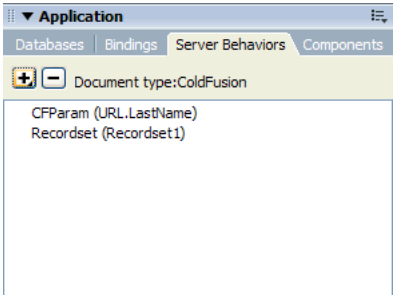


If you want to define sources of dynamic content for your page and add the content to the page, select Window > Bindings. The Bindings panel appears.



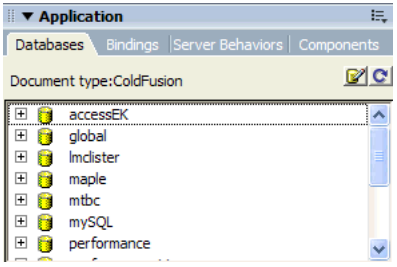
For more information, select Help from the panel group's pop-up menu.

If you want to add server-side logic to your dynamic pages, select Window > Server Behaviors. The Server Behaviors panel appears.



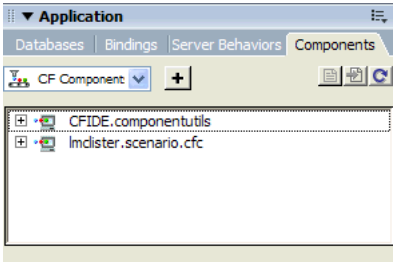
A server behavior is the set of instructions inserted in a dynamic page at design time and executed on the server at runtime. For more information, select Help from the panel group's pop-up menu.

If you want to explore databases or create database connections, select Window > Databases. The Databases panel appears.



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.



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 within Dreamweaver

After connecting to your database, you can view its structure and data within Dreamweaver.

To view the 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 461.)

If no database appears in the panel, you must create a database connection. For more information, see "Connecting to a database" on page 461.

2 To display the tables, stored procedures, and views in the database, click the Plus (+) sign 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 indicate 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 select View Data from the pop-up menu.

Dreamweaver also identifies the primary key of each table, and the data types of each column.

Viewing live data in Design view

Dreamweaver can display the dynamic content of a page while you work on the page in Design view. For more information on Design view, see "Switching between views in the Document window" on page 39.

To view dynamic content in Design view:

1 Make sure Dreamweaver is properly configured to display live data.

For more information, see "Requirements for displaying live data" on page 501.

2 Select View > Live Data.

Dreamweaver displays the page in Design view complete with dynamic content.

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 505.)

When you make a change to the page that affects 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.

The following illustration shows a dynamic page with Live Data turned off:

The following illustration shows the same page with Live Data turned on:

Related topics

- "Copying dependent files" on page 502
- "Providing the page with expected parameters" on page 503
- "Refreshing the page" on page 504
- "Troubleshooting Live Data view" on page 504
- "Working in Design view without live data" on page 505

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 461.

  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 "Connecting to a database" on page 461.
- Copy related files (if any) to the folder. For more information, see "Copying dependent files" on page 502.
- Provide the page with any parameters a user would normally provide. For instructions, see "Providing the page with expected parameters" on page 503.

If you have difficulty getting Live Data view to work, see "Troubleshooting Live Data view" on page 504.

Related topics

- "Viewing live data in Design view" on page 500
- "Working in Design view without live data" on page 505

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 461). 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) and 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 in 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.

Related topics

- "Viewing live data in Design view" on page 500
- "Requirements for displaying live data" on page 501
- "Refreshing the page" on page 504
- "Troubleshooting Live Data view" on page 504

Providing the page with expected parameters

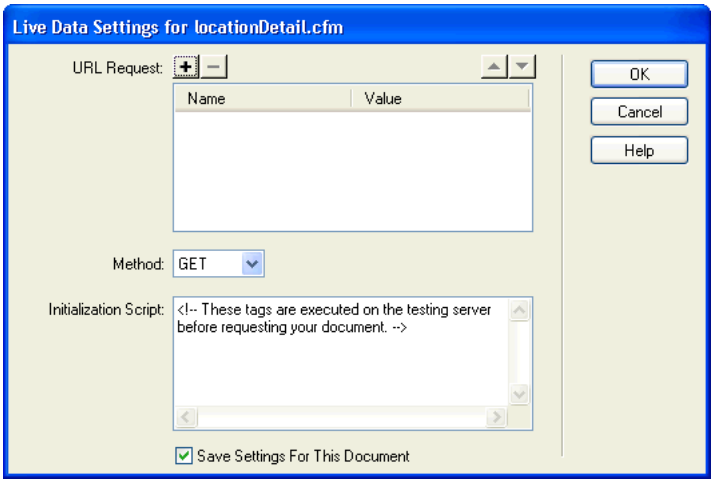
To generate dynamic content, some pages require parameters from the user—for example, a page needs the ID number of a record to find and display that record. Without that data, Dreamweaver cannot generate dynamic content to display in Design view.

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, select 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 "URL parameters" on page 518.

Related topics

- "Viewing live data in Design view" on page 500
- "Requirements for displaying live data" on page 501
- "Copying dependent files" on page 502
- "Troubleshooting Live Data view" on page 504

Refreshing the page

To refresh a 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 that affects dynamic content.

  You can also select the Auto Refresh option in 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.

Related topics

- "Viewing live data in Design view" on page 500
- "Requirements for displaying live data" on page 501
- "Copying dependent files" on page 502
- "Providing the page with expected parameters" on page 503

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 text box should specify a folder capable of processing dynamic pages (see "Specifying where dynamic pages can be processed" on page 461). 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 folder	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 "Connecting to a database" on page 461.

Related topics

- "Viewing live data in Design view" on page 500
- "Requirements for displaying live data" on page 501
- "Copying dependent files" on page 502
- "Providing the page with expected parameters" on page 503
- "Refreshing the page" on page 504

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. Select Edit > Preferences > Invisible Elements or Dreamweaver > Preferences > Invisible Elements (Mac OS X).
2. In the Show Dynamic Text As pop-up menu, select {}.
3. Click OK.

Related topics

- "Viewing live data in Design view" on page 500
- "Requirements for displaying live data" on page 501

Previewing dynamic pages in a browser

Web application developers often debug their pages by checking them 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 500.

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 461.

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, select Window > Site to open the Site panel, select the file under Local Folder, and click the blue up arrow in 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 and 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. 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 to an application other than ColdFusion:

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.
2. Right-click (Windows) or Control-click (Macintosh) the connection, and select 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 at the top of the panel and create it. For more information, see "Connecting to a database" on page 461.
3. In the dialog box for the connection, click Advanced.

   The Restrict dialog box appears.
4. Specify your schema or catalog.
5. Click OK.

CHAPTER 31
The Workflow for Dynamic Page Design

A key benefit provided by Macromedia Dreamweaver MX 2004 is the ability to create dynamic websites without having to be experienced in programming languages. The Dreamweaver visual tools let you develop dynamic websites without having to hand code the complex programmatic logic required to create a site that displays 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 as follows:

- "Designing the page" on page 508
- "Creating a source of dynamic content" on page 508
- "Adding dynamic content to a web page" on page 510
- "Enhancing the functionality of a dynamic page" on page 510
- "Testing and debugging the page" on page 511

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. You should 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 8, "Presenting Content with Tables," on page 163
- Chapter 9, "Laying Out Pages in Layout Mode," on page 183
- Chapter 12, "Working with Pages," on page 253
- Chapter 13, "Inserting and Formatting Text," on page 273
- Chapter 14, "Inserting Images," on page 303
- Chapter 15, "Linking and Navigation," on page 315
- Chapter 17, "Adding Audio, Video, and Interactive Elements," on page 355

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.

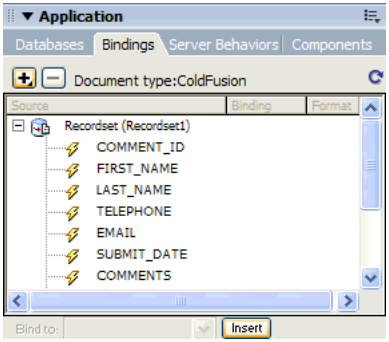
Before you can use these content sources in a web page, you must do the following:

- 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.

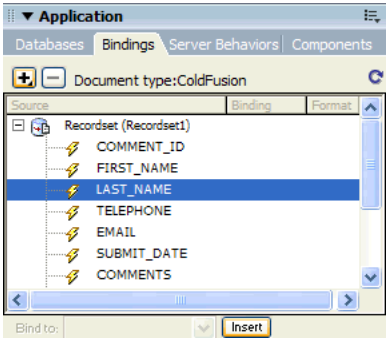
Different technology vendors may use different terminology for a recordset. In ASP and ColdFusion, a recordset is defined as a *query*. In JSP, a recordset is called a *resultset*. ASP.NET refers to a recordset as a *DataSet*. If you are using other sources of data, such as user input or server variables, 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 open the Recordset dialog box 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, “Obtaining Data for Your Page,” on page 515
- “Appendix A, “Beginner’s Guide to Databases,” on page 741,”

Adding dynamic content to a web page

After you define a recordset or other data source, and add it 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 as 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 do one of the following:
- Place it at the insertion point 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 the following chapters:

- Chapter 33, "Defining Sources of Dynamic Content," on page 525
- Chapter 39, "Creating Forms," on page 607

Enhancing the functionality of a dynamic page

In addition to adding dynamic content, Dreamweaver lets you easily incorporate complex application logic into web pages by 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 website 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 Application category of the Insert bar, or the Server Behaviors panel. To use the Server Behaviors panel, select Window > Server Behaviors, click the Plus (+) button on the panel, and select 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 tables 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.
- Display more than one record from a database record.
- Create recordset navigation links that allow users to view the previous or next records from a database record.
- Add a record counter to help users keep track of how many records where 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, "Adding Dynamic Content to Web Pages," on page 545
- Chapter 35, "Displaying Database Records," on page 555
- Chapter 40, "Building ColdFusion Applications Rapidly," on page 629
- Chapter 41, "Building ASP.NET Applications Rapidly," on page 675
- Chapter 42, "Building ASP and JSP Applications Rapidly," on page 703
- Chapter 43, "Building PHP Applications Rapidly," on page 719

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. For guidelines to help you test a website's usability and cross-platform compatibility, see "Site testing guidelines" on page 111. For more information about designing websites for people with auditory, visual, and other disabilities, see "Using Dreamweaver accessibility features" on page 49. You should become familiar with the issues described in this section and consider how your application's functionality might affect people with disabilities.

This section covers the following topics:

- "Testing dynamic content" on page 512
- "Editing dynamic content on a page" on page 513
- "Deleting dynamic content" on page 513

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 "Checking for browser compatibility" on page 434.)

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 selecting 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 in 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.

Related topics

- "Editing dynamic content on a page" on page 513
- "Deleting dynamic content" on page 513

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).

Related topics

- "Testing dynamic content" on page 512
- "Deleting dynamic content" on page 513

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.

Related topics

- "Testing dynamic content" on page 512
- "Editing dynamic content on a page" on page 513

CHAPTER 32
Obtaining 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 515
- "Collecting data submitted by users" on page 516
- "Accessing data stored in session variables" on page 520

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 having the website retrieve 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 site's 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 use to support your website, you can use Macromedia Dreamweaver MX 2004 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 741.

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 2004 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 753.

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.

Form parameters

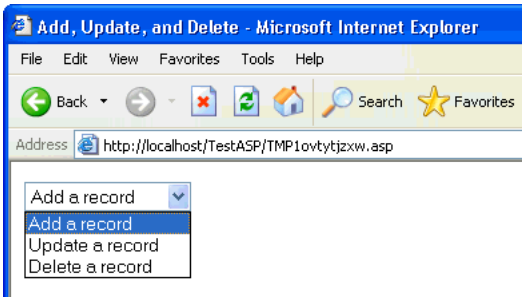
Form parameters are sent to the server using an HTML form using either the `POST` or `GET` method. When using the `POST` method, parameters are sent in the body of the message. In contrast, the `GET` method appends parameters to the requested URL.

You can use Dreamweaver to quickly design HTML forms that send form parameters to the server. For instructions, see Chapter 39, "Creating Forms," on page 607. Be aware of the method you use to transmit information from the browser to the server.

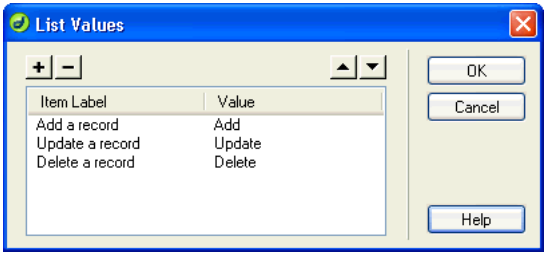
Form parameters take the names of their corresponding form objects. For example, if your form contains a text field named `txtLastName`, then the following form parameter is sent to the server when the user clicks the Submit button:

`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, checkbox, or list/menu form object to control the possible values the user can submit. This prevents users from typing information incorrectly 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):



Related topics

- "URL parameters" on page 518
- "Creating URL parameters using HTML links" on page 519
- "Retrieving form and URL parameters" on page 520

URL parameters

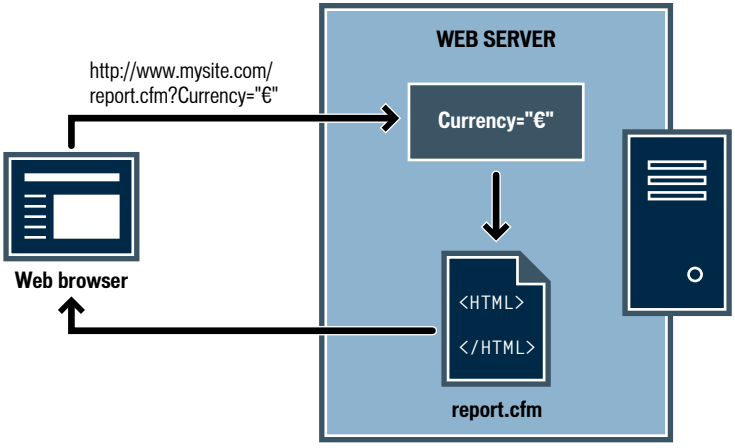
A URL parameter is a name/value pair appended to a URL. The parameter begins with a question mark (?) and takes the form `name=value`. If more than one URL parameter exists, each parameter is separated by an ampersand (&). The following example shows a URL parameter with two name/value pairs:

```
http://server/path/document?name1=value1&name2=value2
```

URL parameters let you 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 the developers of the site want to reach the widest possible audience, they have designed the site to support foreign currencies. When users log in to the site, they can select the 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 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 user name 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.

Related topics

- "Creating URL parameters using HTML links" on page 519
- "Retrieving form and URL parameters" on page 520

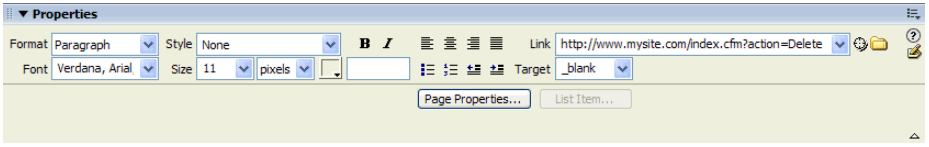
Creating URL parameters using HTML links

Creating 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), or by appending the URL parameters at the end of the link URL in the Property inspector Link text box.

In the following example, 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/index.cfm?action=Add">Add a record</a>
<a href="http://www.mysite.com/index.cfm?action=Update">Update a record</a>
<a href="http://www.mysite.com/index.cfm?action=Delete">Delete a record</a>
```

The Property inspector (Window > Properties) lets you create the same URL parameters by selecting the link, and appending the URL parameter values at the end of the link URL in the Link text box.



Related topics

- "Collecting data submitted by users" on page 516
- "URL parameters" on page 518
- "Retrieving form and URL parameters" on page 520
- "Understanding URL and form parameters" on page 528
- "Defining URL parameters" on page 534
- "Defining form parameters" on page 535

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 534.

After defining the form or URL parameter in Dreamweaver, you can insert its value within a page. For more information, see Chapter 34, "Adding Dynamic Content to Web Pages," on page 545.

Related topics

- "Accessing data stored in a database" on page 516
- "Form parameters" on page 517
- "URL parameters" on page 518
- "Creating URL parameters using HTML links" 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 application's pages for the duration of the user's visit. For example, when users log on to 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.

Understanding session variables

Web servers (or more specifically, 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 user name and password login to accomplish this. If multiple customized pages will be displayed, a way to keep track of which users are logged in is necessary, as most users would find it unacceptable to provide their user name and 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 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.

Related topics

- "Collecting information to store in session variables" on page 521
- "Storing information in session variables" on page 522
- "Example of information stored in session variables" on page 523
- "Retrieving data from session variables" on page 524

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.

Related topics

- "Understanding session variables" on page 520
- "Storing information in session variables" on page 522
- "Example of information stored in session variables" on page 523
- "Retrieving data from session variables" on page 524

Storing URL parameters in session variables

A typical example of storing URL parameters in session variables is a product catalog that 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 is stored in a session variable while the user continues to shop. When the user proceeds to the check-out page, the product ID stored in the session variable is retrieved.

Storing form parameters in session variables

A form-based survey is a typical example of a page that stores form parameters in session variables. The form sends the selected information back to the server, where an application page scores the survey and stores the responses in a session variable to be passed to an application that might tally up the responses gathered from the survey population. Or the information might be stored in a database for later use.

Storing information in session variables

After 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

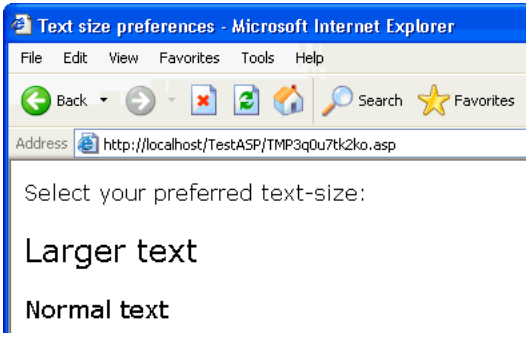
```
<% Session("prodID") = Request.Form("txtProduct") %>
```

Related topics

- "Understanding session variables" on page 520
- "Collecting information to store in session variables" on page 521
- "Example of information stored in session variables" on page 523
- "Retrieving data from session variables" on page 524

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.

Related topics

- “Understanding session variables” on page 520
- “Collecting information to store in session variables” on page 521
- “Storing information in session variables” on page 522
- “Retrieving data from session variables” on page 524

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 536.

After you define the session variable in Dreamweaver, you can insert its value in a page. For more information, see Chapter 34, “Adding Dynamic Content to Web Pages,” on page 545.

Related topics

- “Understanding session variables” on page 520
- “Collecting information to store in session variables” on page 521
- “Storing information in session variables” on page 522
- “Example of information stored in session variables” on page 523

CHAPTER 33
Defining Sources of Dynamic Content

Dynamic websites require a data source from which to retrieve and display dynamic content. Macromedia Dreamweaver MX 2004 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, see "HTML and web technologies resources" on page 29.

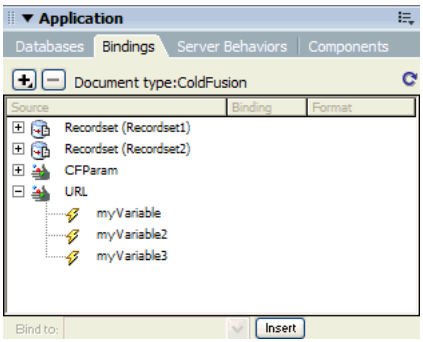
This chapter contains the following sections:

- "About dynamic content sources" on page 526
- "Defining URL parameters" on page 534
- "Defining session variables" on page 536
- "Defining application variables for ASP and ColdFusion" on page 537
- "Defining server variables" on page 538
- "Caching content sources" on page 542
- "Changing or deleting content sources" on page 543
- "Copying a recordset from one page to another page" on page 543

About 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. After you create a content source and include it in the Bindings panel, you can easily insert it into the currently selected page.



Related topics

- "Understanding recordsets" on page 526
- "Understanding URL and form parameters" on page 528
- "About session variables" on page 528
- "ASP and ColdFusion application variables" on page 529
- "ASP server variables" on page 529
- "ColdFusion server variables" on page 529
- "JavaBeans and JSP" on page 530

Understanding recordsets

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 Appendix A, "Beginner's Guide to Databases," on page 741 and Appendix B, "SQL Primer," on page 753.

If you will be writing SQL for use with ASP.NET, there are some conditions specific to ASP.NET that you should be aware of. To learn about these conditions, see "Writing SQL for ASP.NET" on page 527.

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 24, "Database Connections for ColdFusion Developers," on page 463
- Chapter 26, "Database Connections for ASP Developers," on page 469
- Chapter 25, "Database Connections for ASP.NET Developers," on page 465
- Chapter 27, "Database Connections for JSP Developers," on page 479
- Chapter 28, "Database Connections for PHP Developers," on page 487

Related topics

- "Defining a recordset" on page 530

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. These conditions are described in the next sections.

Parameters

The syntax you use to reference parameters varies depending on the database connection in use (for example, OLE DB or Microsoft SQL Server).

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 enclose the code in parentheses ( ).

```
SELECT * FROM Employees WHERE HireDate > "+ (Request.queryString("hireDate"))
```

Related topics:

- Chapter 25, "Database Connections for ASP.NET Developers," on page 465.

Understanding URL and form 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.

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 32, "Obtaining Data for Your Page," on page 515.

Related topics

- "Defining URL parameters" on page 534
- "Defining form parameters" on page 535

About session variables

Session variables let you 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 520.

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 "Collecting data submitted by users" on page 516 and "Accessing data stored in session variables" on page 520.

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.

Related topics

- "ASP and ColdFusion application variables" on page 529
- "Defining session variables" on page 536

ASP and ColdFusion 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.

For more information, see "Defining application variables for ASP and ColdFusion" on page 537 and "Adding Dynamic Content to Web Pages" on page 545

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`.

Related topics

- "Defining ASP server variables" on page 538

ColdFusion server variables

You can define the following ColdFusion server 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.

Related topics

- "Defining ColdFusion server variables" on page 539
- "Defining JSP server variables" on page 540
- "Defining PHP server variables" on page 540

JavaBeans and JSP

JavaBeans components are architectural elements of multitier 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 class path. (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 class path. (Your application server uses this copy of the class at runtime.) The application server's class path varies from application server to application server, but generally the class path is to a WEB-INF folder with a classes/bean subfolder.

If Dreamweaver and the application server are both running on the same system, and the application server uses the system class path (not an internal class path), a single copy of the JavaBeans class can reside on the computer in the system class path. 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 class path or in the Dreamweaver Configuration/classes folder.

Related topics

- "Defining JavaBeans (JSP only)" on page 540

Defining a recordset

You can define a recordset using either the Dreamweaver simple Recordset dialog box, which lets you easily construct simple SQL statements, or you can use the advanced Recordset dialog box, which lets you write your own SQL statements, or use the graphical Database Items tree to create a SQL statement.

If you are not familiar with writing SQL statements to retrieve information from a database, use the simple Recordset dialog box.

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 532.)

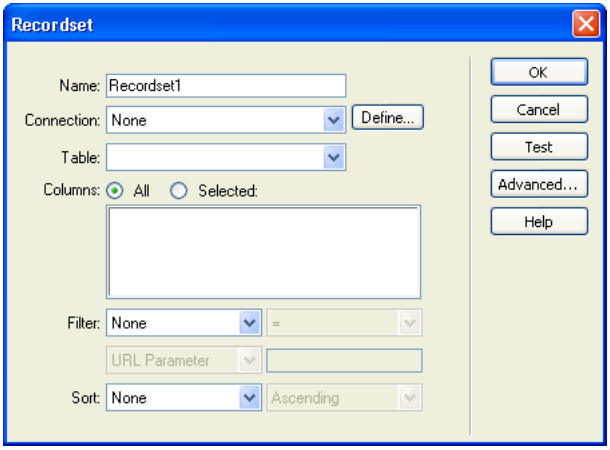
To define a recordset without writing SQL:

1 In the Document window, open the page that will use the recordset.

2 Select 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 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 Help button in the Recordset dialog box.

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 "Creating SQL queries using the Database Items tree" on page 533 for rules specific to ASP.NET.

If you want to create a simple recordset without manually writing SQL, see "Creating SQL queries using the Database Items tree" on page 533.

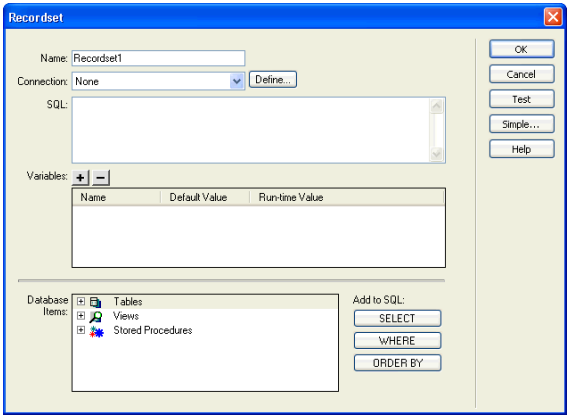
To define a recordset writing SQL:

1 In the Document window, open the page that will use the recordset.

2 Select 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 Help button in the Recordset dialog box.

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 the data from the recordset appears.

6 Click OK to add the recordset to the list of available content sources in the Bindings panel.

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. After 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 follows:

```
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 emplNo, emplName
FROM Employees
WHERE emplJob = 'varJob'
ORDER BY emplName
```

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 `emplNo` and click the Select button.
   - Select `emplName` and click the Select button.
   - Select `emplJob` and click the Where button.
   - Select `emplName` and click the Order By button.
3. Place the insertion point after `WHERE emplJob` 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. Before you begin, make sure you pass a form or URL parameter to the server. For more information, see "URL parameters" on page 518 and "Understanding URL and form parameters" on page 528."Understanding URL and form parameters" on page 528.

To define a URL parameter:

1 In the Document window, open the page that will use the variable.

2 Select Windows > Bindings to display the Bindings panel.

3 In the Bindings panel click the Plus (+) button and select one of the following from the pop-up menu:

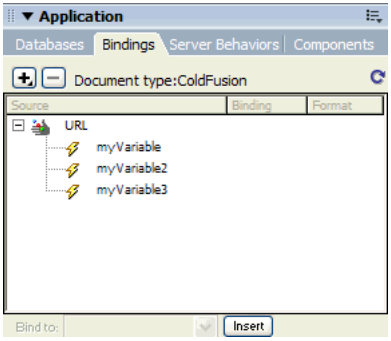
Document Types	Menu item in Bindings panel for URL variable
ASP	Request Variable › Request.QueryString
ColdFusion	URL Variable
JSP	Request Variable
PHP	URL Variable

The URL Variable dialog box is displayed.

4 Enter the name of the URL variable in the text box 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 34, "Adding Dynamic Content to Web Pages," on page 545.

Related topics:

- "Defining form parameters" on page 535
- "Defining session variables" on page 536

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 more information, see Chapter 32, "Obtaining Data for Your Page," on page 515 and

To define a form parameter:

1 In the Document window, open the page that will use the variable.

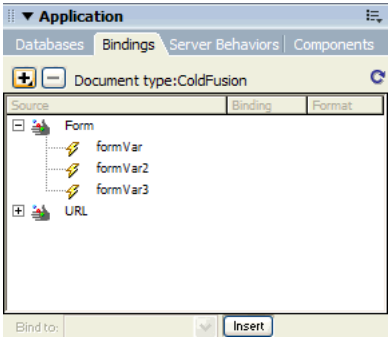
2 Select Windows > Bindings to display the Bindings panel.

3 In the Bindings panel click the Plus (+) button and select 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. 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.

4 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 32, "Obtaining Data for Your Page," on page 515.

Related topics

- "About dynamic content sources" on page 526
- "Understanding URL and form parameters" on page 528
- "Defining URL parameters" on page 534

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 "Understanding session variables" on page 520.

Before defining session variables for a page, you must create them in the source code. For instructions, see the following topics:

- "Collecting data submitted by users" on page 516
- "Accessing data stored in session variables" on page 520
- "Storing information in session variables" on page 522

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 ColdFusion example instantiates a session called `username`, and assigns it the value `Cornelius`:

```
<CFSET session.username = Cornelius>
```

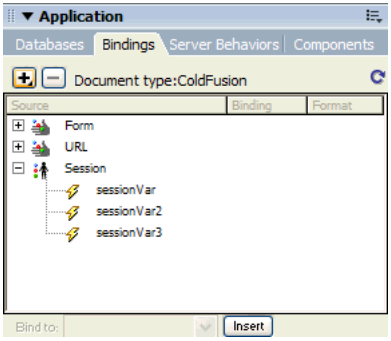
2 Select Window > Bindings to display the Bindings panel.

3 Click the Plus (+) button and select Session Variable from the pop-up menu.

4 Enter the name of the variable you defined in the source code.

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 34, "Adding Dynamic Content to Web Pages," on page 545.

Related topics

- "Collecting data submitted by users" on page 516
- "Accessing data stored in session variables" on page 520
- "Understanding session variables" on page 520
- "Storing information in session variables" on page 522

Defining application variables for ASP and ColdFusion

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.

Note: There are no application variable objects in JSP or PHP.

To define an application variable for a page:

1 Open a dynamic document type in the Document window.

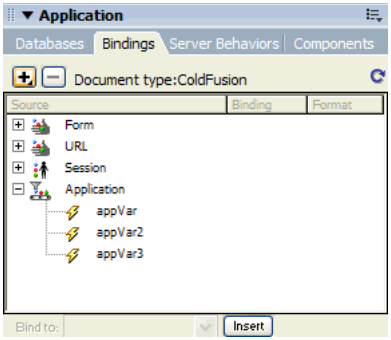
2 Select Window > Bindings to display the Bindings panel.

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 under the Application icon.



Once you define the application variable, you can use its value in a page. For more information, see Chapter 34, "Adding Dynamic Content to Web Pages," on page 545.

Related topics:

- "ASP and ColdFusion application variables" on page 529

Using a variable as a data source for a ColdFusion recordset

When you define a recordset for a page in the Bindings panel, Dreamweaver enters the name of the ColdFusion data source in the `cfquery` tag on the page. For more flexibility, you can store a data source name in a variable and use the variable in the `cfquery` tag. Dreamweaver provides a visual method of specifying such a variable in your recordsets.

To visually specify a variable as a data source for a recordset:

1. Make sure a ColdFusion page is active in the Document window.
2. In the Bindings panel, click the Plus (+) button and select Data Source Name Variable from the pop-up menu.

   The Data Source Name Variable dialog box appears.
3. Define a variable and click OK.
4. When defining the recordset, select the variable as the Data Source for the recordset.

   In the Recordset dialog box, the variable appears in the Data Source pop-up menu along with the ColdFusion data sources on the server.
5. Complete the Recordset dialog box and click OK.
6. Initialize the variable.

   Dreamweaver does not initialize the variable for you so that you can initialize it how and where you want. You can initialize the variable in the page code (before the `cfquery` tag), in an include file, or in some other file as a session or application variable.

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 534
- “Defining form parameters” on page 535
- “Defining session variables” on page 536
- “Defining application variables for ASP and ColdFusion” on page 537

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 ColdFusion server variables” on page 539
- “Defining JSP server variables” on page 540
- “Defining JavaBeans (JSP only)” on page 540

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 panel (Window > Bindings).

2 Click the Plus (+) button and select Request Variable from the pop-up menu.

The Request Variable dialog box is displayed. Select 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, select Server Variables. If you want to access the information in the `Request.Form` collection, select Form.

For more information on the ASP server variables, click the Help button.

3 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`.

4 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. To learn more about these server variables, see "ASP server variables" on page 529.

To define a server variable for a ColdFusion page:

1 Open the Bindings panel (Window > Bindings).

2 Click the Plus (+) button and select 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.

Related topics:

- "Defining URL parameters" on page 534
- "Defining session variables" on page 536
- "Defining application variables for ASP and ColdFusion" on page 537
- Chapter 34, "Adding Dynamic Content to Web Pages," on page 545

Defining PHP server variables

You can define a PHP server variable as a source of dynamic content for PHP document types.

To define a server variable for a PHP page:

1 Open the Bindings panel (Window > Bindings).

2 Click the Plus (+) button and select 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.

Related topics:

- Chapter 33, "Defining URL parameters," on page 534
- Chapter 34, "Adding Dynamic Content to Web Pages," on page 545

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 select 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.

Related topics:

- Chapter 33, "Defining URL parameters," on page 534
- Chapter 34, "Adding Dynamic Content to Web Pages," on page 545

Defining JavaBeans (JSP only)

In Dreamweaver, JavaBeans components 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 class path. (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 class path. (Your application server uses this copy of the class at runtime.) The application server's class path varies from application server to application server, but generally the class path is to a WEB-INF folder with a classes/bean subfolder.

If Dreamweaver and the application server are both running on the same system, and the application server uses the system class path (not an internal class path), a single copy of the JavaBeans class can reside on the computer in the system class path. 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).

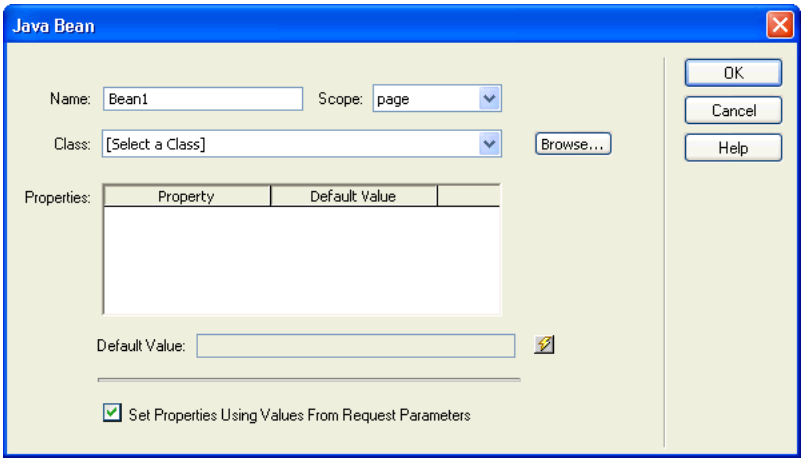
The folder structure must match the JavaBeans' package. For example, if the JavaBeans' package is called `com.ardvark.myBean`, then the package must be stored in /com/ardvark/ within the class path or in the Dreamweaver Configuration/classes folder.

To define a JavaBean for a JSP page:

1 Select Window > Bindings to display the Bindings panel.

2 Click the Plus (+) button and select 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 Help button in the dialog box.

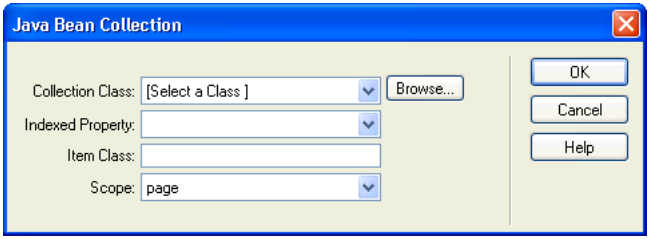
4 The newly defined bean appears in the Bindings panel.

To define a JavaBean collection for a JSP page:

1 Select Window > Bindings to display the Bindings panel.

2 Click the Plus (+) button and select 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.

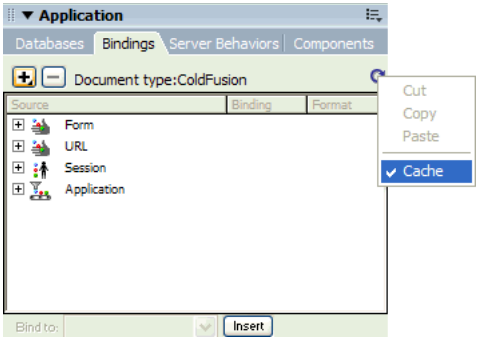
Caching content sources

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 "Editing dynamic content" on page 552 and "Deleting dynamic content" on page 552.

To change a content source in the Bindings panel:

1. In the Bindings panel (Window > Bindings), double-click the name of the content source you want to edit.
2. Make your changes in the dialog box that appears.
3. If satisfied with your work, click OK.

To delete a content source from the Bindings panel:

1. In the Bindings panel (Window > Bindings), select the content source from the list.
2. Click the Minus (-) button.

Related topics

- "About dynamic content sources" on page 526

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 select 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 select Paste from the pop-up menu.

Related topics:

- "About dynamic content sources" on page 526
- "Changing or deleting content sources" on page 543

CHAPTER 34
Adding Dynamic Content to Web Pages

This chapter describes the most efficient ways of making various page elements dynamic. Once 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 33, "Defining Sources of Dynamic Content," on page 525.

In Macromedia Dreamweaver MX 2004, 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, replace a text string, or insert it as 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:

- "About adding dynamic content" on page 546
- "Making text dynamic" on page 547
- "Making images dynamic" on page 548
- "Making HTML attributes dynamic" on page 549
- "Making ActiveX, Flash, and other object parameters dynamic" on page 551
- "Editing dynamic content" on page 552
- "Deleting dynamic content" on page 552
- "Creating dynamic pages in a Contribute site" on page 553

About adding dynamic content

You can add dynamic content to a page by selecting a content source in the Bindings panel. Dreamweaver inserts a server-side script in the page's code instructing the server to transfer the data from the content source to the page's HTML 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.

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 563), or you can create a repeated region to display more than one record on a single page (see "Displaying multiple recordset results" on page 566).

Dynamic text

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. (See "Applying typographic and page layout elements to dynamic data" on page 556.)

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. (See "Using predefined data formats" on page 562.)

You can replace regular text on your page with dynamic text, or you can add dynamic text at the insertion point on the page.

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)
```

Related topics:

- "Editing dynamic content" on page 552
- "Deleting dynamic content" on page 552

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.

To add dynamic text:

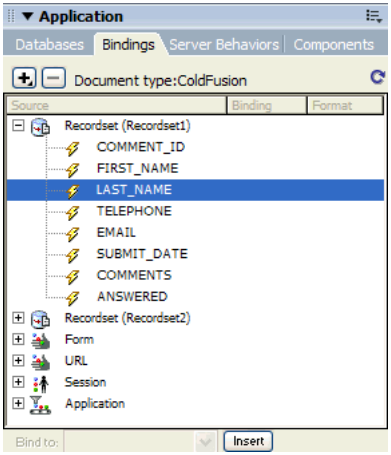
1 Open the Bindings panel by choosing Window > Bindings.

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 33, "Defining Sources of Dynamic Content," on page 525.)

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 content source 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 `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 the Document window. You can solve the problem by using empty curly braces as placeholders.

To create dynamic text placeholders:

1 Select Edit > Preferences > Invisible Elements (Windows) or Dreamweaver > Preferences > Invisible Elements (Macintosh).

2 In the Show Dynamic Text As pop-up menu, select { }.

3 Click OK.

Related topics

- "About adding dynamic content" on page 546
- "Dynamic text" on page 546

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:

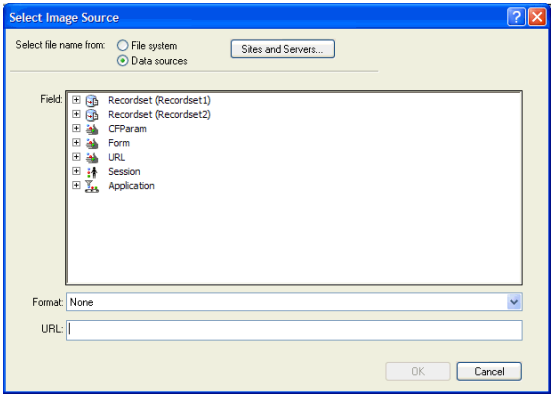
1 With the page open in Design view (View > Design), place the insertion point where you want the image to appear on the page.

2 Select Insert > Image.

The Select Image Source dialog box appears.

3 Click the Data Sources option (Windows) or the Data Source button (Macintosh).

A list of content sources appears.



4 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 "Understanding recordsets" on page 526.

5 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.

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 33, "Defining Sources of Dynamic Content," on page 525.

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 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 How you bind a dynamic content source to the HTML attribute depends on where it is located:

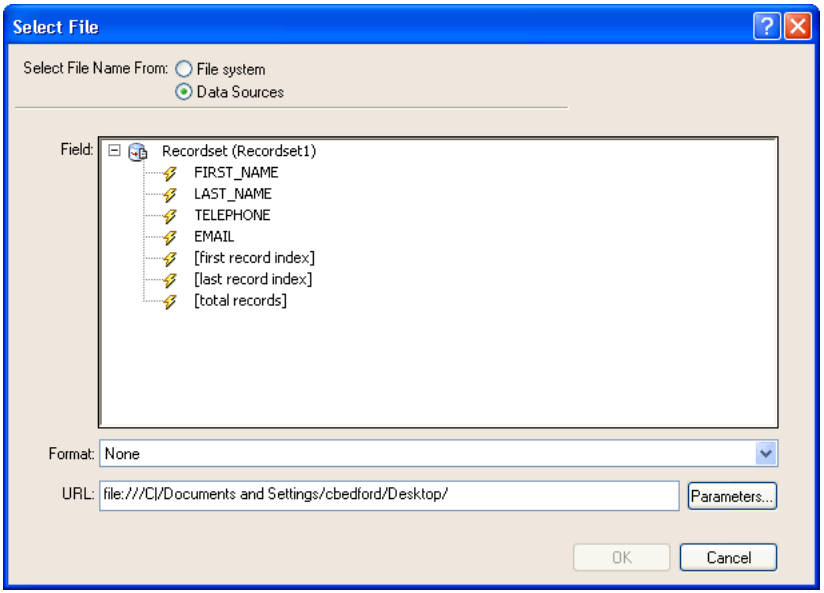
- If the attribute you want to bind has a folder icon next to it in the Property inspector. Click the folder icon to open a file selection dialog box; then click the Data Sources option to display a list of data sources.
- If the attribute you want to bind does not have a folder icon next to it, click the List tab (the lower of the two tabs) on the left side of the inspector.

  The Property inspector's List view appears.
- 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.

3 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. Select the Data Sources option to display a list of content sources.

4 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 32, "Obtaining Data for Your Page," on page 515.

5 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, Macromedia Flash, Shockwave, Director, and Generator objects.

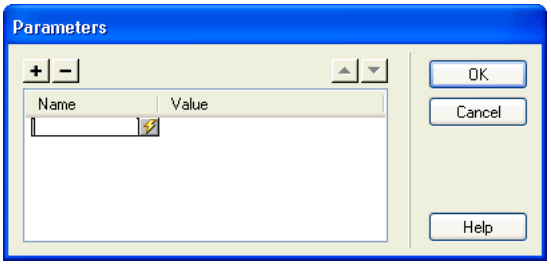
Before starting, make sure the fields in your recordset hold data that's appropriate for the object parameters you want to bind.

To make object parameters dynamic:

1 In Design view, select an object on the page and open the Property inspector (Window > Properties).

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, and 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 33, "Defining Sources of Dynamic Content," on page 525.

6 Click OK.

Editing dynamic content

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).

If you edit a recordset in the Live Data window when the Auto Refresh option is not selected, you must refresh the page to see your changes. To refresh the page, click the Refresh button or select View > Refresh Live Data.

Related topics

- "Editing dynamic content" on page 552

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.

Related topics

- "Editing dynamic content" on page 552
- "Editing dynamic content" on page 552

Creating dynamic pages in a Contribute site

When a Contribute user edits a page containing dynamic content or invisible elements (such as scripts and comments), Contribute displays the dynamic content and invisible elements as yellow markers. By default, Contribute users can't select or delete these markers.

If you want Contribute users to be able to select and delete dynamic content and other invisible elements from a page, you can change permission-group settings to allow them to do so; Contribute users normally can never edit dynamic content, even when you allow them to select it.

Note: Using some server technologies, you can display static text using a server tag or function. To allow Contribute users to edit the static text in a dynamic page that uses such a server technology, place the static text outside of the server tags.

To allow a group of Contribute users to delete code from pages:

1 Select Site > Administer Contribute Site.

2 If certain required options for Contribute compatibility aren't enabled, a dialog box appears, asking if you want to enable those options. Click OK to enable those options and Contribute compatibility. For more information, see "Preparing a site for use with Contribute" on page 138.

3 Do one of the following:

- If the Administrator Password dialog box appears, type the administrator password and click OK.
- If the site has no administrator, a dialog box appears asking if you want to become the website's administrator. Click Yes, then type and confirm an administrator password for the website, and click OK.

The Administer Website dialog box appears.

4 Select a permission group and click the Edit Group button.

5 Select the Editing category and deselect the option to protect scripts and forms. For more information, see "Creating templates for a Contribute site" on page 229, and *Administering Macromedia Contribute*.

6 Click OK to close the Permission Group dialog box.

7 Click OK to close the Administer Website dialog box.

Related topics

- "Managing Contribute Sites with Dreamweaver" on page 133
- "Creating templates for a Contribute site" on page 229

CHAPTER 35
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 2004 provides many methods of displaying dynamic content, and provides several built-in server behaviors that let you both enhance the presentation of dynamic content, and allow users to more easily search through and navigate information returned from a database.

This chapter contains the following sections:

- "About displaying database records" on page 555
- "Using predefined data formats" on page 562
- "Customizing existing data formats" on page 562
- "Creating recordset navigation links" on page 563
- "Showing and hiding regions based on recordset results" on page 566
- "Displaying multiple recordset results" on page 566
- "Creating a table with a Repeat Region server behavior" on page 567
- "Creating a record counter" on page 569
- "Creating ASP.NET DataGrid and DataList web controls" on page 570
- "Creating a record counter" on page 569

About displaying database records

Databases and other sources of dynamic content provide you with more power and flexibility in searching, sorting, and viewing large stores of information. Using a database to store content for web sites makes sense when you need to store large amounts of information, and then retrieve and display that information in a meaningful way. Dreamweaver provides you with several tools and prebuilt behaviors to help you effectively retrieve and display information stored in a database. The following sections describe the Dreamweaver server behaviors and formatting elements, and how you can use them to display dynamic content.

Server behaviors and formatting elements

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.

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 the Dreamweaver "Currency - 2 Decimal Places" format. This format 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.

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 let 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.

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 the Dreamweaver 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 formatting features, and how to apply them to dynamic data elements, see Chapter 8, "Presenting Content with Tables," on page 163 and Chapter 13, "Inserting and Formatting Text," on page 273.

Navigating database recordset results

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.

Related topics

- "Creating recordset navigation links" on page 563

Custom recordset navigation bars

If you want to create a recordset navigation bar that uses more complex layout and formatting styles than the simple table created by the Recordset Navigation Bar server object, you might 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.

You can assign the following individual server behaviors to navigation links:

- Move to first page
- Move to last page
- Move to next page
- Move to previous page

If you would prefer to use the Dreamweaver 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 563.

Navigation bar design tasks

When creating a custom navigation bar, begin by creating its visual representation using the Dreamweaver 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. For more information, see "Understanding recordsets" on page 526.

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 previous page	Move to previous page
Go to next page	Move to next page
Go to last page	Move to last page

Related topics

- "Displaying multiple recordset results" on page 558

Displaying 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:

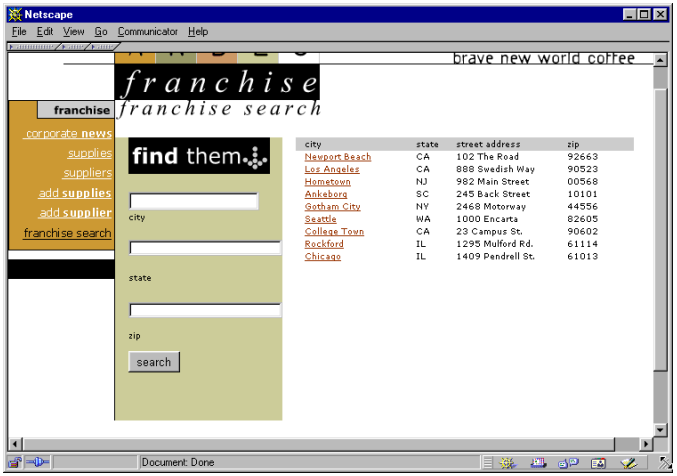
- Show If Recordset Is Empty
- Show If Recordset Is Not Empty
- Show If First Page
- Show If Not First Page
- Show If Last Page
- Show If Not Last Page

To learn how to apply the Show Region server behaviors, see "Showing and hiding regions based on recordset results" on page 566.

Displaying multiple recordset results

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 following example illustrates how the Repeat Region server behavior is applied to a table row, and specifies that nine records are 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.

To learn how to apply the Repeat Region server behavior, see "Displaying multiple recordset results" on page 566 and "Creating a table with a Repeat Region server behavior" on page 567.

Record counters

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:

- "Understanding recordsets" on page 526
- "Creating recordset navigation links" on page 563
- "Displaying multiple recordset results" on page 566
- "Creating a table with a Repeat Region server behavior" on page 567

Once you have the above elements in the page, you can create a record counter.

Simple record counters

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 page-design tools. To learn more about this record counter, see "Creating a record counter using the Recordset Navigation Status object" on page 569.

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.

The Record Count server behaviors are:

- Display Starting Record Number
- Display Ending Record Number
- Display Total Records

"Creating custom record counters" on page 569 leads you through the steps to create a record counter by applying the individual Record Counter server behaviors to a page's design elements.

About the ASP.NET DataGrid and DataList controls

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 lay out 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.

Related topics

- "Creating ASP.NET DataGrid and DataList web controls" on page 570
- "Adding an ASP.NET DataGrid to a page" on page 570
- "Adding an ASP.NET DataList to your page" on page 571
- "Modifying a DataGrid or DataList object" on page 572

The ASP.NET DataGrid web control

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 shown in the following table:

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 type 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.
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 Button	The Delete Button lets a user delete a particular row by clicking a button.

Before using 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 "Understanding recordsets" on page 526.

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/library/default.asp?url=/library/en-us/cpgenref/html/cpcondatagridwebcontrol.asp
- Microsoft ASP.NET at: http://asp.net

Related topics

- "Adding an ASP.NET DataGrid to a page" on page 570
- "Adding an ASP.NET DataList to your page" on page 571
- "Modifying a DataGrid or DataList object" on page 572

Using predefined data formats

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.

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 Select 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.

Related topics

- "Creating new data formats" on page 563

Customizing existing data formats

You can customize the Dreamweaver 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 Select 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.

Related topics

- "Using predefined data formats" on page 562
- "Creating new data formats" on page 563

Creating new data formats

You can create new data formats to suit any type of dynamic data you 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 Select 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.

Related topics

- "Customizing existing data formats" on page 562
- "Creating new data formats" on page 563

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.

You can create recordset navigation links either by using the Recordset Navigation Bar server behavior or you can create a custom recordset navigation bar.

Related topics

- "Navigating database recordset results" on page 556
- "Custom recordset navigation bars" on page 557
- "Creating a custom recordset navigation bar" on page 565

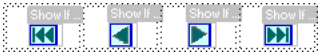
Creating a navigation bar using the Recordset Navigation Bar server behavior

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 the Dreamweaver 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.

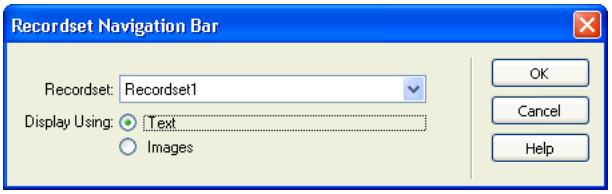
If you want to build the navigation bar block by block using the Dreamweaver design tools and the Server Behaviors panel, see "Creating a custom recordset navigation bar" on page 565.

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.

Related topics

- "Creating a custom recordset navigation bar" on page 565

Creating a custom recordset navigation bar

You can create a custom recordset navigation bar that uses more complex layout and formatting styles than that offered by the simple table used by the Recordset Navigation Bar server object.

To create your own recordset navigation bar, you must:

- Design navigation links using either text or images
- Place the links in the page in design view
- Assign individual server behaviors to each navigation link

To learn more about designing a custom recordset navigation bar, see "Custom recordset navigation bars" on page 557. If you prefer to use the Dreamweaver 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 563.

This procedure describes how to assign individual server behaviors to the navigation links.

To assign server behaviors to recordset 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. Select Recordset Paging from the pop-up menu; then select a server behavior appropriate to that link from the listed server behaviors.

   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.

   The server behavior is assigned to the navigation link.

Related topics

- "Navigating database recordset results" on page 556
- "Custom recordset navigation bars" on page 557

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. To learn more about the Show Region server behaviors, and how they can be used to show or hide recordset results, see "Displaying and hiding regions based on recordset results" on page 558.

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 Select Show Region from the pop-up menu, and then select one of the listed server behaviors.

4 Click OK.

Related topics

- "About displaying database records" on page 555
- "Creating a table with a Repeat Region server behavior" on page 567

Displaying multiple recordset results

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 tables, table rows, or a series of table rows.

To learn more about using the Repeat Region server behavior, see "Displaying multiple recordset results" on page 558 and "Creating a table with a Repeat Region server behavior" on page 567.

To create a repeated region:

1 In Design view, select a region that contains dynamic content.

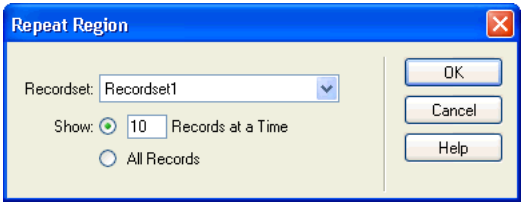
The selection can be anything, including a table, a table row, or even a paragraph of text.

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 Select 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.

Related topics

- "About displaying database records" on page 555
- "Showing and hiding regions based on recordset results" on page 566

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.

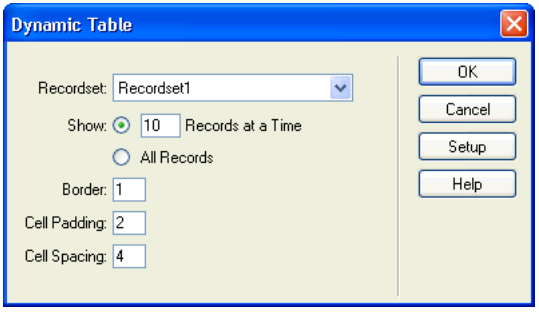
Note: The Dynamic Table server object is not available when using ASP.NET document types. To create a table containing dynamic content and repeating regions, you must manually insert the table, insert dynamic content from the Bindings panel, and apply a Repeat Region server behavior if needed. For more information, see "Displaying multiple recordset results" on page 566.

To create a dynamic table:

1 To insert a dynamic table:

- Select Insert > Application Objects > Dynamic Data > Dynamic Table to display the Dynamic Table dialog box.
- From the Application category of the Insert bar, select the Dynamic Table button in the Dynamic Data menu.

The Dynamic Table dialog box appears.



2 Select the recordset you want to use from the Recordset pop-up menu.

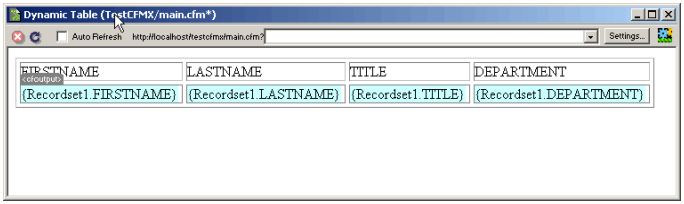
3 Select the number of records to display per page.

4 Input values for the table border, cell padding, and cell spacing if desired.

The Dynamic Table dialog box 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 let users know where they are within a given set of records relative to the total number of records returned. For this reason record counters are a useful behavior that can significantly add to the usability of a web page.

To learn more about record counters, see "Record counters" on page 559.

Creating a record counter using the Recordset Navigation Status object

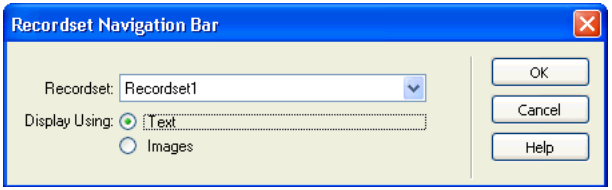
The Recordset Navigation Status object creates a text entry on the page to display the current record status.

To use the Recordset Navigation Status server object:

1 Place the insertion point where you want to insert the record counter.

2 Select 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 the Dreamweaver 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

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 "Record counters" on page 559. After you have created 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 569. The record counter in this example will appear as follows:

Displaying records `StartRow` thru `EndRow` of `RecordSet.RecordCount`.

In this example, the text in sans-serif font represents the record count placeholders that will be inserted in the page.

To create a custom record counter:

1 In Design view, enter the counter's text on the page. The text can be anything you want. For example:

```
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 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 now look similar to the following example:

```
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 the Preview in Browser feature in Dreamweaver. Make sure the Preview Using Live Data Server option is selected in Preferences (Edit > Preferences > Preview in Browser (Windows) or Dreamweaver > Preferences > Preview in Browser (Macintosh)); then select File > Preview in Browser.

Creating ASP.NET DataGrid and DataList web controls

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. To learn more about ASP.NET DataGrid and DataList controls, see "About the ASP.NET DataGrid and DataList controls" on page 560.

Adding an ASP.NET DataGrid to a page

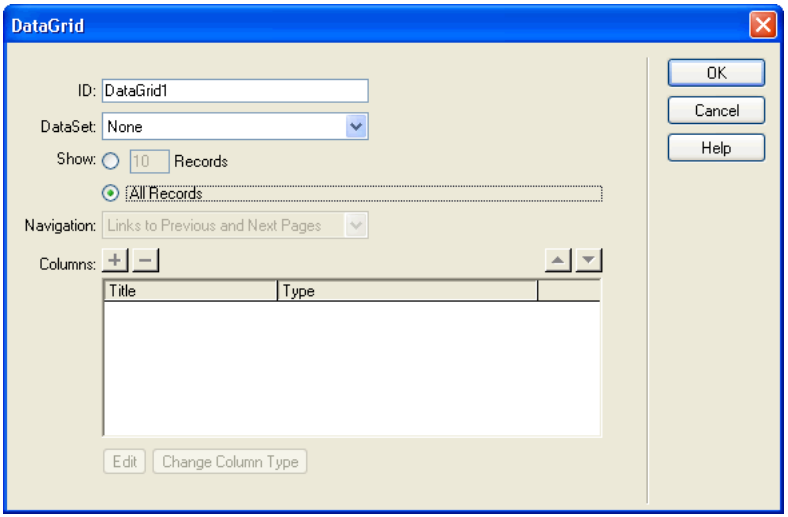
DataGrid controls let you format and display dynamic data in multi-column grids that are rendered as tables. For more information on DataGrids, and the types of formatting control they provide, see "About the ASP.NET DataGrid and DataList controls" on page 560.

Note: 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 "Understanding recordsets" on page 526.

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 is displayed 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.

Related topics

- "About the ASP.NET DataGrid and DataList controls" on page 560
- "Modifying a DataGrid or DataList object" on page 572

Adding an ASP.NET DataList to your page

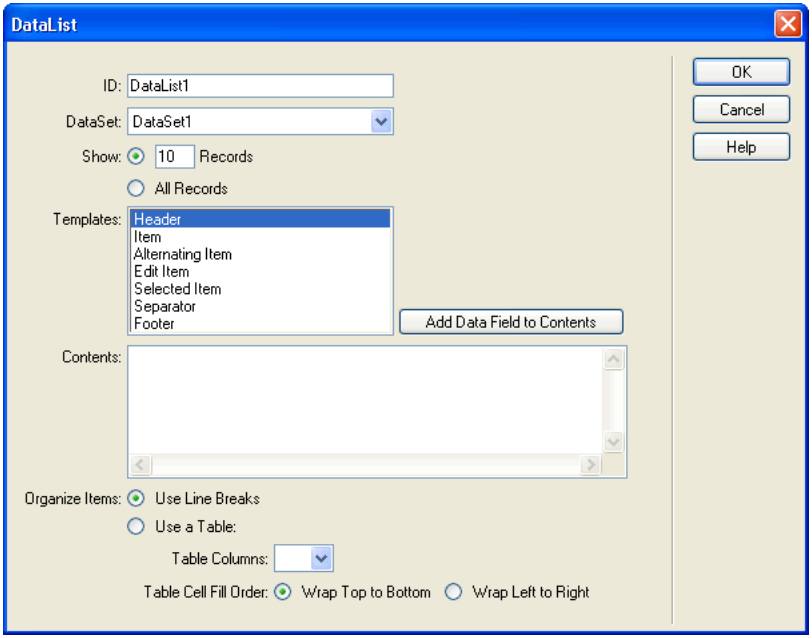
The Dreamweaver DataList server behavior allows you to insert an ASP.NET DataList control into a page.

Note: 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 "Understanding recordsets" on page 526.

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 is displayed 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

You can always modify the DataGrid and DataList objects to suit your page design needs.

To modify a DataGrid and DataList object:

- 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 box to change its attributes.

CHAPTER 36
Using ColdFusion Components

Macromedia ColdFusion MX 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 2004 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:

- "About ColdFusion components" on page 573
- "Visually creating a component in Dreamweaver" on page 574
- "Viewing ColdFusion components in Dreamweaver" on page 575
- "Editing ColdFusion components in Dreamweaver" on page 576
- "Building web pages that use ColdFusion components" on page 577

About 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 the ColdFusion documentation from within 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 customs 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 manually from the server.

Related topics

- "About ColdFusion components" on page 573
- "Editing ColdFusion components in Dreamweaver" on page 576
- "Building web pages that use ColdFusion components" on page 577

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 461). If you want to view components located on another server, change the testing server settings.

You can view any of the following information about your CF components:

- 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 (+) button beside the package name to view the components stored in the package.
   - To list the functions of a component, click the Plus (+) button beside the component name.
   - 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 (+) button beside them.
   - 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 button 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.

Related topics

- "About ColdFusion components" on page 573
- "Visually creating a component in Dreamweaver" on page 574
- "Building web pages that use ColdFusion components" on page 577

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 selecting Site > Edit Sites.
- The component must be stored in the local site folder or any of its subfolders on your hard disk.

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 575.

To edit a component:

1. Open any ColdFusion page in Dreamweaver and display the components in the Components panel (Window > Components).
2. 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 575.
3. 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.
4. To edit a specific function, argument, or property, double-click the item in the tree view.
5. Make your changes by hand in Code view.
6. 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 the ColdFusion documentation from within 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 button on the panel toolbar (the second button 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 the ColdFusion documentation from within Dreamweaver (Help > Using ColdFusion).
6. Save the page (File > Save).

CHAPTER 37
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 2004 lets you create pages and sites that are consumers of web services. Dreamweaver currently supports the creation of web service consumers using Macromedia ColdFusion MX, 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 contains the following sections:

- "About web services" on page 580
- "About proxy generators" on page 582
- "Adding a web service proxy using the WSDL description" on page 585
- "Adding a web service to a page" on page 586
- "Editing the UDDI web service site list" on page 588

About 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 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.

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.

The Dreamweaver web service workflow

To create a page or site that is a consumer of 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 Web Service Description Language (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 "About proxy generators" on page 582.

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 on either of the following:

- The local computer where you are developing the web service consumer.
- 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.

For more information, see "About proxy generators" on page 582.

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 586.

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 https://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.

Related topics

- "Editing the UDDI web service site list" on page 588

Web service software components

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 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.

Related topics

- "Adding a web service proxy using the WSDL description" on page 585

Web service references

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/

About 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.

Related topics

- "Configuring proxy generators for use with Dreamweaver" on page 583

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.

Configuring proxy generators for use with Dreamweaver

When you install a web service proxy generator, you must configure it to work with Dreamweaver.

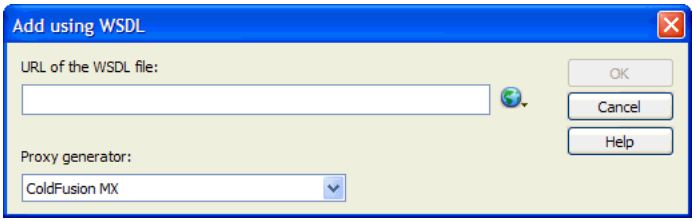
To learn more about proxy generators, see "About proxy generators" on page 582.

To configure a proxy generator to work with Dreamweaver:

1 Select Window > Components to open the Components panel.

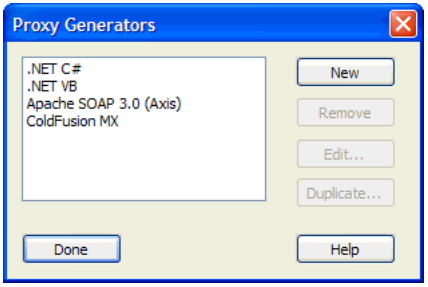
2 In the Components panel, select Web Services from the pop-up menu in the upper left of the panel, then click the Plus (+) button and select Add Using WSDL.

The Add Using WSDL dialog box appears.



3 In the Add Using WSDL dialog box, select Edit Proxy Generator List from the Proxy Generator pop-up menu.

The Proxy Generator dialog box appears.

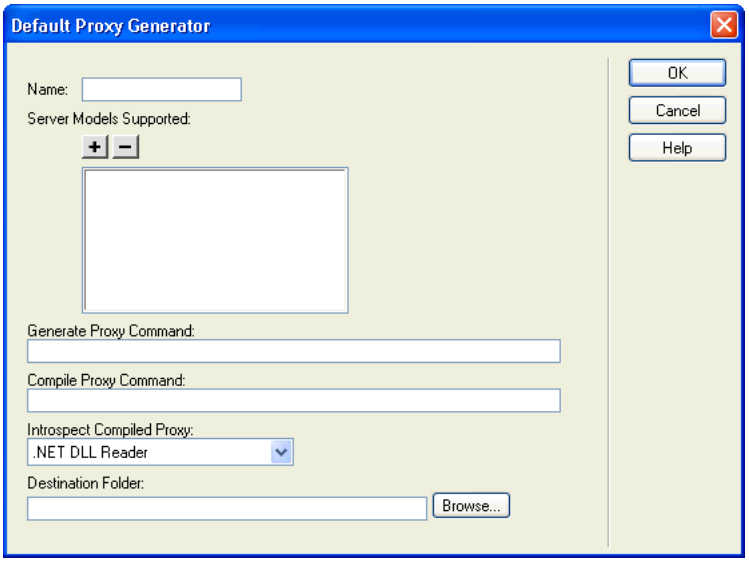


4 Click New, select the proxy generator from the pop-up menu, and click Done.

If the proxy generator you want to use does not appear in the list, select Default Proxy Generator to display the Default Proxy Generator dialog box.

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.

Note: The ColdFusion MX proxy generator is not editable.



5 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.
- Returns the proxy source code and the compiled proxy in the specified destination folder.

Related topics

- "About proxy generators" on page 582
- "Obtaining additional proxy generators" on page 582

Adding a web service proxy using the WSDL description

After you have specified a proxy generator (see "About proxy generators" on page 582) 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.

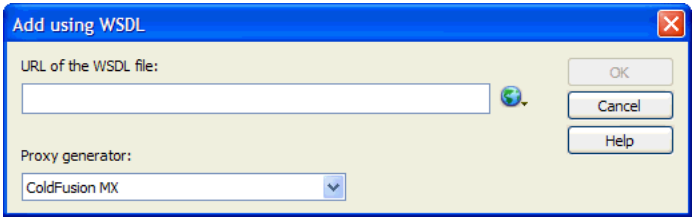
To select a web service and generate a proxy from its WSDL file:

1 Open the page that you want to add the web service to.

2 Select Window > Components to open the Components panel.

3 In the Components panel, select 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 start a web browser, click the UDDI browse button and select one of the listed web service registries. Dreamweaver will start 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 588.

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 "About proxy generators" on page 582.

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 Components panel. You can now add the web service to a page.

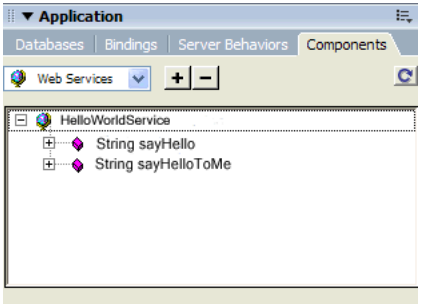
Related topics

- "Web service software components" on page 581

Adding a web service to a page

After selecting a web service, generating its proxy, and adding it to the Components panel, you can insert it into a page.

The illustration below shows the Components panel with the web service proxy `Helloworld` added. The `Helloworld` proxy provides one method, `sayHello`, which prints "Hello World."



The following example instantiates 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: 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://www.mysite.com:8500:8500/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: Hello world!`" 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://www.mysite.com:8500/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 be sure to deploy both the proxy and any dependent 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. For more information, see "Finding web service publishers" on page 581.

To edit the web service site list:

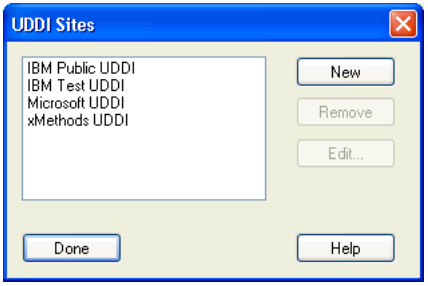
1 In Dreamweaver, select Window > Components to open the Components panel.

2 In the Components panel, select Web Services from the pop-up menu in the upper left of the panel, and 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 select Edit UDDI Site List from the pop-up menu.

The UDDI Sites dialog box appears.



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 add a new site, or modify and existing one, click the New or Edit button and complete the dialog box that displays.
- To remove an existing site, select it in the list and click the Remove button.

CHAPTER 38
Adding Custom Server Behaviors

Macromedia Dreamweaver MX 2004 comes with a set of built-in server behaviors that lets 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:

- "About custom server behaviors" on page 589
- "Installing third-party server behaviors" on page 598
- "Using the Server Behavior Builder" on page 598
- "Using parameters in server behaviors" on page 601
- "Positioning code blocks" on page 602
- "Server behaviors" on page 590
- "Editing and modifying server behaviors" on page 605

About custom server behaviors

Before creating your own server behaviors, you should check the Macromedia Exchange website to see if another party has already created a server behavior that supplies the functionality you'd like to add to your website. Often, a third-party developer has created and tested a server behavior that will address your needs.

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.

If you are going to create your own server behavior, you should be familiar with the web programming language used by your website. This chapter provides guidelines specific to creating server behaviors using Dreamweaver. It does not instruct you in programming languages or server behavior testing.

Server behaviors

If you are a developer proficient in Macromedia 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 598. For information on the syntax supported by Dreamweaver server behaviors, see "Using parameters in server behaviors" on page 601.
- Specifying where the code block should be inserted within the page's HTML code.

  For information on positioning code blocks within a page see "Positioning code blocks" on page 602.
- 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 603.
- Testing the server behavior before making it available to others.

  For guidelines on testing server behaviors see "Testing server behaviors" on page 597.

About 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.

Related topics

- "Coding guidelines" on page 596

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.

For more information, see "Using the Server Behavior Builder" on page 598

Runtime 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 601.

Conditions and repeating elements in 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.

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 general information on coding, see "Coding guidelines" on page 596.

Related topics

- "Making code blocks conditional" on page 593
- "Repeating code blocks" on page 594

Code block positioning within web pages

When you create code blocks using the Server Behavior Builder (see "Using the Server Behavior Builder" for more information), you must specify where to insert them in the page's HTML code. You do this using the Server Behavior Builder dialog box's Insert Code and Relative Position pop-up menus, which let you select where to insert the code block within the document, and then specify a position relative to another tag in the page.

For example, if you insert a code block above the opening `<html>` tag, you must then specify the code blocks position relative to other tags, scripts, and server behaviors in that section of the page's HTML code. Typical examples include positioning a behavior either before or after any recordset queries that might also exist in the page code above the opening `<html>` tag.

When you select a positioning option from the Insert Code pop-up menu, the options available in the Relative Position pop-up menu change to provide relevant options for that part of the page. For example, if you select Above the `<html>` Tag in the Insert Code pop-up menu, then the positioning options available in the Relative Position pop-up menu reflect choices relevant for that part of the page.

The code block insert options, and the relative positioning options available for each, are shown in the table below.

Insert Code options	Relative position options
Above the <html> Tag	• At the beginning of the file • Just before the recordsets • Just after the recordsets • Just above the `<html>` tag • Custom position
Below the </html> Tag	• Before the end of the file • Before the recordset close • After the recordset close • After the `</html>` tag • Custom position
Relative to a Specific Tag	Select a tag from the Tag pop-up menu, and then choose from the tag positioning options.
Relative to the Selection	Before the selection After the selection Replace the selection Wrap the selection

If you want to specify a custom position, you must assign a *weight* to the code block. 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 would enter a weight of 60 for the first block, 65 for the second, and 70 for the third.

By default, 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.

Related topics

- "About code blocks" on page 590
- "Positioning code blocks" on page 602

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 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 603.

Making code blocks conditional

Dreamweaver lets you develop code blocks that incorporate control statements that execute conditionally. The Server Behavior Builder uses `if`, `elseif`, and `else` statements, and may also contain server behavior parameters. This enables 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 594) 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.

Related topics

- "About code blocks" on page 590

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 allow a user 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. For more information, see "Using the loop directive's _length and _index variables" on page 596.

When you later create a dialog box for the server behavior (see "Creating a dialog box for a custom server behavior" on page 603), 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 "Positioning code blocks" on page 602) 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 Builder 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 would only output the attribute name `import=` on the first iteration of the loop—this includes 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 follows:

```
<@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.

The following is a list of coding guidelines to keep in mind when creating server behaviors:

Error checking is an important requirement. 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?

Unique names help to ensure that your code is clearly identifiable and avoids 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.

Code prefixes allow you to identify your own runtime functions and global variables in a page. One convention is to use your initials. Never use the `MM_` prefix, as it is reserved for Macromedia use only. Macromedia precedes all functions and global variables with the prefix `MM_` to prevent them from conflicting with any code that you write.

```
var MM_ERROR_STRING = "...";
function MM_hideLayer() {
```

Avoid similar code blocks so that the code you write doesn'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.

Related topics

- "About custom server behaviors" on page 589
- "Server behaviors" on page 590

Testing server behaviors

The Macromedia Exchange recommends performing 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.

Installing third-party server behaviors

You can download and install server behaviors created by independent developers from the Macromedia Exchange website.

To access Macromedia Exchange:

1 In Dreamweaver select Help > Dreamweaver Exchange.

Your browser opens 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 selecting Get More Server Behaviors.

To install a server behavior or other extension in Dreamweaver:

1 Start the Extension Manager by selecting Commands > Manage Extensions.

2 Select File > Install Package in the Extension Manager.

For more information, see the Extension Manager online help.

Related topics

- "About custom server behaviors" on page 589
- "Server behaviors" on page 590

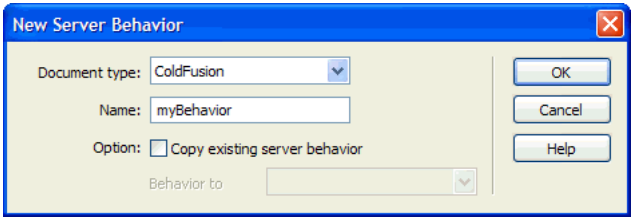
Using the Server Behavior Builder

The Server Behavior Builder lets you add the code block or blocks that the behavior inserts into a page.

To write server behavior code blocks:

1 In the Server Behaviors panel (Window > Server Behaviors), click the Plus (+) button and select New Server Behavior from the pop-up menu.

The New Server Behavior dialog box appears.



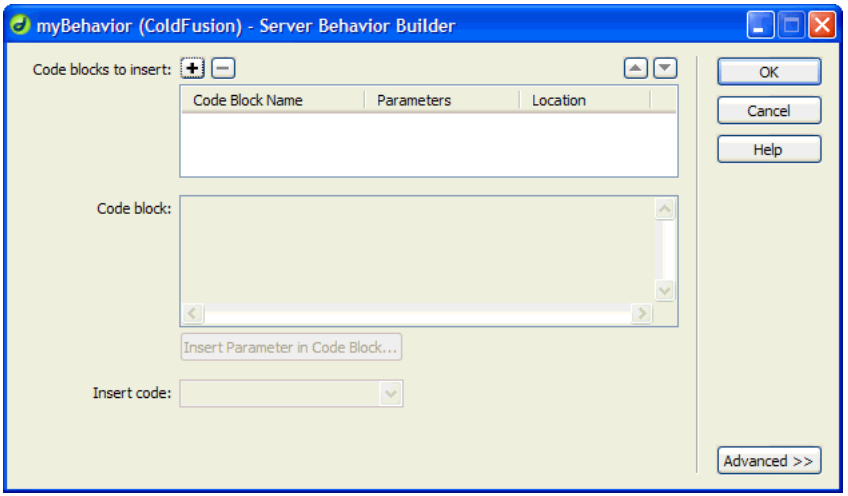
2 From the Document Type pop-up menu, select the document type that you are developing the server behavior for.

3 In the Name text box, enter a name for the server behavior.

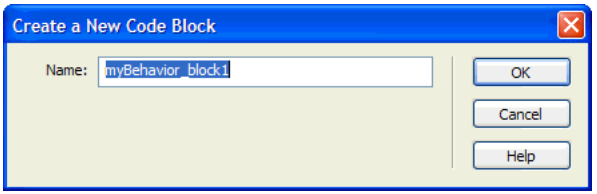
4 If you want to copy an existing server behavior to add to the behavior you are creating, select the Copy Existing Server Behavior checkbox.

When this checkbox is selected, a list of available server behaviors is displayed in the Behavior to Copy pop-up menu.

5 Click OK.
The Server Behavior Builder dialog box is displayed.

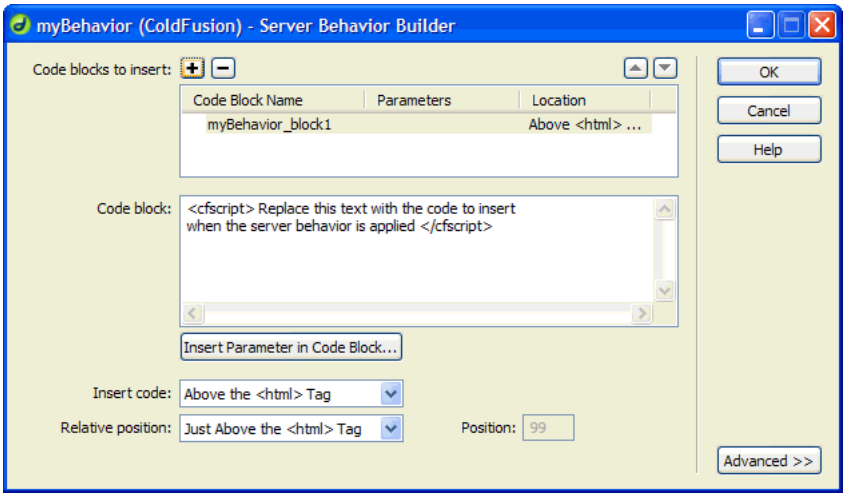


6 To add a new code block, click the Plus (+) button.
The Create a New Code Block dialog box is displayed.



7 Enter a name for the code block you want to create and click OK.

The name you entered in the dialog box appears in the Server Behavior Builder, with the appropriate scripting tags visible in the Code block text box.



8 In the Code Block text box, enter the code necessary to implement the server behavior.

When entering code in the Code Block text box:

- You can insert only a single tag or code block for each named code block (for example, `myBehavior_block1`, `myBehavior_block2`, `myBehavior_blockn`, etc.). If you need to enter multiple tags or code blocks, you must create an individual code block for each using the Server Behavior Builder.
- To include runtime parameters in a code block:

a Place the insertion point in the code block where you'd like to insert the parameter.

b Click the Insert Parameters in Code Block button.

The Insert Parameters in Code Block dialog box appears.

c Enter a name for the parameter in the Parameter Name text box.

d Click OK.

The parameter name is inserted in the code block.

- Repeat steps 6 through 8 for each new code block you want to create.

9 In the Parameter Name pop-up menu, enter a name for the parameters, and click OK.

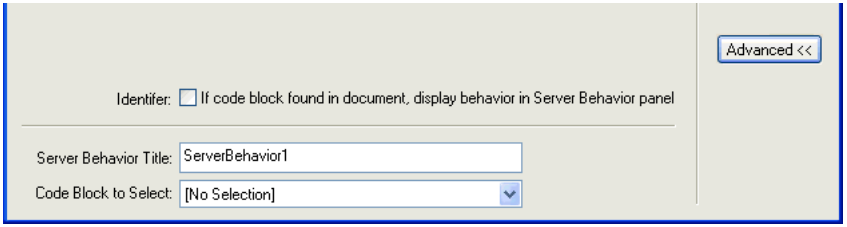
The parameter is inserted into the code block at the location where you placed the insertion point prior to defining the parameter.

10 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 602.

11 You can specify additional information about the server you are creating by using the Advanced options panel.

12 Click the Advanced button to display more options.



13 If you need to create more code blocks, repeat steps 7 through 13 as needed.

14 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 603.

15 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.

Related topics

- "Positioning code blocks" on page 602
- "Repeating code blocks" on page 594
- "Coding guidelines" on page 596
- "Positioning code blocks" on page 602

Using parameters in server behaviors

You can include parameters in a server behavior's code (see "Parameters in server behaviors" on page 592 for more information), 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 ASP server behavior example below contains the parameter `formParam`, which requires the person inserting the behavior to supply 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 603.

Positioning code blocks

When you create code blocks using the Server Behavior Builder (see "Using the Server Behavior Builder" for more information), you must specify where to insert them in the page's HTML code. The Insert Code and Relative Position pop-up menus let you choose where to insert the code block in the document, and then specify a position relative to another tag in the page.

To learn more about the code block positioning options and how they affect your custom server behavior, see "Code block positioning within web pages" on page 591.

To position a code block (general instructions):

1. Using the Server Behavior Builder, write a code block according to the section "Using the Server Behavior Builder" on page 598.
2. In the Server Behavior Builder dialog box, select a position in which to insert the code block from the Insert Code pop-up menu.
3. In the Server Behavior Builder dialog box, select a position relative to that which you selected in the Insert Code pop-up menu.
4. If you have completed the authoring of the code block, click OK.

   The server behavior is listed in the Server Behaviors panel (Window > Server Behavior), and can be viewed by clicking the click the Plus (+) button.
5. Test the server behavior and ensure that it functions properly.

   See "Testing server behaviors" on page 597 for more information.

To position a code block relative to another tag on the page:

1. In the Insert Code pop-up menu, select 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 select 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, select 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 follows:

   ```
   <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.

Related topics

- “About custom server behaviors” on page 589
- “Code block positioning within web pages” on page 591
- “Using the Server Behavior Builder” on page 598

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 select 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 as follows:

  `@@parameterName@@`

For example, if the server behavior contains the following code block:

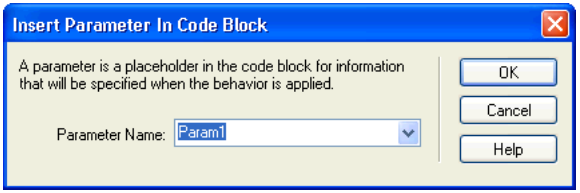
```
<% Session("lang_pref") = Request.Form("Form_Object_Name"); %>
```

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 and 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:

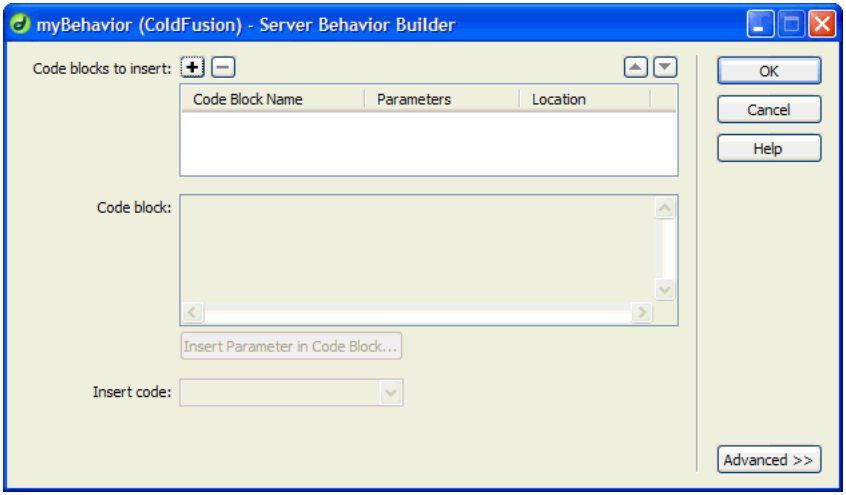


Note: Parameter names in the server behavior code cannot have any spaces. Therefore, the dialog box labels cannot have any spaces. If you want to include spaces in the label, you can edit the generated HTML file.

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 (Optional) You can change the display order of the dialog box controls by selecting a parameter and clicking the up and down arrows.

3 (Optional) 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 labeled 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 select 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 (Optional) You can change the display order of the dialog box controls by selecting a parameter and clicking the up and down arrows.

5 (Optional) If you want, change a parameter's control by selecting the parameter and choosing another control in the Display As column.

6 Click OK.

Editing and modifying server behaviors

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.

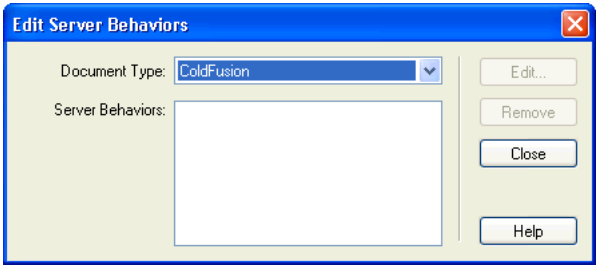
If you want both the old and new versions of the behavior to appear in the panel:

- Click the Plus (+) button on the Server Behaviors panel (Window > Server Behaviors), select New Server Behavior, and create a copy of the old server behavior.

To edit the code of a server behavior created with the Server Behavior Builder:

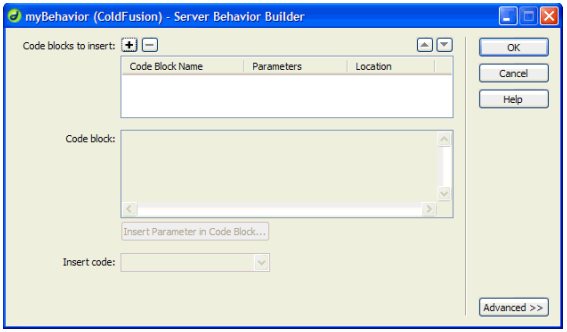
1 In the Server Behaviors panel (Window > Server Behaviors), click the Plus (+) button and select Edit Server Behaviors from the pop-up menu.

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 (Optional) You can change or add parameter markers to the code.

For instructions, see "Creating a dialog box for a custom server behavior" on page 603.

5 (Optional) If you want, change where the code block is inserted in the page's HTML code by choosing another option in the Insert Code pop-up menu.

For instructions, see "Positioning code blocks" on page 602.

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 39
Creating Forms

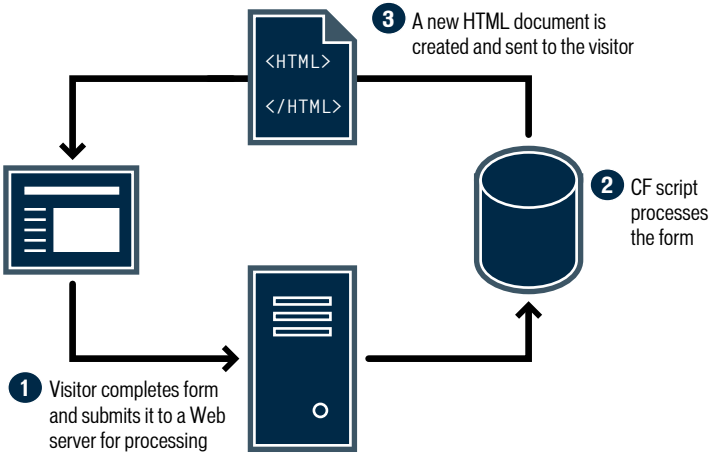
You can use Macromedia Dreamweaver MX 2004 to create forms with text fields, password fields, radio buttons, checkboxes, pop-up menus, clickable buttons, and other form objects. Dreamweaver can also write code that validates the information a visitor provides. For example, you can check that an e-mail address a user enters contains an "@" symbol, or that a required text field contains a value.

About forms

Forms let you obtain information from visitors to your website. Visitors enter information using form objects such as text fields, list boxes, checkboxes, and radio buttons, and then click a button to submit the information.

Client-side role of forms

Forms support the client side of the client-server relationship. When a visitor enters information into a form displayed in a web browser (the client) and clicks the submit button, the information is sent to the server where a server-side script or application processes it. Common server-side technologies used for processing form data include Macromedia ColdFusion, Microsoft Active Server Pages (ASP), and PHP. The server responds by sending requested information back to the user (or client), or performing some action based on the form's contents.

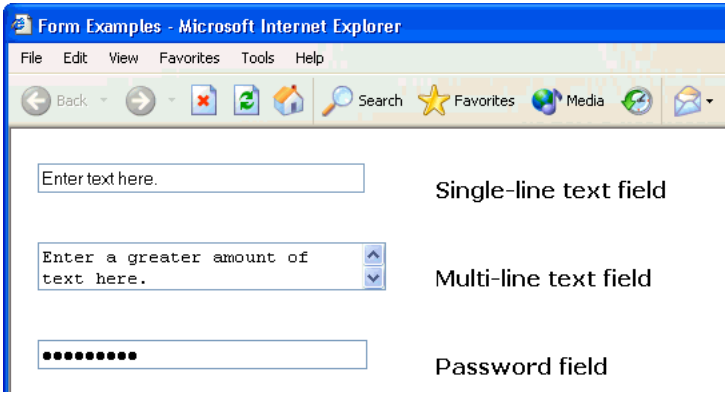


Note: You can also send form data directly to an e-mail recipient.

Form objects

In Dreamweaver, form input types are called *form objects*. Form objects are the mechanisms that allow users to input data. You can add the following form objects to a form:

Text fields accept any type of alphanumeric text entry. The text can be displayed as a single line, multiple lines, and as a password field where entered text is replaced by asterisks or bullets to hide the text from onlookers.

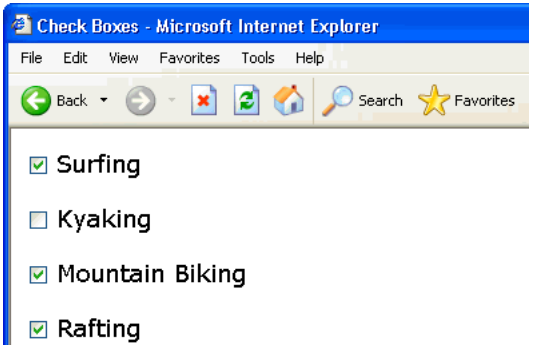


Note: Passwords and other information sent to a server using a password field are not encrypted. The transferred data can be intercepted and read as alphanumeric text. For this reason, you should always provide encryption for data you want to keep secure.

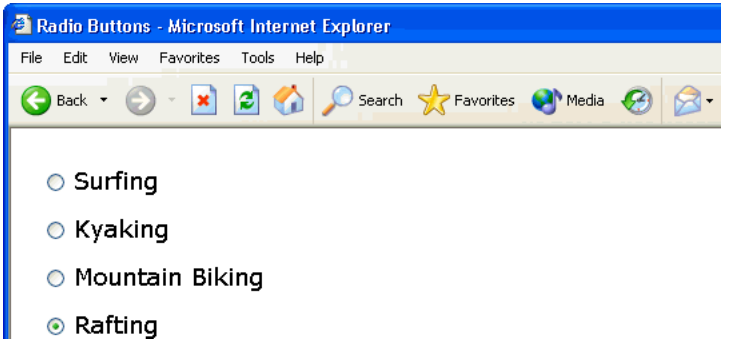
Hidden fields store information entered by a user, such as a name, e-mail address, or viewing preference, and then use that data when the user next visits the site.

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.

Checkboxes allow multiple responses within a single group of options. A user can select as many options as apply. The example below illustrates this by showing three checkbox items selected: Surfing, Mountain Biking, and Rafting.



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). In the example below, *Rafting* is the currently selected option. If the user clicks *Surfing*, the *Rafting* button is automatically cleared.



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 menu that allows users to select only a single item.

Note: A pop-up menu on an HTML form 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 393.

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.

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 such as Submit or Reset buttons.

Related topics

- "Creating HTML forms" on page 611
- "Inserting HTML form objects" on page 613

Dynamic form objects

A dynamic form object is a form object whose initial state is determined by the server when the page is requested from the server, not by the form designer at design time. For example, when a user requests a PHP page containing a form with a menu, a PHP script in the page automatically populates the menu with values stored in a database. The server then sends the completed page to the user's browser.

Making form objects dynamic can simplify site maintenance. For example, many sites use menus to present users with a set of options. If the menu is dynamic, you can add, remove, or change menu items in a single place—the database table in which the items are stored—to update all instances of the same menu on the site.

A menu is not the only type of dynamic form object. You can also create and use dynamic radio buttons, checkboxes, text fields, and image fields.

Related topics

- "Inserting HTML form objects" on page 613
- "Inserting dynamic HTML form objects" on page 617

Creating HTML forms

This section describes how to create HTML forms in Dreamweaver.

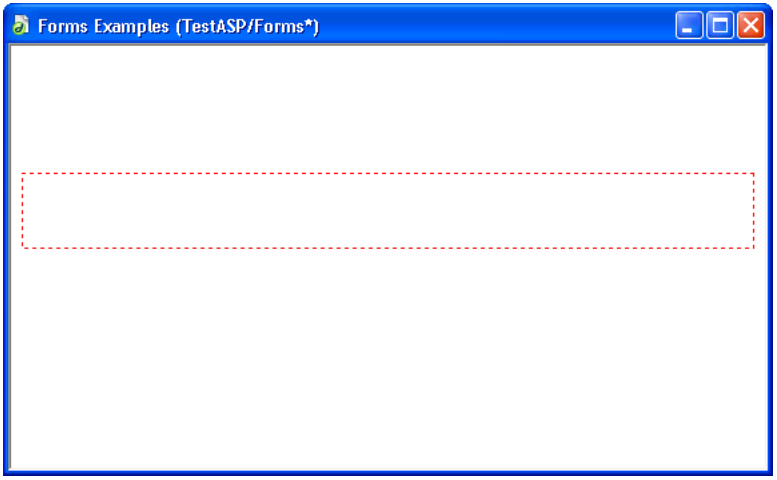
You can also use Dreamweaver to create ASP.NET web forms. For more information, see "Creating ASP.NET forms" on page 623.

To create an HTML form:

1 Open a page and place the insertion point where you want the form to appear.

2 Select Insert > Form, or select the Forms category in the Insert bar and click the Form icon.

Dreamweaver inserts an empty 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 Specify the page or script that will process the form data.

In the Document window, click the form outline to select the form. In the Property inspector (Window > Properties), type the path in the Property inspector's Action text box, or click the folder icon to navigate to the appropriate page or script.

4 Specify the method to use to transmit the form data to the server.

In the Property inspector, select one of the following options in the Method pop-up menu:

Default uses the browser's default setting to send the form data to the server. Typically the default is the `GET` method.

GET appends the value to the URL requesting the page.

POST embeds the form data in the HTTP request.

For more information, click the Help icon in the Property inspector.

5 Insert form objects.

Place the insertion point where you want the form object to appear in the form, and then select the object in the Insert > Form menu, or in the Forms category of the Insert bar.

For more information, see "Inserting HTML form objects" on page 613 or "Inserting dynamic HTML form objects" on page 617.

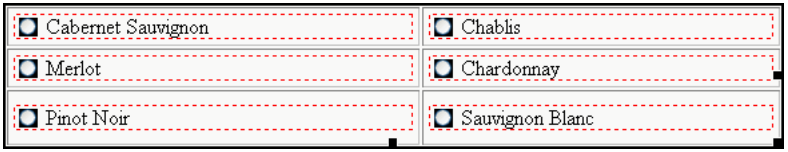
If desired, set the properties of the form object after inserting it in the page. For more information, select the form object and click the Help icon in the Property inspector.

6 Adjust the layout of the form as desired.

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.

Use tables to provide structure for form objects and field labels. When using tables in forms make sure all the `table` tags are included between the `form` tags.



Related topics

- "Validating HTML form data" on page 620
- "Attaching JavaScript behaviors to HTML form objects" on page 620
- "Attaching custom scripts to HTML form buttons" on page 621
- "Creating accessible HTML forms" on page 622

Inserting HTML form objects

You can use Dreamweaver to quickly insert HTML form objects into your forms.

If you're working on an ASP.NET web application, you can use Dreamweaver to quickly insert ASP.NET form controls. For more information, see "Adding ASP.NET form controls to a page" on page 623.

You can create a blank HTML form (Insert > Form > Form) before inserting form objects in it. For more information, see "Creating HTML forms" on page 611. If you don't create a blank form and attempt to insert a form object, Dreamweaver will ask you if you want to create one.

Inserting HTML text fields

You can create a text field that consists of one line or multiple lines. You can also create a password text field that hides the text the user enters.

To insert a text field:

1. Place the insertion point inside the form outline.
2. Select Insert > Form > Text Field.

   A text field appears in the document.
3. In the Property inspector, set the text field properties as desired.

   For more information, click the Help icon in the Property inspector.
4. To label the text field on the page, click beside the text field and type the label.

Related topics

- "Validating HTML form data" on page 620
- "Displaying dynamic content in HTML text fields" on page 618
- "Form objects" on page 608

Inserting HTML checkboxes

Use HTML checkboxes when you want to let users select more than one option from a set of options.

Note: Use radio buttons when you want users to select only one choice from a set of options. For more information, see "Inserting HTML radio buttons" on page 614.

To insert a checkbox:

1. Place the insertion point inside the form outline.
2. Select Insert > Form > Check Box.

   A checkbox appears in the document.
3. In the Property inspector, set the checkbox properties as desired.

   For more information, click the Help icon in the Property inspector.
4. To label the checkbox, click beside the checkbox on the page and type the label text.

Related topics

- "Dynamically preselecting HTML checkboxes" on page 618

Inserting HTML radio buttons

Use HTML radio buttons when you want users to 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.

Note: Use checkboxes when you want let users select more than one option. For more information, see "Inserting HTML checkboxes" on page 613.

To insert a group of radio buttons:

1. Place the insertion point inside the form outline.
2. Select Insert > Form > 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 by selecting Insert > Form > Radio Button.

   A radio button appears in the document.
3. In the Property inspector, set the radio button properties as desired.

   For more information, click the Help icon in the Property inspector.
4. To label the radio button, click beside the radio button on the page and type the label text.

Related topics

- "Dynamically preselecting HTML radio buttons" on page 619

Inserting HTML form menus

An HTML form menu lets a visitor select one or more items from a list. Menus are useful when you have a limited amount of space, but need to display many items. They're also useful when you want to control the values returned to the server. Unlike text fields where users can type anything they want, including invalid data, you set the exact values returned by a menu.

You can insert two types of menus in a form: a menu that drops down when the user clicks it or a menu that displays a scrollable list of items that can be selected. This type is called a list menu.

To insert a menu:

1. Place the insertion point inside the form outline.
2. Select Insert > Form > List/Menu.

   A menu appears in the document.
3. In the Property inspector, set the menu properties as desired.

   For more information, click the Help icon in the Property inspector.

Related topics

- “Creating a dynamic HTML form menu” on page 617
- “Making existing HTML form menus dynamic” on page 618

Inserting standard buttons

Buttons control form operations. Use a button to submit form data to the server or to reset the form. Standard form buttons are typically labeled Submit, Reset, or Send. You can also assign other processing tasks that you've defined in a script. For example, the button might calculate the total cost of items selected based on assigned values.

To create a button:

1. Place the insertion point inside the form outline.
2. Select Insert > Form > Button.

   A button appears in the form.
3. In the Property inspector, set the button properties as desired.

   For more information, click the Help icon in the Property inspector.

Inserting image buttons

You can use images as button icons. Using an image to perform tasks other than submitting data requires attaching a behavior to the form object.

To create an image button:

1. In the document, place the insertion point inside the form outline.
2. Select Insert > Form > Image Field.

   The Image Source dialog box appears.
3. Select the image for the button in the Select Image Source dialog box and click OK.

   A image field appears in the form.
4. In the Property inspector, set the image field properties as desired.

   To create a submit button, enter Submit in the Image Field text box in the Property inspector.

   For more information, click the Help icon in the Property inspector.
5. To attach a JavaScript behavior to the button, select the image and then select the behavior in the Behaviors panel (Window > Behaviors).

   For more information, “Attaching JavaScript behaviors to HTML form objects” on page 620 and “Attaching custom scripts to HTML form buttons” on page 621.

Related topics

- “Inserting standard buttons” on page 615

Inserting hidden fields

You can use hidden fields to store and submit information not entered by the user. The information is hidden from the user.

To create a hidden field:

1. In Design view, place the insertion point inside the form outline.
2. Select Insert > Form > Hidden Field.

   A marker appears in the document. If you don't see a marker, select View > Visual Aids > Invisible Elements to see the marker.
3. In the Property inspector's HiddenField text box, type a unique name for the field.
4. In the Value text box, type the value you want to assign to the field.

Inserting file-upload fields

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 text box.

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 to display its Property inspector.
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 and select Insert > Form > File Field.

   A file field is inserted within the form.
6. In the Property inspector, set the file field properties as desired.

   For more information, click the Help icon in the Property inspector.

Inserting dynamic HTML form objects

You can insert HTML form objects whose initial state is determined by the server when the page is requested from the server, not by the form designer at design time.

Related topics

- "Dynamic form objects" on page 610
- "Creating ASP.NET forms" on page 623

Creating a dynamic HTML form menu

You can dynamically populate an HTML form menu or list menu with entries from a database.

For most pages, you can use an HTML menu object. For ASP.NET pages, you must use a DropDownList or ListBox form control. For more information, see "Creating a dynamic ASP.NET menu" on page 624.

Before you begin, you must insert an HTML form in a ColdFusion, PHP, ASP, or JSP page, and you must define a recordset or other source of dynamic content for the menu. For more information, see "Defining a recordset" on page 530.

To insert a dynamic HTML form menu:

1 Click inside the HTML form on your page.

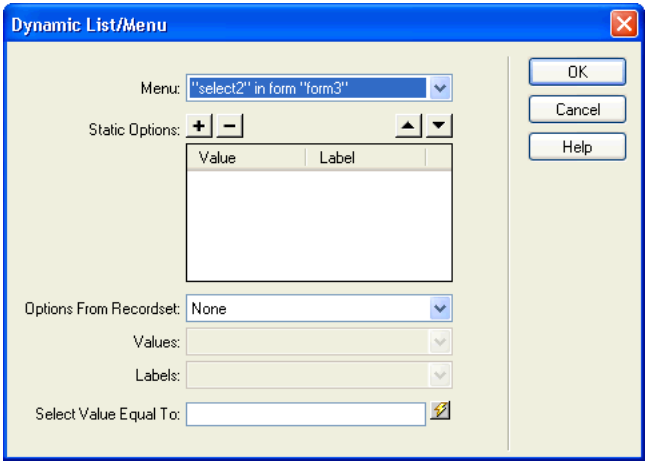
2 Select Insert > Form > List/Menu.

Dreamweaver inserts a List/Menu form object into the page.

3 Select the List/Menu form object.

The Property inspector displays the List/Menu properties.

4 In the Property inspector, click the Dynamic button to display the Dynamic List/Menu dialog box.



5 Complete the dialog box and click OK.

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

Making existing HTML form menus dynamic

You can make an existing HTML form menu or list menu dynamic.

This section deals with HTML form objects. For ASP.NET menu objects such as DropDownList or ListBox controls, see "Making an existing ASP.NET menu dynamic" on page 624.

Before you begin, you must create the form in a ColdFusion, PHP, ASP, or JSP page, and you must define a recordset or other source of dynamic content for the menu. For more information, see "Defining a recordset" on page 530.

To make an existing HTML form menu dynamic:

1. In Design view, 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, click the Help button in the dialog box.

Related topics

- "Creating a dynamic HTML form menu" on page 617

Displaying dynamic content in HTML text fields

You can display dynamic content in HTML text fields.

For ASP.NET pages, you must use the ASP.NET TextBox control. For more information, see "Displaying dynamic content in an ASP.NET TextBox control" on page 625.

Before you begin, you must create the form in a ColdFusion, PHP, ASP, or JSP page, and you must define a recordset or other source of dynamic content for the text field. For more information, see "Defining a recordset" on page 530.

To make HTML text fields dynamic:

1. Select the text field in the HTML form on your page.
2. In the Property inspector, click the lightning bolt icon beside the Init Val text box.

   The Dynamic Data dialog box appears.
3. Select the recordset column that will supply a value to the text field, then click OK.

The text field will display the dynamic content when the form is viewed in a browser.

Dynamically preselecting HTML checkboxes

You can let the server decide whether to select a checkbox when the form is displayed in a browser.

For ASP.NET pages, you must use the ASP.NET CheckBox control. For more information, see "Dynamically preselecting ASP.NET CheckBox controls" on page 625.

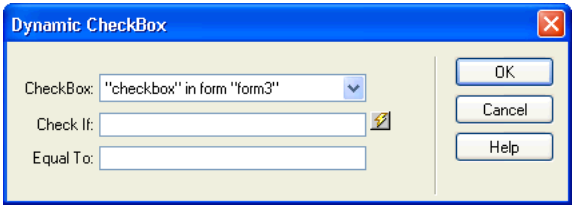
Before you begin, you must create the form in a ColdFusion, PHP, ASP, or JSP page, and you must define a recordset or other source of dynamic content for the checkboxes. For more information, see "Defining a recordset" on page 530. Ideally, the source of content should contain Boolean data, such as Yes/No or true/false.

To dynamically preselect an HTML checkbox:

1 Select a checkbox form object on your page.

2 In the Property inspector, click the Dynamic button.

The Dynamic CheckBox dialog box appears.



3 Complete the dialog box and click OK.

For instructions, click the Help button in the dialog box.

The checkbox will appear selected or not selected, depending on the data when the form is viewed in a browser.

Dynamically preselecting HTML radio buttons

You can let the server decide whether to select an HTML radio button when the form is displayed in a browser.

If you want to dynamic preselect ASP.NET radio button objects such as RadioButton or RadioButtonList controls, see "Dynamically preselecting an item in an ASP.NET RadioButtonList" on page 626.

Before you begin, you must create the form in a ColdFusion, PHP, ASP, or JSP page, and insert at least one group of HTML radio buttons (see "Inserting HTML radio buttons" on page 614). You must also define a recordset or other source of dynamic content for the radio buttons. For more information, see "Defining a recordset" on page 530. Ideally, the source of content should contain Boolean data, such as Yes/No or true/false.

To dynamically preselect an HTML radio button:

1 In Design view, select a radio button in the group.

2 In the Property inspector, click the Dynamic button.

The Dynamic Radio Group dialog box appears.

3 Complete the dialog box and click OK.

For instructions, click the Help button in the dialog box.

Validating HTML form data

Dreamweaver can add JavaScript code that checks the contents of specified text fields to ensure that the user has entered the correct type of data.

Note: This feature only works with HTML forms. For ASP.NET forms, you can insert ASP.NET validation controls in Code view. For more information, consult the ASP.NET documentation.

To validate form data:

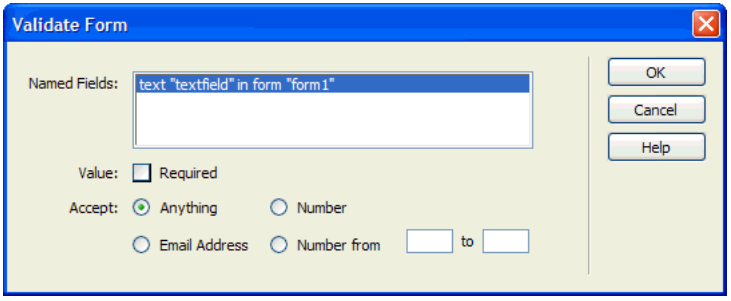
1 Create a HTML form that includes at least one text field and one Submit button.

Make sure every text field you want to validate has a unique name.

2 Select the Submit button.

3 In the Behaviors panel (Window > Behaviors), click the Plus (+) button and select the Validate Form behavior from the list.

The Validate Form dialog box appears.



Note: The Validate Form behavior is available only if a text field has been inserted into the document.

4 Set the validation rules for each text field and click OK.

For example, you might specify that a text field for a person's age accepts only numbers. For more information, see "Validate Form" on page 397.

Related topics

- Chapter 18, "Using JavaScript Behaviors," on page 371

Attaching JavaScript behaviors to HTML form objects

You can attach JavaScript behaviors stored in Dreamweaver to HTML form objects such as buttons.

Note: This feature does not work with ASP.NET form controls because these controls are processed on the server.

To attach a JavaScript behavior to an HTML form object:

1 Select the HTML form object.

2 In the Behaviors panel (Window > Behaviors), click the Plus (+) button and select a behavior from the list.

For more information, see Chapter 18, "Using JavaScript Behaviors," on page 371.

Attaching custom scripts to HTML form buttons

Some forms use JavaScript or VBScript to perform form processing or some other action on the client as opposed to sending the form data to the server for processing. You can use Dreamweaver to configure a form button to run a specific client-side script when the user clicks the button.

Note: This feature does not work with ASP.NET forms.

To run a script on the client:

1 Select a Submit button in a form.

2 In the Behaviors panel (Window > Behaviors), click the Plus (+) button and select Call JavaScript from the list.

For more information on this behavior, see "Call JavaScript" on page 377.

3 In the Call JavaScript text box that appears, enter the name of the JavaScript function you want to run when the user clicks the button, and click OK.

For example, you can enter the name of a function that doesn't exist yet, such as `processMyForm()`.

4 If your JavaScript function doesn't exist in the `head` section of the document yet, add it now.

For example, you could define the following JavaScript function in the `head` section of the document to display a message when the user clicks the Submit button.

```
function processMyForm(){
  alert('Thanks for your order!');
  }
```

Creating accessible HTML forms

When you insert an HTML 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. You can also change accessibility attributes after inserting the object.

To add an accessible form object:

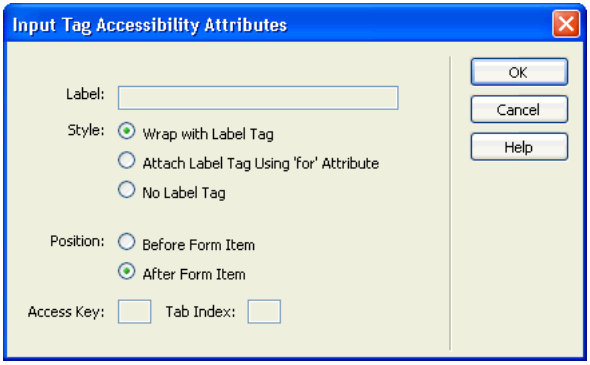
1 The first time you add accessible form objects, activate the Accessibility dialog box for form objects (see "Optimizing the workspace for accessible page design" on page 53).

This is a one-time-only step.

2 In the document, place the insertion point where you want the form object to appear.

3 Select Insert > Form and select a form object to insert.

The Input 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.

Note: If you press Cancel, the form object appears in the document, but Dreamweaver does not associate accessibility tags or attributes with it.

6 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), then select Edit Tag.

Creating ASP.NET forms

You can create ASP.NET forms in the Dreamweaver design environment and then modify the properties of the form controls without hand-coding them.

Adding ASP.NET form controls to a page

You can use the visual design environment in Dreamweaver to add ASP.NET form controls to your page.

To add an ASP.NET form control to your page:

1. Open an ASP.NET page in Design view (View > Design).
2. Position the insertion point where you want the form control to appear on the page.

   Form controls must be inserted inside a form that has an `runat='server'` attribute in its tag. If your page does not have a form, Dreamweaver automatically creates one with a `runat='server'` attribute when you insert the first ASP.NET form control. If your page contains an HTML form, Dreamweaver automatically adds the `runat='server'` attribute to the existing form tag when you insert the first form control.
3. In the ASP.NET category of the Insert bar, select the form control's icon.
4. Complete the control's dialog box.

   For instructions, click the Help button on the dialog box. For more information on the properties of each control, see the documentation on the Microsoft website at http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpgenref/html/cpconaspsyntaxforwebcontrols.asp.

After you're done, Dreamweaver inserts the form control on your page. If you want, click the control on the page and change any of the properties in the Property inspector.

Setting the properties of ASP.NET form controls

You can modify the properties of ASP.NET form controls in the design environment.

To modify the properties of an ASP.NET form control in the design environment:

1. In Design view, select the form control on the page.

   The Property inspector displays properties of the control.
2. Modify the control's properties in the Property inspector.

   For more information on the properties of each control, see the documentation on the Microsoft website at http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpgenref/html/cpconaspsyntaxforwebcontrols.asp.
3. To set more properties, click the Tag Editor button in the Property inspector and set the properties in the tag editor that appears.

   The tag editor lets you set more properties, including events.

Creating a dynamic ASP.NET menu

You can dynamically populate an ASP.NET menu control such as DropDownList or ListBox with entries from a database.

Before you begin, you must define a DataSet or other source of dynamic content for the menu. For more information, see "Defining a recordset" on page 530.

To insert a dynamic menu in an ASP.NET page:

1. Open the ASP.NET page and position the insertion point where you want the menu to appear.
2. Select Insert > ASP.NET Objects; then select asp:DropDownList or asp:ListBox.

   The form control's dialog box appears.
3. Complete the dialog box and click OK.

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

Dreamweaver inserts the form control at the insertion point in your page. If you didn't insert a form, Dreamweaver inserts one that includes a `runat="server"` attribute.

Related topics

- "Setting the properties of ASP.NET form controls" on page 623
- "Making an existing ASP.NET menu dynamic" on page 624

Making an existing ASP.NET menu dynamic

You can make an existing ASP.NET menu object dynamic. ASP.NET menu objects include the DropDownList and ListBox controls.

Before you begin, you must define a DataSet or other source of dynamic content for the menu. For more information, see "Defining a recordset" on page 530.

To make an existing DropDownList or ListBox object dynamic:

1. In Design view, select the DropDownList or ListBox object you want to make dynamic.
2. In the Property inspector, click the List Items button.

   The List Items dialog box appears.
3. Select the From Database option and complete the dialog box.

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

Related topics

- "Creating a dynamic ASP.NET menu" on page 624
- "Making existing HTML form menus dynamic" on page 618

Displaying dynamic content in an ASP.NET TextBox control

You can display dynamic content in ASP.NET TextBox form controls.

Before you begin, you must define a DataSet or other source of dynamic content for the TextBox. For more information, see "Defining a recordset" on page 530.

To make an ASP.NET TextBox dynamic:

1. In Design view, select the TextBox on your page.
2. In the Property inspector, click the lightning bolt icon beside the Text text box.

   The Dynamic Data dialog box appears.
3. Select the recordset column that will supply a value to the TextBox object and click OK.

The TextBox control will display the dynamic content when the page is viewed in a browser.

Related topics

- "Displaying dynamic content in HTML text fields" on page 618

Dynamically preselecting ASP.NET CheckBox controls

You can let the server decide whether to select certain CheckBox controls when the ASP.NET page is loaded in a browser.

Before you begin, you must define a DataSet or other source of dynamic content for the checkboxes. For more information, see "Defining a recordset" on page 530. Ideally, the source of content should contain Boolean data, such as Yes/No or true/false.

To dynamically preselect an ASP.NET checkbox:

1. Select a CheckBox form control on the ASP.NET page.
2. In the Property inspector, select the Dynamic option.

   The Dynamic CheckBox dialog box appears.
3. Complete the dialog box and click OK.

   For instructions, click the Help button in the dialog box.

The checkbox will appear selected or not selected, depending on the data, when the page is loaded in a browser.

Related topics

- "Dynamically preselecting HTML checkboxes" on page 618

Dynamically preselecting an item in an ASP.NET RadioButtonList

You can let the server decide whether to select a radio button in a RadioButtonList control when the page is loaded in a browser.

Before you begin, you must insert at least one ASP.NET RadioButtonList control in your page. You must also define a DataSet or other source of dynamic content for the radio buttons. For more information, see "Defining a recordset" on page 530. Ideally, the source of content should contain Boolean data, such as Yes/No or true/false.

To dynamically preselect a radio button in a RadioButtonList:

1 In Design view, select the RadioButtonList control.

2 In the Property inspector, click List Items.

The List Items dialog box appears.

3 Select the From Database option and complete the dialog box.

For instructions, click the Help button in the dialog box.

Related topics

- "Dynamically preselecting HTML radio buttons" on page 619

PART VIII
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. You can quickly build any of these pages with Macromedia Dreamweaver MX.

This part contains the following chapters:

Chapter 40, “Building ColdFusion Applications Rapidly,” on page 629

Chapter 41, “Building ASP.NET Applications Rapidly,” on page 675

Chapter 42, “Building ASP and JSP Applications Rapidly,” on page 703

Chapter 43, “Building PHP Applications Rapidly,” on page 719

CHAPTER 40
Building ColdFusion Applications Rapidly

You can use the tools in Macromedia Dreamweaver MX 2004 to build a ColdFusion web application rapidly and with little or no coding.

About rapid application development (all servers)

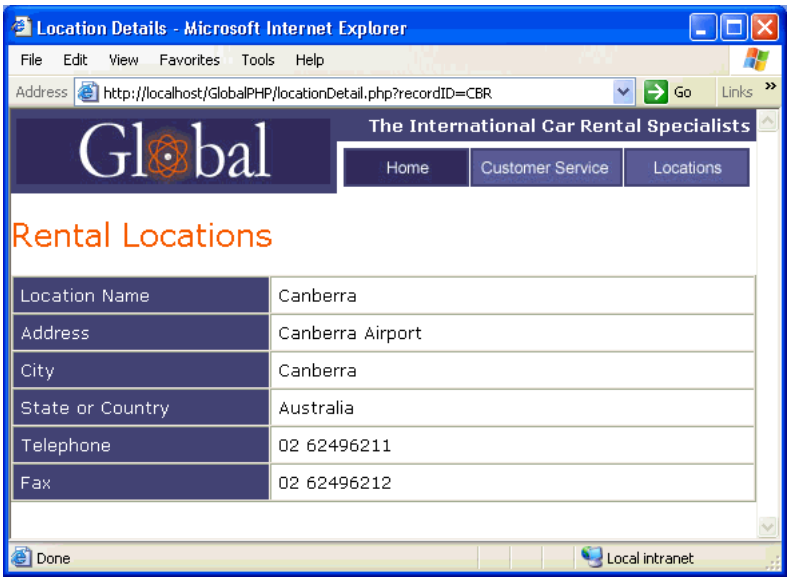
Rapid application development (RAD) is a software development process designed to make building applications easier and faster. RAD tools typically handle the details of a software project much like computer-assisted-design (CAD) tools take care of the details of drawing a wall or inserting a window in a building plan.

Dreamweaver is a RAD tool for website designers as well as web application developers. You can use Dreamweaver to build pages that search, insert, update, or delete records in a database. Dreamweaver handles the details of creating the pages; little or no coding is required from the developer. (Dreamweaver also provides a full-featured coding environment for developers who like to code all or part of their pages.)

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 a fictional company.

When a user clicks one of the linked location names, 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 "About search/results pages" on page 631.

A detail page can also be used to update or delete the record displayed.

Related topics

- "Building master/detail pages (ColdFusion)" on page 634
- "Building master/detail pages (ASP.NET)" on page 675
- "Building master/detail pages (ASP and JSP)" on page 703
- "Building master/detail pages (PHP)" on page 719

About search/results pages

In most cases, 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, which performs most of the work. The result's page does the following tasks:

- Reads the search parameters submitted by the search page
- Connects to the database and search 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 use ASP.NET, you can combine both the search page and the results page into one 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.

Dreamweaver inserts the SQL query in your page. When 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 might have information about 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, and 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.

Related topics

- "Building search/results pages (ColdFusion, ASP, JSP, PHP)" on page 641
- "Building a database search page (ASP.NET)" on page 681

About advanced database manipulation objects

Though you can use server behaviors to build pages that modify databases, you can also use database manipulation objects such as stored procedures, ASP command objects, or JSP prepared statements to build the pages.

Stored procedures

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.

Related topics

- "Using stored procedures to modify databases (ColdFusion)" on page 665
- "Using stored procedures to modify databases (ASP.NET)" on page 700
- "Using stored procedures to modify databases (ASP and JSP)" on page 714

ASP command objects

An ASP 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. For more information, see "Using ASP commands to modify a database" on page 716.

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();
```

For more information on how to create JSP prepared statements using rapid application development (RAD) tools in Dreamweaver, see "Using JSP prepared statements to modify a database" on page 718.

Building master/detail pages (ColdFusion)

With Dreamweaver, you can create sets of pages that present information in two levels of detail: a master page that lists records and a detail page that displays more detail about each record. This section describes how to build these kinds of master/detail pages.

Building master/detail pages in one operation (ColdFusion, ASP, JSP, PHP)

When developing web applications, you can quickly build master/detail pages using the Master/Detail Page Set application object. An application object lets you build a complete set of dynamic pages by completing only one or two dialog boxes.

The method you use is identical for ColdFusion, ASP, JSP, and PHP pages. For information on building ASP.NET master/detail pages, see "Building master/detail pages (ASP.NET)" on page 675.

To complete the master/detail page set with an application object:

1. In Dreamweaver, create a blank dynamic page by selecting File > New > Dynamic, selecting a dynamic page, and clicking Create.

   This page will become the master page.

2. Define a recordset for the page.

   You can define a recordset at design time (see "Defining a recordset" on page 530) or the user can define one at runtime (see "Building search/results pages (ColdFusion, ASP, JSP, PHP)" on page 641).

   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.

3 Open the master page in Design view and select Insert > Application Objects > Master Detail Page Set.

The Master Detail Page Set dialog box appears.

4 Complete the dialog box.

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

5 Click OK.

The application 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.

6 Customize the layout of the master and detail pages to suit your needs.

You can fully customize the layout of each page using the Dreamweaver page-design tools. You can also edit the server behaviors by double-clicking them in the Server Behaviors panel.

After creating master/detail pages with the application object, use the Server Behaviors panel (Window > Server Behaviors) to modify the various building blocks the application object inserts into the pages. For more information, see "Editing dynamic content on a page" on page 513.

You can also build master/detail pages using individual server behaviors.

Building master/detail pages block by block (ColdFusion)

You can add the basic building blocks of master/detail pages separately using the Server Behaviors panel.

You can also add the building blocks all at once using the Master/Detail Page Set application object. For more information, see "Building master/detail pages in one operation (ColdFusion, ASP, JSP, PHP)" on page 634.

This section covers the steps to building master/detail pages with server behaviors:

- "Creating the master page (ColdFusion)" on page 636
- "Creating the links to the detail page (ColdFusion)" on page 638
- "Creating a URL parameter for the links (ColdFusion)" on page 638
- "Finding and displaying the requested record on the detail page (ColdFusion)" on page 639

Creating the master page (ColdFusion)

This section describes how to create a master page that lists database records. You can use a dynamic table to list the records on a ColdFusion page.

Before you start, make sure you define a ColdFusion data source for your database. For more information, see Chapter 24, "Database Connections for ColdFusion Developers," on page 463.

To create a master page:

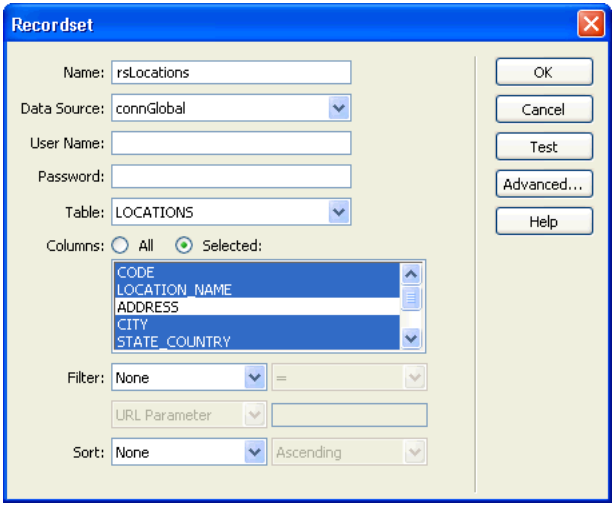
1 In Dreamweaver, create a new ColdFusion page.

Select File > New > Dynamic, select ColdFusion, and click Create. A blank ColdFusion page opens in Dreamweaver.

2 Define a recordset for the page.

In the Bindings panel, click the Plus (+) button, select Recordset (Query), and complete the Recordset dialog box. For more information, click the Help button in the dialog box. If you want to write your own SQL statement, click the Advanced button to open the advanced Recordset dialog box.

Make sure the recordset contains all the table columns you need to create your dynamic table. The recordset must also include the table column containing the unique key of each record—that is, the record ID column. In the following example, the CODE column contains information that uniquely identifies each record.



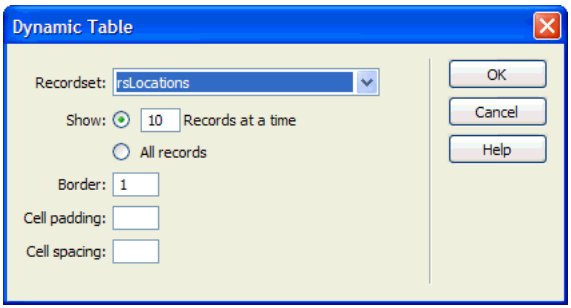
Typically, the recordset on the master page extracts a few columns from a database table while the recordset on the detail page extracts more columns from the same table to provide the extra detail.

The recordset can be defined by the user at runtime. For more information, see "Building search/results pages (ColdFusion, ASP, JSP, PHP)" on page 641.

3 Insert a dynamic table to display the records on the page.

Place the insertion point where you want the dynamic table to appear on the page, then selection Insert > Application Objects > Dynamic Data > Dynamic Table.

The Dynamic Table dialog box appears.



4 Complete the Dynamic Table dialog box and click OK.

Dreamweaver inserts a dynamic table on the page.

If you need help completing the dialog box, click the Help button in the dialog box.

5 If you want, delete the dynamic table column containing the record IDs.

If you don't want to show record IDs to users, you can delete the column from the dynamic table. Click anywhere on the page to move the focus to the page. Move the cursor near the top of the column in the dynamic table until the column cells are outlined in red, and then click to select the column. Press Delete to delete the column from the table.

The next step is to create the links to the detail page.

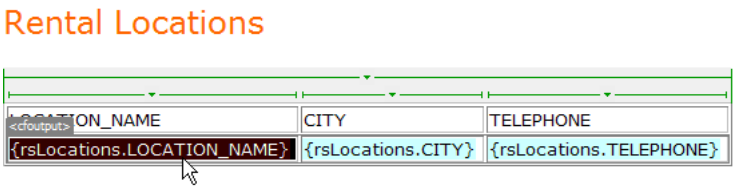
Creating the links to the detail page (ColdFusion)

After adding the dynamic table to the master page , you must create links that open the detail page. This section describes how to create the links. The next section describes how to modify the link so that it also passes the ID of the record that the user selects. The detail page will use this ID to find the requested record in the database and display it.

To create links to the detail page:

1 In the repeated row in the dynamic table, select the text or image that will serve as a link.

In the following example, the {rsLocations.LOCATION_NAME} placeholder is selected. The links will be applied to the location names in the column.



2 In the Property inspector, click the folder icon beside the Link text box.

3 Browse and select the detail page.

The detail page appears in the Link text box in the Property inspector.

In the dynamic table, the selected text appears linked. When the page runs on the server, the link is applied to the text in every table row.

The next step is to create a URL parameter that passes the record ID to the detail page.

Creating a URL parameter for the links (ColdFusion)

The links in the dynamic table not only have to open the detail page, they must pass the ID of the record the user selected. The detail page uses this ID to find the requested record in the database and displays it.

The record ID is passed to the detail page in a URL parameter. For more information, see "URL parameters" on page 518.

This section describes how to create a URL parameter that passes a record ID to the detail page.

To create the URL parameter:

1 On the master page, select the link in the dynamic table.

If Live Data view is turned on, select the link in the first row.

2 In the Link text box in the Property inspector, add the following string at the end of the URL:

```
?recordID=#recordsetName.fieldName#
```

The question mark tells the server that what follows is one or more URL parameters. The word recordID is the name of the URL parameter (you can make up any name you like). Make a note of the name of the URL parameter because you'll use it in the detail page later.

The expression after the equal sign is the value of the parameter. In this case, the value is generated by a ColdFusion expression that returns a record ID from the recordset. A different ID is generated for each row in the dynamic table. In the ColdFusion expression, replace recordsetName with the name of your recordset, and replace fieldName with the name of the field in your recordset that uniquely identifies each record. In most cases, the field will consist of a record ID number. In the following example, the field consists of unique location codes.

```
locationDetail.cfm?recordID=#rsLocations.CODE#
```

When the page runs, the values of the recordset's CODE field are inserted in the corresponding rows in the dynamic table. For example, if the Canberra, Australia, rental location has the code CBR, then the following URL will be used in the Canberra row in the dynamic table:

```
locationDetail.cfm?recordID=CBR
```

3 Save the page.

The next step is to find and display the requested record on the detail page.

Finding and displaying the requested record on the detail page (ColdFusion)

After completing the master page, switch to the detail page. You must find the requested record in the database and display it on the 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.

If you don't have a detail page yet, create a blank ColdFusion page (File > New).

2 In the Bindings panel (Window > Bindings), click the Plus (+) button and select Recordset (Query) from the pop-up menu.

The simple Recordset dialog box appears. If the advanced Recordset dialog box appears instead, click Simple to switch to the simple Recordset dialog box.

3 Name the recordset, then select a ColdFusion data source and the database table that will provide data to your recordset.

4 In the Columns 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, make sure the recordset on the detail page contains at least one column in common with the recordset on the master page. The common column is usually the record ID column, but it can also be the join field of related tables.

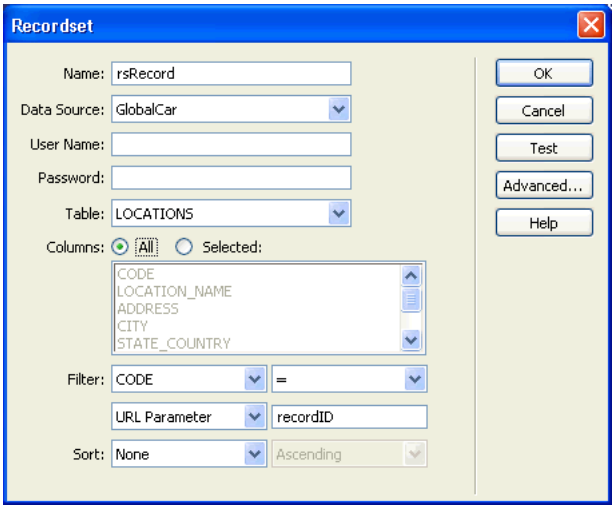
To include only some of the table's columns in the recordset, click Selected and choose the desired columns by Control-clicking (Windows) or Command-clicking (Macintosh) them in the list.

5 Complete the Filter section as follows to find and display the record specified in the URL parameter passed by the master page:

- From the first pop-up menu in the Filter area, select the column in the recordset 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, select the column containing record ID numbers. In the example discussed in the previous section, the recordset column called CODE contains the values that match the value of the URL parameter passed by the master page.
- 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 uses a URL parameter to pass information to the detail page.
- In the fourth text box, enter the name of the URL parameter passed by the master page. For example, if the URL that the master page used to open the detail page included the suffix locationDetail.cfm?recordID=CBR, then enter recordID.

The Recordset dialog box should look similar to the following:



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.

For more information, see"Making text dynamic" on page 547.

After uploading both the master and detail pages to the server, you can open the master page in a browser. After clicking a detail link on the master page, the detail page opens with more information about the selected record.

Building search/results pages (ColdFusion, ASP, JSP, PHP)

You can use Dreamweaver to build a set of pages to let users search your database. The method you use is identical for ColdFusion, ASP, JSP, and PHP pages. For information on building ASP.NET search/results pages, see "Building a database search page (ASP.NET)" on page 681.

Building the search page

A search page on the web typically contains form fields in which the user enters search parameters. 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 or a new page, and select Insert > Form > Form.

An empty form is created on the page. You might need 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 selecting Form from the Insert menu.

Form objects include text fields, 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.

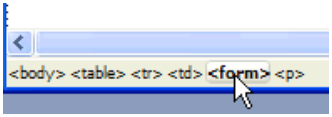
For more information on form objects, see "Inserting HTML form objects" on page 613.

3 Add a Submit button to the form (Insert > Form > Button).

4 If you want, 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 text box.

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 text box in 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, select 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.

Related topics

- "About search/results pages" on page 631

Building the results page

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.

Related topics

- "About search/results pages" on page 631
- "Building the search page" on page 641
- "Creating a detail page for a results page" on page 646

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 (see "Searching with multiple search parameters" on page 644).

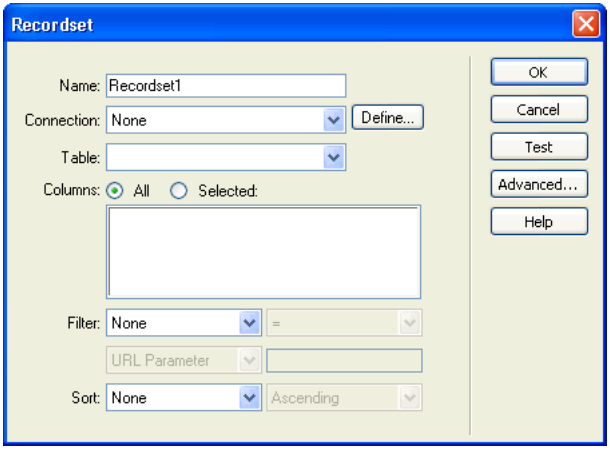
To create the recordset to hold the search results:

1 Open your results page in the Document window.

If you don't have a results page yet, create a blank dynamic page (File > New).

2 Create a new recordset by opening the Bindings panel (Window > Bindings), clicking the Plus (+) button, and selecting Recordset from the pop-up menu.

3 Make sure the simple Recordset 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 select 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 select the desired columns by Control-clicking (Windows) or Command-clicking (Macintosh) them in the list.

You should include only the columns containing information you want to display on the results page.

Leave the Recordset dialog box open for now. You'll use it next to retrieve 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 database table in which to search for a match.

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).

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.

The search page uses either a form variable or a URL parameter to pass information to the results page.

4 In the fourth text box, enter the name of the form object that accepts the search parameter on the search page.

The name of the object doubles as the name of the form variable or URL parameter. 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 object called Location that displays a list of countries. Here's how your Filter section should look:

5 If you want, 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 test 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 results" on page 645.

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.

Note: If you have only one search condition, you can use the simple Recordset dialog box to define your recordset (see "Searching with only one search parameter" on page 642).

To search for records in a database using SQL:

1. Open the results page in Dreamweaver, and then create a new recordset by opening the Bindings panel (Window > Bindings), clicking the Plus (+) button, and selecting Recordset from the pop-up menu.
2. Make sure the advanced Recordset 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 select 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.

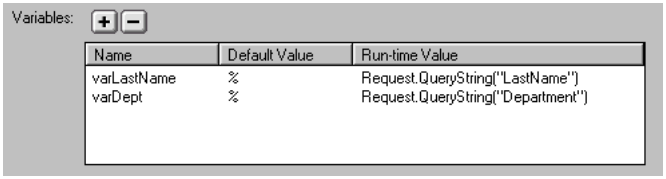
   Make sure the statement includes a WHERE clause with variables to hold the search parameters. In the following example, the variables are called `varLastName` and `varDept`:

   ```
   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 532.

   For help on SQL syntax, see Appendix B, "SQL Primer," on page 753.
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 a 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 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, 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.

Dreamweaver inserts the SQL query in your page.

The next step is to display the recordset on the results page.

Displaying the results

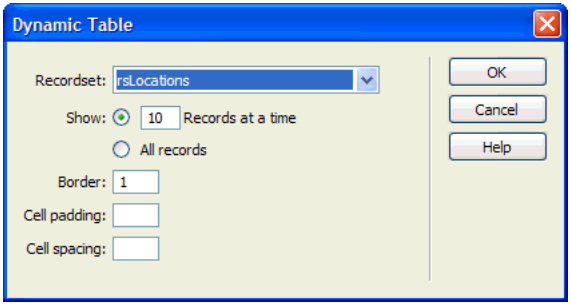
After creating a recordset to hold the search results, you must display the information on the results page. Displaying the records can be a simple matter of dragging individual columns from the Bindings panel to the results page. You can add navigation links to move forward and backward through the 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.

This section describes how to display the results using a dynamic table.

To display the search results using a dynamic table:

1 Place the insertion point where you want the dynamic table to appear on the results page, and select Insert > Application Objects > Dynamic Data > Dynamic Table.

The Dynamic Table dialog box appears.



2 Complete the Dynamic Table dialog box, selecting the recordset you defined to hold the search results.

If you need help completing the Dynamic Table dialog box, click the Help button in the dialog box.

3 Click OK.

Dreamweaver inserts a dynamic table that will display search results.

For more information on other methods of displaying dynamic content on a page, see Chapter 35, "Displaying Database Records," on page 555.

Creating a detail page for a results page

Your set of search/results pages can include a detail page to display more information about specific records on the results page. In this situation, your results page also doubles as the master page in a master/detail page set. For more information, see the following topics:

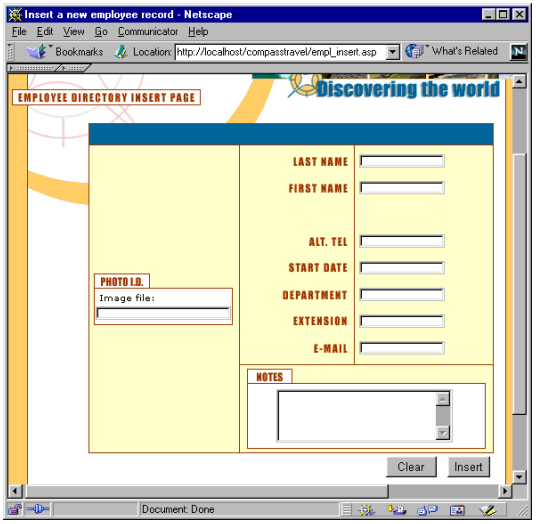
- "Building master/detail pages (ColdFusion)" on page 634
- "Building master/detail pages (ASP and JSP)" on page 703
- "Building master/detail pages (PHP)" on page 719

Related topics

- "Building the search page" on page 641
- "Building the results page" on page 642

Building a record insert page (all servers)

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 consists of 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 application object (see "Building the insert page in one operation" on page 647), or you can add them separately using the Dreamweaver form tools and the Server Behaviors panel (see "Building an insert page block by block" on page 648).

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 the insert page in one operation

You can add the basic building blocks of an insert page in a single operation using the Record Insertion Form application object. The application 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 648.

After placing the building blocks on the page, you can use the Dreamweaver design tools to customize the form, or the Server Behaviors panel to edit the Insert Record server behavior.

To build the insert page with the Record Insertion Form application object:

1 Open the page in Design view, and select Insert > Application Objects > Insert Record > Record Insertion Form Wizard.

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 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 application object. For more information, see "Building the insert page in one operation" on page 647.

The procedure to build the insert page consists of two steps:

- Adding an HTML form to the page to let users enter data
- Adding the Insert Record server behavior to insert records in a database table

To add an HTML form to an insert page:

1. Create a new dynamic 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 selecting Insert > Form > Form.

   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 text box.

   You don't need 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 such as a text field (Insert > Form > Text Field) 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 menus, checkboxes, and radio buttons.

   For more information on form objects, see "Inserting HTML form objects" on page 613.
5. Add a Submit button to the form (Insert > Form > Button).

   You can 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 text box.

To add a server behavior to insert records in a database table:

1. In the Server Behaviors panel (Window > Server Behaviors), click the Plus (+) button and select Insert Record from the pop-up menu.

   The Insert 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 insert records in a database table by clicking the Submit button on the form.

To edit the server behavior, open the Server Behaviors panel (Window > Server Behaviors) and double-click the Insert Record behavior.

Building pages to update a record (ColdFusion)

Your application can contain a set of pages that lets users update existing records in a database table. The pages normally consist of a search page, a results page, and an update page. The search and results page let users retrieve the record and the update page lets users modify the record.

This section describes the steps to build pages to update a record:

- "Searching for the record to update (ColdFusion)" on page 649
- "Creating the links to open the update page (ColdFusion)" on page 650
- "Creating a URL parameter for update links (ColdFusion)" on page 651
- "Retrieving the record to update (ColdFusion)" on page 652
- "Completing the update page in one operation (ColdFusion)" on page 652
- "Completing the update page block by block (ColdFusion)" on page 653

Related topics

- "Building master/detail pages (ColdFusion)" on page 634
- "Building a record insert page (all servers)" on page 646
- "Building pages to delete a record (ColdFusion)" on page 655

Searching for the record to update (ColdFusion)

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. The user enters search criteria in the search page and selects the record on the results page. When the user clicks the record, the update page opens and displays the record in an HTML form.

For instructions on creating pages to search for the record to update, see "Building search/results pages (ColdFusion, ASP, JSP, PHP)" on page 641.

After creating the search/results pages, the next step is to create links on the results page that open the update page when clicked.

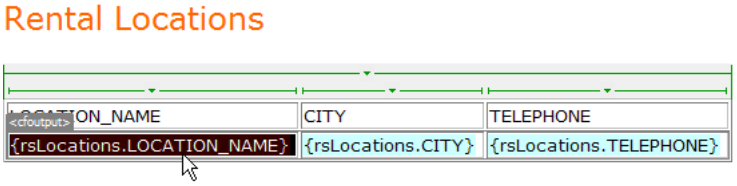
Creating the links to open the update page (ColdFusion)

After creating the search/results pages, you must create links on the results page to open the update page and display the selected record in an HTML form. This section describes how to create the links. The next section describes how to modify the link so that it also passes the ID of the record the user selects. The update page will use this ID to find the requested record in the database and display it.

To create links to the update page:

1 On the results page, select the repeated text or dynamic content placeholder you want to be linked.

In the following example, the {rsLocations.LOCATION_NAME} placeholder is selected. The links will be applied to the location names in the column.



2 In the Property inspector, click the folder icon beside the Link text box.

3 Browse and select the update page.

The filename of the update page appears in the Link text box.

On the results page, the placeholder for the dynamic content appears linked. If you upload the pages to the server and run a search, you can see that the link is applied to every record listed on the results page.

The next step is to create a URL parameter to pass the ID of the record the user selected.

Creating a URL parameter for update links (ColdFusion)

The links on the results page not only have to open the update page, they must pass the ID of the record the user selected. The update page will use this ID to find the requested record in the database and display it.

The record ID is passed to the update page in a URL parameter. For more information, see "URL parameters" on page 518.

This section describes how to create a URL parameter that passes a record ID to the update page.

To create the URL parameter:

1 Select the link on the results page.

2 In the Link text box in the Property inspector, add the following string at the end of the URL:

```
?recordID=#recordsetName.fieldName#
```

The question mark tells the server that what follows is one or more URL parameters. The word `recordID` is the name of the URL parameter (you can make up any name you like). Make a note of the name of the URL parameter because you'll use it in the update page later.

The expression after the equal sign is the value of the parameter. In this case, the value is generated by a ColdFusion expression that returns a record ID from the recordset. A different ID is generated for each row in the dynamic table. In the ColdFusion expression, replace `recordsetName` with the name of your recordset, and replace `fieldName` with the name of the field in your recordset that uniquely identifies each record. In most cases, the field will consist of a record ID number. In the following example, the field consists of unique location codes.

```
locationDetail.cfm?recordID=#rsLocations.CODE#
```

When the page runs, the values of the recordset's CODE field are inserted in the corresponding rows in the dynamic table. For example, if the Canberra, Australia, rental location has the code CBR, then the following URL will be used in the Canberra row in the dynamic table:

```
locationDetail.cfm?recordID=CBR
```

3 Save the page.

The next step is to modify the update page so that it can find the requested record in the database and display it on the page.

Retrieving the record to update (ColdFusion)

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 Create a new ColdFusion page in Dreamweaver and save it.

The page will become your update page.

2 In the Bindings panel (Window > Bindings), click the Plus (+) button and select Recordset.

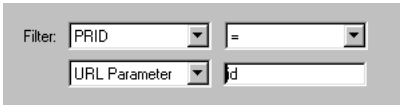
If the advanced dialog box appears, click the Simple button to open the simple dialog box.

3 Name the recordset and specify where the data you want to update is located using the Connection and Table pop-up menus.

4 Click the Selected option and select a key column (usually the record ID column) and the columns containing the data to be updated.

5 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:



6 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 in one operation (ColdFusion)

An update page has three building blocks:

- A filtered recordset to retrieve the record from a database table (see "Retrieving the record to update (ColdFusion)" on page 652)
- 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 final two building blocks of an update page in a single operation using the Record Update Form application object. The application object adds both an HTML form and an Update Record server behavior to the page.

Before you can use the application object, your web application must be able to identify the record to update, and your update page must be able to retrieve it. See "Searching for the record to update (ColdFusion)" on page 649, "Creating the links to open the update page (ColdFusion)" on page 650, "Creating a URL parameter for update links (ColdFusion)" on page 651 and "Retrieving the record to update (ColdFusion)" on page 652.

After the application 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.

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.

To build the update page with the Record Update Form application object:

1 Open the page in Design view, then select Insert > Application Objects > Update Record > Record Update Form Wizard.

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 application 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 (ColdFusion)

An update page has three building blocks:

- A filtered recordset to retrieve the record from a database table (see "Retrieving the record to update (ColdFusion)" on page 652)
- 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 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 "Searching for the record to update (ColdFusion)" on page 649, "Creating the links to open the update page (ColdFusion)" on page 650, "Creating a URL parameter for update links (ColdFusion)" on page 651 and "Retrieving the record to update (ColdFusion)" on page 652.

Completing the update page consists of three tasks:

- Adding an HTML form to the page to let users modify the data
- Displaying the record in the form by binding the form objects to database table columns
- Adding the Update Record server behavior to update the database table after the user modifies the record

To add an HTML form to an update page:

1 Create a new Coldfusion 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 selecting Insert > Form > Form.

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 text 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 such as a text field (Insert > Form > Text Field) 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 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 "Inserting HTML form objects" on page 613.

6 Add a Submit button to the form (Insert > Form > Button).

You can 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 text box.

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 (ColdFusion)" on page 652.

2 Bind each form object to data in the recordset, as described in the following sections:

- "Displaying dynamic content in HTML text fields" on page 618
- "Dynamically preselecting HTML checkboxes" on page 618
- "Dynamically preselecting HTML radio buttons" on page 619
- "Creating a dynamic HTML form menu" on page 617
- "Making existing HTML form menus dynamic" on page 618

To add a server behavior to update the database table:

1 In the Server Behaviors panel (Window > Server Behaviors), click the Plus (+) button and select 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 clicking the Submit button on the form.

To edit the server behavior, open the Server Behaviors panel (Window > Server Behaviors) and double-click the Update Record behavior.

Related topics

- "Completing the update page in one operation (ColdFusion)" on page 652

Building pages to delete a record (ColdFusion)

Your application can contain a set of pages that lets users delete records in a database. The pages normally consist of a search page, a results page, and a delete page. The search and results page let users retrieve the record and the delete page lets users delete the record.

This section describes the following steps to build ColdFusion pages to delete records:

- "Searching for the record to delete (ColdFusion)" on page 655
- "Creating links to a confirmation page (ColdFusion)" on page 656
- "Creating a URL parameter to pass to the confirmation page (ColdFusion)" on page 658
- "Displaying the record on the confirmation page (ColdFusion)" on page 659
- "Adding logic to delete the record (ColdFusion)" on page 663

Searching for the record to delete (ColdFusion)

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. The user enters search criteria in the search page and selects the record on the results page. When the user clicks the record, the delete page opens and displays the record in an HTML form.

For instructions on creating pages to search for the record to delete, see "Building search/results pages (ColdFusion, ASP, JSP, PHP)" on page 641.

After creating the search/results pages, the next step is to create links on the results page to let users select a record to delete and display a confirmation page.

Creating links to a confirmation page (ColdFusion)

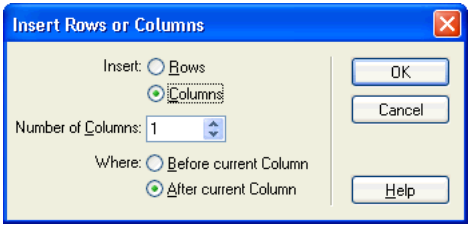
After creating the results page, you must create links that the user can click to confirm the deletion of the associated record from the database. This section describes how to create the links. The next section describes how to modify the link so that it also passes the ID of the record that the user wants to delete. The confirmation page uses this ID to find and display the record.

To create links to a confirmation page:

1. On the results page, create a new column in the table used to display records by clicking inside the last table column and selecting Modify > Table > Insert Rows or Columns.

   The Insert Rows or Columns dialog box appears.

2. Select the Columns option, then select the After Current Column option and click OK.



   Dreamweaver adds a column to the table.

3. In the newly created table column, enter the string `Delete` in the row containing the dynamic content placeholders. Make sure you enter the string inside the tabbed repeat region.

   You can also insert an image with a word or symbol for delete.

   If Live Data view is turned on, enter the string in the first row of records and click the Refresh icon.

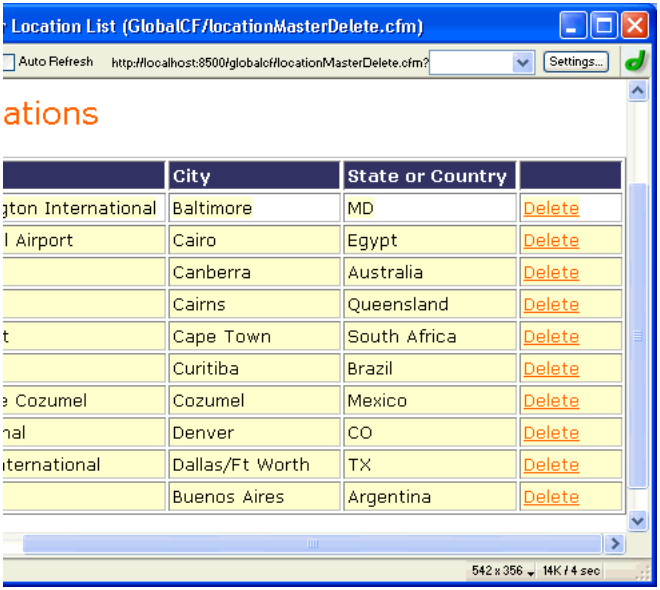
4. Select the `Delete` string to apply a link to it.

   If Live Data view is turned on, select the string in the first row of records.

5. In the Property inspector, enter the confirmation page in the Link text box.

   You can enter any filename you want.

After clicking outside the Link text box, the `Delete` string appears linked in the table. If you turn on Live Data view (View > Live Data), you can see that the link is applied to the same text in every table row. If Live Data view is already turned on, click the Refresh icon to apply the links to each row.



After creating the links, the next step is to create URL parameters for the links.

Creating a URL parameter to pass to the confirmation page (ColdFusion)

The links on the results page not only have to open the confirmation page, they must pass the ID of the record the user wants to delete. The confirmation page will use this ID to find the record in the database and display it.

You must pass the record ID to the confirmation page with a URL parameter. This section describes how to create a URL parameter to pass the record ID to the confirmation page.

To create the URL parameter:

1 Select the Delete link on the results page.

If Live Data view is turned on, select the link in the first row.

2 In the Link text box in the Property inspector, add the following string at the end of the URL:

```
?recordID=#recordsetName.fieldName#
```

The question mark tells the server that what follows is one or more URL parameters. The word recordID is the name of the URL parameter (you can make up any name you like). Note the name of the URL parameter because you'll use it in the confirmation page later.

The expression after the equal sign is the value of the parameter. In this case, the value is generated by a ColdFusion expression that returns a record ID from the recordset. A different ID is generated for each row in the dynamic table. In the ColdFusion expression, replace `recordsetName` with the name of your recordset, and replace `fieldName` with the name of the field in your recordset that uniquely identifies each record. In most cases, the field will consist of a record ID number. In the following example, the field consists of unique location codes:

```
confirmDelete.cfm?recordID=#rsLocations.CODE#
```

When the page runs, the values of the recordset's CODE field are inserted in the corresponding rows in the dynamic table. For example, if the Canberra, Australia, rental location has the code CBR, then the following URL will be used in the Canberra row in the dynamic table:

```
confirmDelete.cfm?recordID=CBR
```

3 Save the page.

After creating a dynamic URL parameter for the delete links, the next step is to display the record on the confirmation page.

Displaying the record on the confirmation page (ColdFusion)

After completing the page listing the records, switch to the confirmation page. The confirmation page shows the record and asks the user if they're sure they want to delete it. When the user confirms the operation by clicking the form button, the web application deletes the record from the database.

Building this page consists of creating an HTML form, retrieving the record to display in the form, displaying the record in the form, and adding the logic to delete the record from the database. Retrieving and displaying the record consists of defining a recordset to hold a single record—the record the user wants to delete—and binding the recordset columns to the form. The steps are outlined in more detail below.

To create an HTML form to display the record:

1 Create a new ColdFusion page and save it as the confirmation page you specified in the previous section.

You specified a confirmation page when you created the Delete link in the previous section. Use this name when saving the file for the first time (for example, deleteConfirm.cfm).

2 Insert an HTML form on the page (Insert > Form > Form).

3 Add a hidden form field to the form.

The hidden form field is required to store the record ID passed by the URL parameter. To add a hidden field, place the insertion point in the form and select Insert > Form > Hidden Field.

4 Add a button to the form.

The user will click the button to confirm and delete the displayed record. To add a button, place the insertion point in the form and select Insert > Form > Button.

5 Enhance the design of the page any way you want and save it.

To retrieve the record the user wants to delete:

1 In the Bindings panel (Window > Bindings), click the Plus (+) button and select Recordset (Query) from the pop-up menu.

The simple Recordset dialog box appears. If the advanced Recordset dialog box appears instead, click Simple to switch to the simple Recordset dialog box.

2 Name the recordset, then select a ColdFusion data source and the database table that contains the records that users can delete.

3 In the Columns area, select the table columns (record fields) you want to display on the page.

To display only some of the record's fields, click Selected and choose the desired fields by Control-clicking (Windows) or Command-clicking (Macintosh) them in the list.

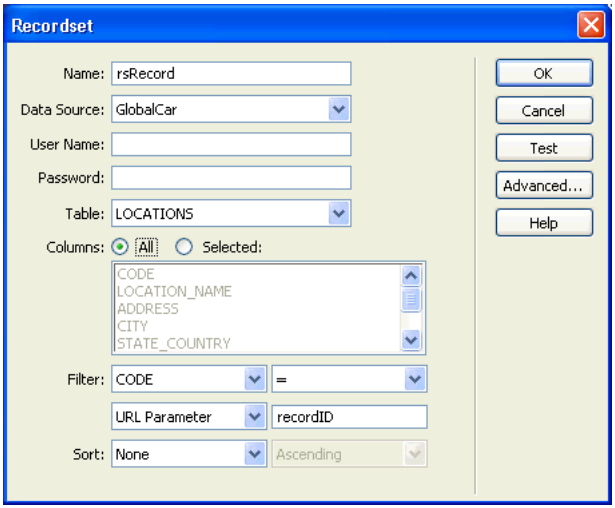
Make sure to include the record ID field even if you won't be displaying it.

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 recordset containing values that match the value of the URL parameter passed by the page with the Delete links. For example, if the URL parameter contains a record ID number, select the column containing record ID numbers. In the example discussed in the previous section, the recordset column called CODE contains the values that match the value of the URL parameter passed by the page with the Delete links.
- From the pop-up menu beside the first menu, select the equal sign, if not already selected.
- From the third pop-up menu, select URL Parameter. The page with the Delete links uses a URL parameter to pass information to the confirmation page.
- In the fourth text box, enter the name of the URL parameter passed by the page with the Delete links.

  For example, if the URL used to open the confirmation page included the suffix confirmDelete.cfm?recordID=CBR, then enter recordID.

  The Recordset dialog box should look as follows.



5 Click OK.

The recordset appears in the Bindings panel.

To display the record the user wants to delete:

1 Select the recordset columns in the Bindings panel and drag them to the confirmation page.

Make sure you insert this read-only dynamic content within the form boundaries. For more information on insert dynamic content in a page, see "Making text dynamic" on page 547.

Next, you must bind the record ID column to the hidden form field.

2 Make sure Invisible Elements are turned on (View > Visual Aids > Invisible Elements), then click the yellow shield icon representing the hidden form field.

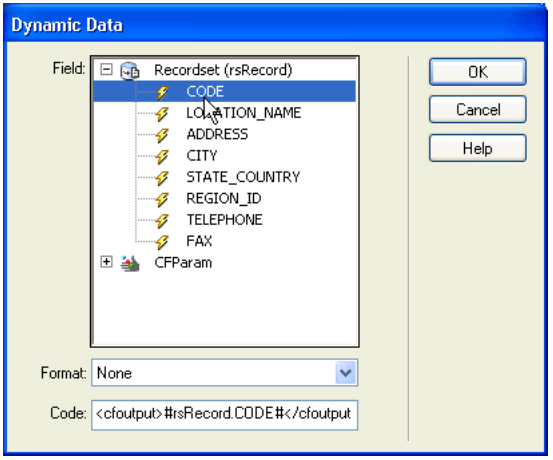
The hidden form field is selected.

3 In the Property inspector, click the lightning bolt icon beside the Value text box.

The Dynamic Data dialog box appears.

4 In the Dynamic Data dialog box, select the record ID column in the recordset.

In the following example, the record ID column selected is CODE.



5 Click OK and save the page.

The completed confirmation page should look similar to the following.

After displaying the record on the confirmation page, the next step is to add logic to delete the record.

Adding logic to delete the record (ColdFusion)

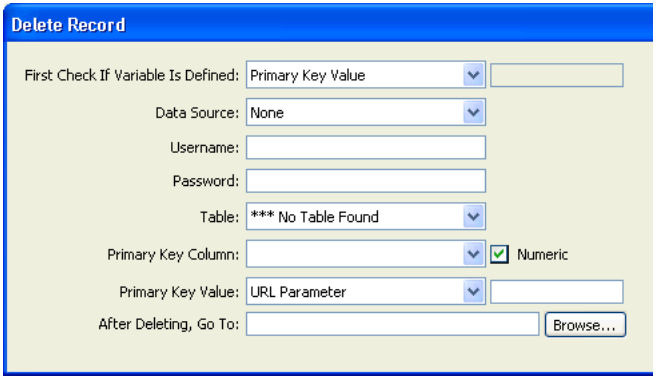
After displaying the selected record on the confirmation page, you must add logic to the page that deletes the record from the database when the user clicks the Confirm Deletion button. You can add this logic quickly and easily in Dreamweaver with the Delete Record server behavior.

To add logic to delete the record displayed in the HTML form:

1 Make sure the confirmation page is open in Dreamweaver.

2 In the Server Behaviors panel (Window > Server Behaviors), click the Plus (+) button and select Delete Record.

The Delete Record dialog box appears.



3 In the First Check If Variable Is Defined text box, make sure Primary Key Value is selected.

You specify the primary key value later in the dialog box.

4 In the Connection pop-up menu, select a connection to the database so that the server behavior can connect to the affected database.

5 In the Table pop-up menu, select the database table that contains the records that will be deleted.

6 In the Primary Key Column pop-up menu, select the table column that contains record IDs.

The Delete Record server behavior will search this column for a match. The column should contain the same record ID data as the recordset column you bound to the hidden form field on the page.

If the record ID is numeric, select the Numeric option.

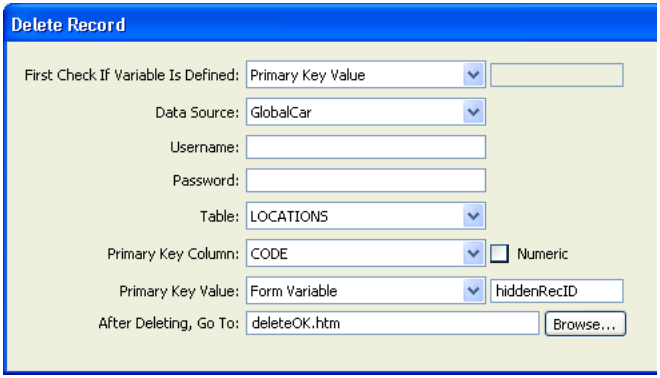
7 In the Primary Key Value pop-up menu, select the variable on your page that contains the record ID identifying the record to be deleted.

The variable is created by your hidden form field. It has the same name as the name attribute of the hidden field and is either a form or URL parameter, depending on the form's method attribute. In our example, the variable is a form variable called hiddenRecID.

8 In the After Deleting, Go To text box, specify a page to open after the record has been deleted from the database table.

You can specify a page that contains a brief success message to the user, or a page listing the remaining records so that the user can verify that the record has been deleted.

The completed Delete Record dialog box should look as follows.



9 Click OK and save your work.

Upload the pages to your web server, open a browser and search for a record to delete. When you click a Delete link on the results page, the confirmation page appears. Click the Confirm button to delete the record from the database. To verify that the record has been deleted, open the page with the Delete links again. The record should no longer appear in the list.

Note: Click Refresh if the record still appears on the page.

Using stored procedures to modify databases (ColdFusion)

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.

Note: mySQL and Microsoft Access databases do not support stored procedures.

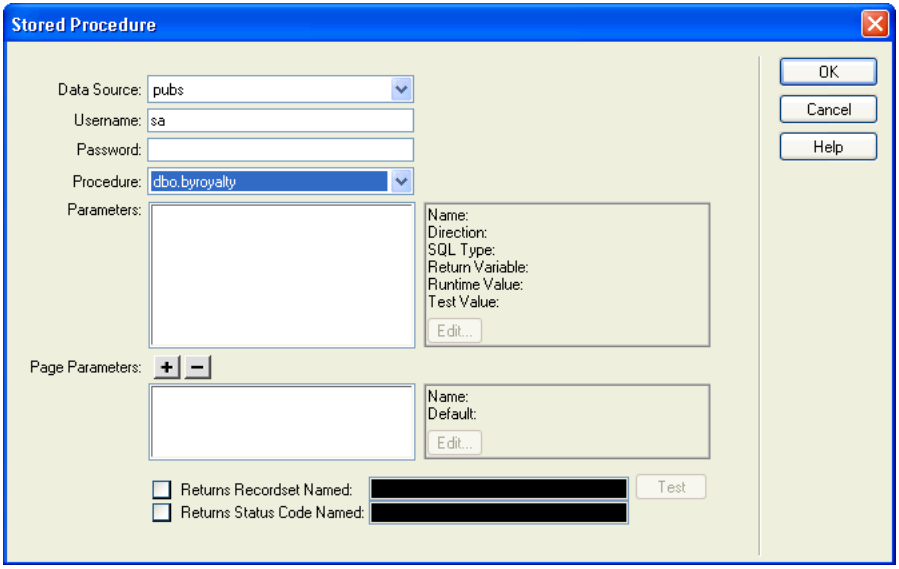
Before you use a stored procedure to modify a database, make sure the stored procedure contains SQL that modifies the database in some way. To create and store one in your database, consult your database documentation and a good Transact-SQL manual.

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 (Window > Bindings), 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 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.

Related topics

- "Stored procedures" on page 632

Building pages that restrict access to your site (ColdFusion, ASP, JSP, PHP)

You can use Dreamweaver to build the following pages to restrict access to your site:

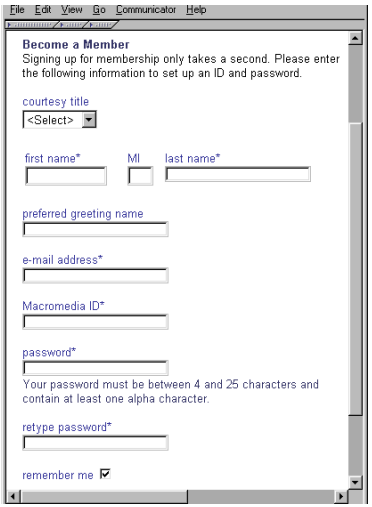
- A page that requires users to register the first time they visit the site (see "Building a registration page" on page 666)
- A page that lets registered users log in to the site (see "Building a login page" on page 669)
- Pages that only authorized users can view (see "Building a page only authorized users can access" on page 671)

The method you use is identical for ColdFusion, ASP, JSP, and PHP pages. Dreamweaver does not have authentication server behaviors for ASP.NET pages.

Building a registration page

Your web application can contain a page that requires users to register the first time they visit your site.

For example, the following page asks first-time users to register:



A registration page is made up of the following building blocks:

- A database table to store login information about the users (see "Storing login information about users" on page 667)
- An HTML form that lets users select a user name and password (see "Letting users choose a user name and password" on page 667)

  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 (see "Updating the table of users in the database" on page 668)
- A Check New Username server behavior to make sure the user name entered by the user is not taken by another user (see "Making sure the chosen user name is unique" on page 668)

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 672).

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.

The next step in creating a registration page is to add an HTML form to the registration page to let users choose a user name and password (if applicable).

Related topics

- "Building a registration page" on page 666

Letting users choose a user name and password

You add an HTML form to the registration page to let users select 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 selecting 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 text 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 668).
4. Add text fields (Insert > Form > 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 "Inserting HTML form objects" on page 613.
5. Add a Submit button to the form (Insert > Form > Button).

   You can 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 text box.

The next step in creating a registration page is to add the Insert Record server behavior to insert records in the table of users in the database.

Related topics

- "Building a registration page" on page 666

Updating the table of users in the database

You must add an Insert Record server behavior to the registration page 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 select Insert Record from the pop-up menu.

The Insert Record dialog box appears.

2 Complete the dialog box, making sure to specify the table of users in the database into which the user data will be inserted.

For instructions, click the Help button in the dialog box.

3 Click OK.

The final step in creating a registration page is to make sure the user name is not used by another registered user.

Related topics

- "Building a registration page" on page 666

Making sure the chosen user name is unique

You can add a server behavior to make sure the user name entered is not taken by another registered user.

To make sure the chosen user name is unique:

1 In the Server Behaviors panel (Window > Server Behaviors), click the Plus (+) button and select 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.

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).

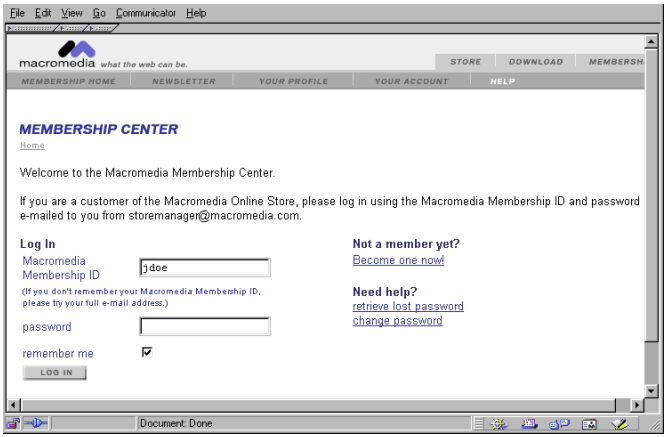
Related topics

- "Building a registration page" on page 666

Building a login page

Your web application can contain a page that lets registered users log in to the site.

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 (see "Creating a database table of registered users" on page 669)
- An HTML form to let users enter a user name and password (see "Letting users log in" on page 670)
- A Log In User server behavior to make sure the entered user name and password are valid (see "Checking the user name and password" on page 670)

A session variable consisting of the user name is created for the user when the user logs in successfully.

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 666.

The next step in building a login page is to add an HTML form to the page to let users log in.

Related topics

- "Building a login page" on page 669

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 text 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 670).

4 Add a user name and a password text field (Insert > Form > 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 > Button).

You can 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 text box.

The next step in building a login page is to add the Log In User server behavior to make sure the entered user name and password are valid.

Related topics

- "Building a login page" on page 669

Checking the user name and password

You must add a Log In User server behavior to the login page to make sure the user name and password a user enters are valid.

To check the user name and password:

1 In the Server Behaviors panel (Window > Server Behaviors), click the Plus (+) button and select 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.

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).

Related topics

- "Building a login page" on page 669

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 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. For more information, see "Redirecting unauthorized users to another page" on page 672.

If you do plan to use authorization levels, you can protect any page on your site with the following building blocks:

- A Restrict Access To Page server behavior to redirect unauthorized users to another page (see "Redirecting unauthorized users to another page" on page 672)
- An extra column in your users database table to store each user's access privileges (see "Storing access privileges in the user database" on page 672)

Whether you use authorization levels or not, 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 673.

Redirecting unauthorized users to another page

To prevent unauthorized users from accessing a page, 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 select 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 select 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 select Paste from the pop-up menu.
5. Repeat steps 3 and 4 for each page you want to protect.

Related topics

- "Storing access privileges in the user database" on page 672

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.

Related topics

- "Redirecting unauthorized users to another page" on page 672

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. Select text or an image on a page to serve as the link.
2. In the Server Behaviors panel (Window > Behaviors), click the Plus (+) button and select 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 select User Authentication > Log Out User.

   The Log Out User dialog box appears.
3. Select the Log Out When Page Loads option.
4. Click OK.

Related topics

- "Building a page only authorized users can access" on page 671

CHAPTER 41
Building ASP.NET Applications Rapidly

You can use the tools in Macromedia Dreamweaver MX 2004 to build an ASP.NET web application rapidly and with little or no coding.

Related topics

- "About rapid application development (all servers)" on page 629

Building master/detail pages (ASP.NET)

With Dreamweaver, you can create sets of pages that present information in two levels of detail: a master page that lists records and a detail page that displays more detail about each record. This section describes how to build these kinds of master/detail pages.

Related topics

- "Enhancing the functionality of a dynamic page" on page 510

Creating the master page (ASP.NET)

This section describes how to create a master page that lists database records. With ASP.NET, you can use a DataGrid to list the records on the page.

Before you start, make sure you define a database connection for the site. For more information, see Chapter 25, "Database Connections for ASP.NET Developers," on page 465.

To create a master page:

1 Create a new ASP.NET page.

Select File > New > Dynamic, select an ASP.NET page format, and click Create. A blank .aspx page opens in Dreamweaver.

2 Define a DataSet for the page.

In the Bindings panel (Window > Bindings), click the Plus (+) button, select DataSet (Query), and complete the dialog box. For more information, click the Help button on the dialog box. If you want to write your own SQL statement, click the Advanced button to open the advanced DataSet dialog box.

Make sure the DataSet contains all the table columns you need to create your master page.

The DataSet must also include the table column containing the unique key of each record—that is, the record ID column.

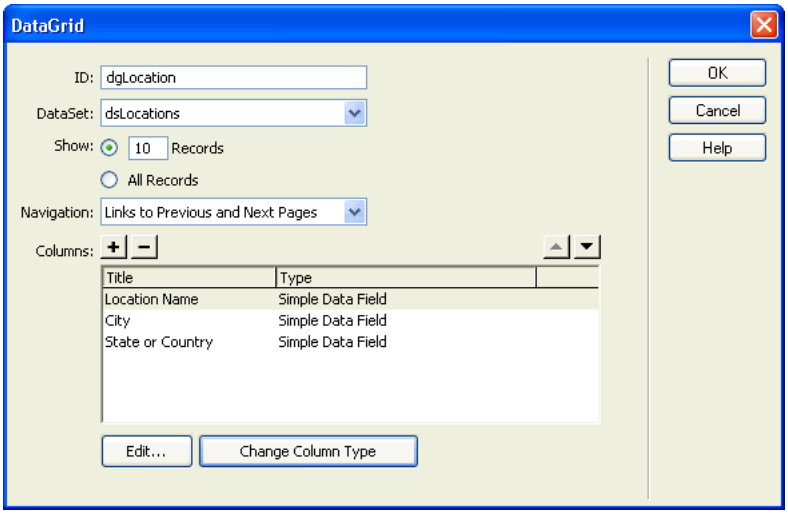
Typically, the DataSet on the master page extracts a few columns from a database table while the DataSet on the detail page extracts more columns from the same table to provide the extra detail.

The DataSet can be defined by the user at runtime. For more information, see "Building a database search page (ASP.NET)" on page 681.

3 Create a DataGrid on the page.

In the Server Behaviors panel (Window > Server Behaviors), click the Plus (+) button, select DataGrid, and complete the DataGrid dialog box. For more information, click the Help button on the dialog box. For now, accept the default column type of Simple Data Field for each column in your DataGrid. In the next section, you'll change the type of one column to Hyperlink (see "Opening a detail page and passing a record ID (ASP.NET)" on page 677).

Here's an example of a DataGrid dialog box that defines three columns.



The next step in creating master/detail pages is to create links that open the detail page and pass the ID of the record the user selected.

Opening a detail page and passing a record ID (ASP.NET)

After adding a DataGrid to the master page, you must create links that open the detail page and pass the ID of the record that the user selected. The detail page will use this ID to find the requested record in the database and display it.

The record ID is passed to the detail page in a URL parameter. For more information, see "URL parameters" on page 518.

In a DataGrid, you create these links by turning a column in the DataGrid into a hyperlink column and setting its attributes, as described in this section.

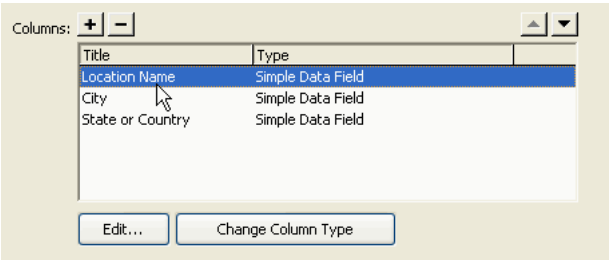
To open the detail page and pass a URL parameter that contains the ID of the record the user clicked:

1 Open the DataGrid dialog box you created in the previous section.

To open the dialog box, double-click your DataGrid in the Server Behaviors panel (Window > Server Behaviors).

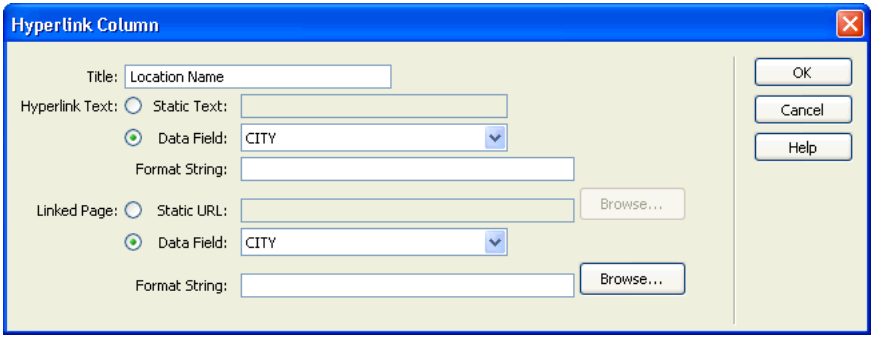
The DataGrid dialog box opens.

2 In the Columns box, select the column you want to apply links to.



3 Click the Change Column Type button and select Hyperlink from the pop-up menu.

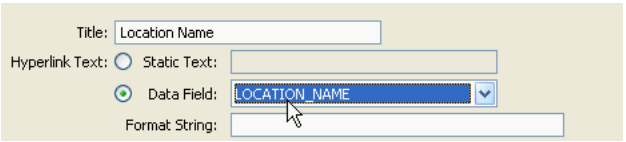
The Hyperlink Column dialog box appears.



4 In the Hyperlink Text area, specify the text to be displayed in the hyperlink column.

If you want to create a generic link such as *Details* for each row in the DataGrid, select the Static Text option and enter the text for the link. Each row in the DataGrid will display the same text (such as *Details*) in the hyperlink column.

If you want to add links to data displayed in the column, select the Data Field option and select a data field in your DataSet. The data field provides the text for the links in your hyperlink column. In the following example, each row in the DataGrid will display a location name.



5 In the Linked Page area, build the URL to apply to the text in the hyperlink column.

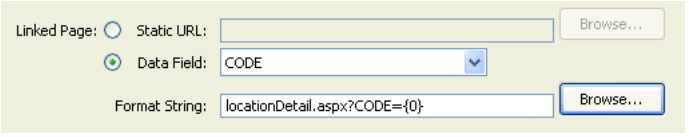
The URL not only has to open the detail page, it must uniquely identify the record to display on that page.

To identify the record to display on the detail page, select the Data Field option and select a field in your DataSet that uniquely identifies each record. In most cases, the field will consist of a record ID number. In the following example, the field consists of unique location codes.

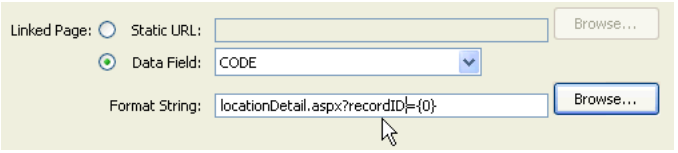
6 In the Format String text box in the Linked Page area, click the Browse button then locate and select your detail page.

Dreamweaver creates a URL to the detail page that includes a URL parameter identifying the record the detail page should display. Make a note of the name of the URL parameter because you'll use it in the detail page later.

For example, if you select locationDetail.aspx as your detail page, Dreamweaver creates the following URL:



In this case, Dreamweaver creates a URL parameter called `CODE`. Dreamweaver copies the name of the data field, but you don't have to use that name. You can change it to something more descriptive, such as recordID, as in the following example.



The `{0}` element is a placeholder corresponding to the data field's value. When the page runs, the values of the DataSet's CODE field are inserted in the corresponding rows in the DataGrid. For example, if the Canberra, Australia, rental location has the code CBR, then the following URL will be used in the Canberra row in the DataGrid:

```
locationDetail.aspx?recordID=CBR
```

7 Click OK to close the Hyperlink dialog box; then click OK to close the DataGrid dialog box.

Dreamweaver updates the DataGrid on your page.

The next step in creating master/detail pages is to modify the detail page so that it can find the requested record in the database and display it on the page.

Finding and displaying the requested record on the detail page (ASP.NET)

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 DataSet on the detail page to hold a single record—the record requested by the master page—and binding the DataSet columns to the page.

To find and display the requested record on the detail page:

1 Switch to the detail page.

If one doesn't exist, select File > New > Dynamic, select an ASP.NET page format, and click Create. A blank .aspx page opens in Dreamweaver.

2 In the Bindings panel (Window > Bindings), click the Plus (+) button and select DataSet (Query) from the pop-up menu.

The simple DataSet dialog box appears. If the advanced DataSet dialog box appears instead, click Simple to switch to the simple DataSet dialog box.

3 Name the DataSet, and select a connection and database table that will provide data to your DataSet.

4 In the Columns area, select the table columns to include in the DataSet.

The DataSet can be identical to or different from the DataSet on the master page. Usually a detail page DataSet has more columns to display more detail.

If the DataSets are different, make sure the DataSet on the detail page contains at least one column in common with the DataSet on 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 DataSet, select the Selected option and select 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 master page:

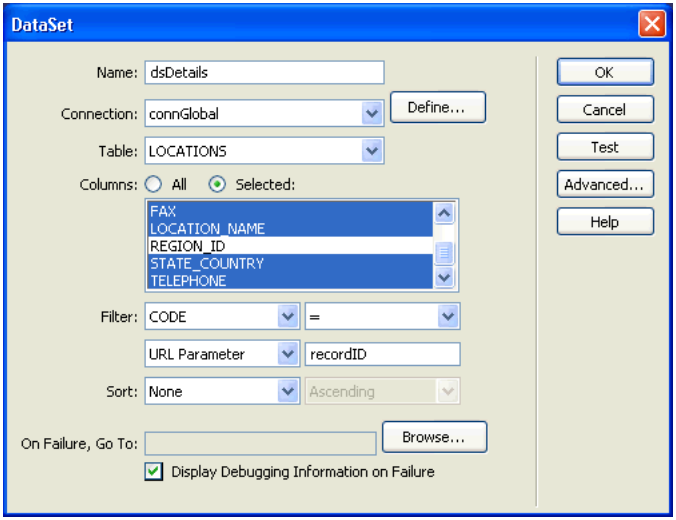
- From the first pop-up menu in the Filter area, select the column in the DataSet containing values that match the value of the URL parameter passed by the master page.

  For example, if the URL parameter contains a record ID number, select the column containing record ID numbers. In the example discussed in the previous section, the DataSet column called CODE contains the values that match the value of the URL parameter passed by the master page.
- Select the equal sign (if it is not already selected) from the pop-up menu beside the first menu.
- Select URL Parameter from the third pop-up menu.

  The master page uses a URL parameter to pass information to the detail page.
- Enter the name of the URL parameter passed by the master page in the fourth box.

  For example, if the URL that the master page used to open the detail page included the suffix locationDetail.aspx?recordID=CBR, then enter `recordID`.

The DataSet dialog box should look as follows:



6 Click OK.

The DataSet appears in the Bindings panel.

7 Bind the DataSet columns to the detail page by selecting the columns in the Bindings panel (Window > Bindings) and dragging them onto the page.

For more information, see "Making text dynamic" on page 547.

The ASP.NET master/detail page set is done.

After creating master/detail pages, you can use the Server Behaviors panel (Window > Server Behaviors) to modify the various building blocks.

Building a database search page (ASP.NET)

You can use Dreamweaver to build a page that searches a database and displays the results in a DataGrid.

This section describes the steps to build a database search page:

- "Adding the search controls (ASP.NET)" on page 681
- "Searching with only one search parameter (ASP.NET)" on page 682
- "Searching with multiple search parameters (ASP.NET)" on page 683
- "Displaying the results in a DataGrid" on page 684
- "Hiding the DataGrid the first time the page loads" on page 685
- "Creating a detail page (ASP.NET)" on page 685

Adding the search controls (ASP.NET)

A search page on the web typically contains form fields in which the user enters search parameters. When the user clicks a button, the search results are displayed.

To add the search form to the page:

1 Open the search page and select Insert > Form > Form.

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.

In the form's Property inspector, you should select `POST` as the form's Method attribute. You don't need to specify an Action attribute for the form. The ASP.NET page will post back to itself and the search results will be displayed on the same page.

2 Add form controls for users to enter their search parameters (Insert > ASP.NET Objects).

You can insert any ASP.NET form control, including a TextBox, CheckBox, RadioButton, ListBox, or DropDownList control. You can add as many controls 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 each control, make sure you specify an ID attribute such as **txtCity** for a TextBox control or **lbxCountry** for a ListBox control.

For more information, see "Adding ASP.NET form controls to a page" on page 623.

3 Add an ASP.NET Button to the form (Insert > ASP.NET Objects > asp:Button).

Make sure you specify an ID attribute for the button, such as *btnSearch*, and text for the button label, such as **Search**.

The search form is done. The next step in creating the search page is to define a DataSet to find and store the search results.

Searching with only one search parameter (ASP.NET)

When the user clicks the page's Search button, the search parameter is sent to the server. The server processes the request, builds a filtered DataSet based on the parameter, populates a DataGrid, and sends the page back to the browser.

Before adding the DataGrid, you must define a DataSet that finds and stores all the records that meet the search criteria.

To create a DataSet to hold the search results:

1. In the Bindings panel (Window > Bindings), click the Plus (+) button and select DataSet (Query).

   If the advanced dialog box appears, click the Simple button to open the simple dialog box.
2. Name the DataSet, and then select a connection and database table containing the data you want users to search.
3. In the Columns area, click the Selected option and select a key column (usually the record ID column) and the columns containing the data you want to display in the DataGrid.

Leave the DataSet dialog box open for now. You'll use it next to retrieve the search parameters sent to the server and create a DataSet filter to exclude records that don't meet the parameters.

To create a DataSet 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. You can use a less restrictive option such as "begins with" or "contains."
3. From the third pop-up menu, select Form Variable.

   The parameter is sent to the server by a form using the POST method.
4. In the fourth text box, enter the name of the form control that sent the search parameter to the server.

   You can get the name by clicking the form control on the form to select it, and checking the control's ID in the Property inspector.
5. If you want, click Test, enter a test value, and click OK to connect to the database and create an instance of the DataSet.

   The test value simulates the value that would otherwise have been returned from the search page. Click OK to close the Dataset.
6. If you're satisfied with the DataSet, click OK.

Dreamweaver inserts code in 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 DataSet. The code in effect builds a recordset containing only the search results.

The next step in creating the search page is to display the search results in a DataGrid. See "Displaying the results in a DataGrid" on page 684.

Searching with multiple search parameters (ASP.NET)

If the search page submits more than one search parameter to the server, then you must write a SQL query and use the search parameters in SQL variables.

Note: If you have only one search condition, you can use the simple DataSet dialog box to define your DataSet (see "Searching with only one search parameter (ASP.NET)" on page 682).

To search for records in a database using SQL:

1. Open the results page in Dreamweaver, and then create a new DataSet by opening the Bindings panel (Window > Bindings), clicking the Plus (+) button, and selecting DataSet from the pop-up menu.
2. Make sure the advanced 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 DataSet and select 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 question mark (?) placeholders for the search parameters. The following example contains two placeholders:

   ```
   SELECT EMPLOYEEID, FIRSTNAME, LASTNAME, DEPARTMENT, EXTENSION
   FROM EMPLOYEE WHERE LASTNAME LIKE ?
   AND DEPARTMENT LIKE ?
   ```

   For help on SQL syntax, see Appendix B, "SQL Primer," on page 753.
5. Give the placeholders the values of the search parameters by clicking the Plus (+) button in the Parameters area and entering the parameter's name, type, and value.

   The parameters must be listed in the same order they appear in the SQL statement.

   In the Name text box, enter any valid parameter name. The name cannot contain any spaces or special characters.

   In the Type pop-up menu, select a data type. For example, if the parameter will hold text, select WChar.

   In the Value box, enter the server variable that will contain the parameter value. For example, if the name of the form control on the search page is txtCity, then a server variable called Request.Form("txtCity") will be created and a value stored in it.

   You can also enter a more complete expression that specifies a default value in case the server variable doesn't exist. For example, if searching a Microsoft Access database, you can use % as a default value. The following expression checks to see if the server variable `Request.Form("txtCity")` exists. If the variable exists (that is, if it's not equal to nothing), the expression returns the variable's value; if it doesn't exist, the expression returns the default value of `%`.

   ```
   (IIf((Request.Form("txtCity") <> Nothing),
   Request.Form("txtCity"), "")) + "%"
   ```

   For more information, see a Visual Basic or C# language reference.

6 If you want to, click Test to create an instance of the DataSet 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 DataSet.

7 If you're satisfied with the DataSet, click OK.

Dreamweaver inserts the SQL query in your page.

The next step is to display the search results in a DataGrid.

Displaying the results in a DataGrid

After creating a DataSet to hold the search results, you can use a DataGrid to display the information on the page.

To display the search results in a DataGrid:

1 On the search page, place the insertion point where you want the DataGrid to appear.

2 In the Server Behaviors panel (Window > Server Behaviors), click the Plus (+) button and select DataGrid.

The DataGrid dialog box appears.

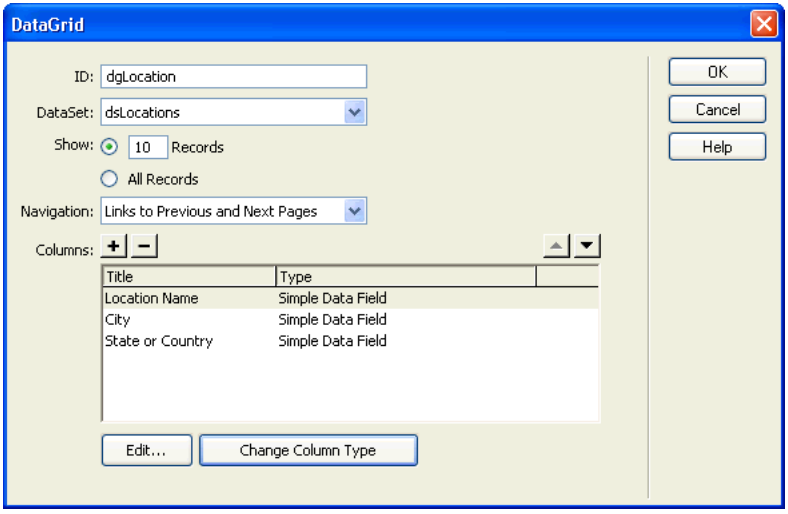
3 In the ID text box, enter a name for the DataGrid such as **dgLocation**.

4 In the DataSet pop-up menu, select the DataSet you defined to hold the search results.

5 Complete the rest of the dialog box as desired.

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

Here's an example of a DataGrid dialog box that will create a DataGrid with three columns and ten rows, and links to previous and next pages of results.



The DataGrid will appear the first time the search page loads even if the user hasn't conducted a search yet. You can hide the DataGrid when the page loads the first time.

Hiding the DataGrid the first time the page loads

When the search page first loads, you can hide the DataGrid that will be used to display the search results.

To hide the DataGrid the first time the page loads:

1 Open the search page in Code view (View > Code).

2 Immediately after the Register directive at the top of the page, enter the following code block if the page language is Visual Basic:

```
<script runat="server">
Sub Page_Load()
  If Not IsPostBack Then
    dgName.Visible = false
  Else
    dgName.Visible = true
  End If
End Sub
</script>
```

where *dgName* is the ID of your DataGrid.

If the page language is C#, enter the following code:

```
<script runat="server">
void Page_Load() {
  if (!IsPostBack) {
    dgName.Visible = false;
  } else {
    dgName.Visible = true;
  }
}
</script>
```

3 Save the page.

Creating a detail page (ASP.NET)

Your search page can include a detail page to display more information about specific records listed in the DataGrid. In this situation, the search page acts as the master page in a master/detail page set. For more information, see "Building master/detail pages (ASP.NET)" on page 675.

Building a record insert page (ASP.NET)

You can use Dreamweaver to build a page that lets users insert new records in a database. The method you use is identical for all server technologies supported by Dreamweaver. For instructions, see "Building a record insert page (all servers)" on page 646.

Building pages to update a record (ASP.NET)

Your application can contain a set of pages that lets users update existing records in a database table. The pages normally consist of a search page, a results page, and an update page. The search and results page let users retrieve the record and the update page lets users modify the record.

This section describes the steps to build pages to update a record:

- "Searching for the record to update (ASP.NET)" on page 686
- "Opening the update page and passing the record ID (ASP.NET)" on page 686
- "Retrieving the record to update (ASP.NET)" on page 689
- "Completing the update page in one operation (ASP.NET)" on page 690
- "Completing the update page block by block (ASP.NET)" on page 691

Related topics

- "Building a record insert page (all servers)" on page 646
- "Building pages to delete a record (ASP.NET)" on page 692

Searching for the record to update (ASP.NET)

When users want to update a record, they must first find that record in the database. Accordingly, you need a search page to work with the update page. The user selects the record to update from the DataGrid of results generated on the search page. When the user clicks the record, the update page opens and displays the record in a form.

For instructions on creating a page to search for the record to update, see "Building a database search page (ASP.NET)" on page 681.

After creating the search page, the next step is to create links on the results page that open the update page when clicked.

Opening the update page and passing the record ID (ASP.NET)

After creating the search page, you must create links that open the update page and pass the ID of the record that the user selected. The update page will use this ID to find the requested record in the database and display it.

The record ID is passed to the update page in a URL parameter. For more information, see "URL parameters" on page 518.

This section assumes you followed the instructions in "Building a database search page (ASP.NET)" on page 681 so that your search page uses a DataGrid to display records.

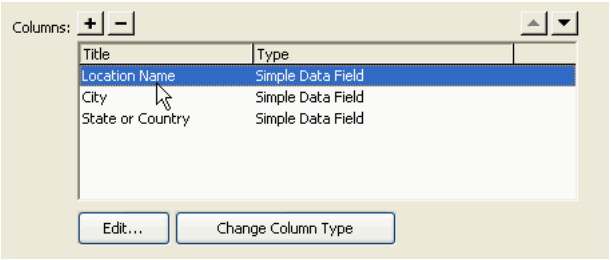
To create links on the search page to open the update page and pass the record ID:

1 Open the search page in Dreamweaver.

2 Double-click the DataGrid object listed in the Server Behaviors panel (Window > Server Behaviors).

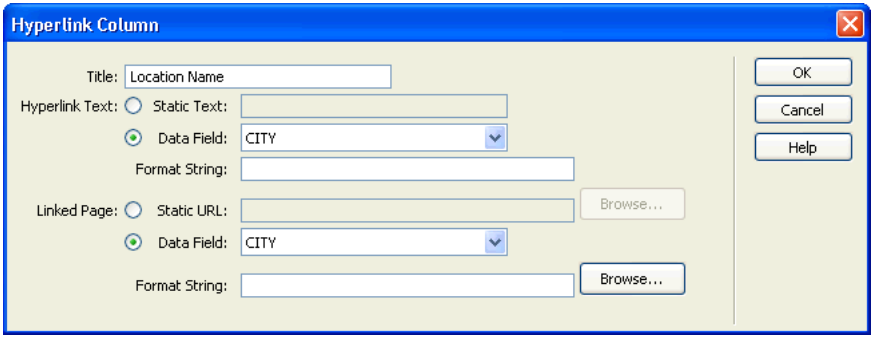
The DataGrid dialog box opens.

3 In the Columns box, select the column you want to apply links to.



4 Click the Change Column Type button and select Hyperlink from the pop-up menu.

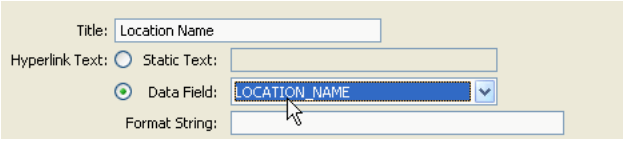
The Hyperlink Column dialog box appears.



5 In the Hyperlink Text area, specify the text to be displayed in the hyperlink column.

If you want to create a generic link such as Update for each row in the DataGrid, select the Static Text option and enter the text for the link. Each row in the DataGrid will display the same text (such as Update) in the hyperlink column.

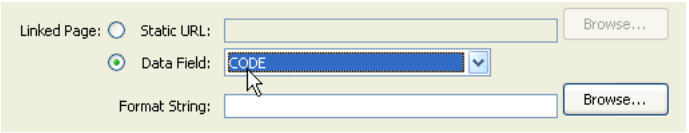
If you want to add links to data displayed in the column, select the Data Field option and select a data field in your DataSet. The data field provides the text for the links in your hyperlink column. In the following example, each row in the DataGrid will display a location name.



6 In the Linked Page area, build the URL to apply to the text in the hyperlink column.

The URL not only has to open the update page, it must uniquely identify the record to display on that page.

To identify the record to display on the update page, select the Data Field option and select a field in your DataSet that uniquely identifies each record. In most cases, the field will consist of a record ID number. In the following example, the field consists of unique location codes.



7 In the Format String text box in the Linked Page area, click the Browse button then locate and select your update page.

Dreamweaver creates a URL to the update page that includes a URL parameter identifying the record the update page should display. Make a note of the name of the URL parameter because you'll use it in the update page later.

For example, if you select locationUpdate.aspx as your update page, Dreamweaver creates the following URL and displays it in the Format String text box:

```
locationUpdate.aspx?CODE={0}
```

In this case, Dreamweaver creates a URL parameter called CODE. Dreamweaver copies the name of the data field, but you don't have to use that name. You can change it in the Format String text box to something more descriptive, such as recordID, as in the following example:

```
locationUpdate.aspx?recordID={0}
```

The {0} element is a placeholder corresponding to the data field's value. When the page runs, the values of the DataSet's CODE field are inserted in the corresponding rows in the DataGrid. For example, if the Canberra, Australia, rental location has the code CBR, then the following URL will be used in the Canberra row in the DataGrid:

```
locationUpdate.aspx?recordID=CBR
```

8 Click OK to close the Hyperlink dialog box, and then click OK to close the DataGrid dialog box.

Dreamweaver updates the DataGrid on your page.

After creating links that open the update page and pass the ID of the record that the user selected, the update page must retrieve the record to update.

Retrieving the record to update (ASP.NET)

After the search page passes a URL parameter to the update page, the update page must read the parameter, retrieve the record from the database table, and store it temporarily in a DataSet.

To retrieve the record to update:

1 Create a new ASP.NET page in Dreamweaver and save it.

The page will become your update page.

2 In the Bindings panel (Window > Bindings), click the Plus (+) button and select DataSet (Query).

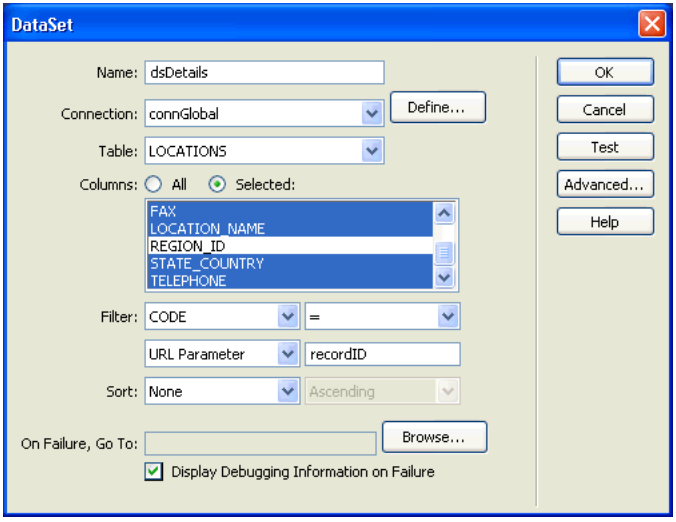
If the advanced dialog box appears, click the Simple button to open the simple dialog box.

3 Name the DataSet, and then select a connection and database table containing the data you want users to update.

4 Click the Selected option and select a key column (usually the record ID column) and the columns containing the data to be updated.

5 Configure the Filter area so that the value of your key column equals the value of the corresponding URL parameter passed by the search page.

This kind of filter creates a DataSet that contains only one record—the record with an ID that matches the value of the URL parameter. For example, if your key column contains record ID information and is called CODE, and if the search page passes the corresponding record ID information in the URL parameter called `recordID`, then here's how your Filter area should look:



6 Click OK.

When the user selects a record on the search page, the update page generates a DataSet containing only the selected record.

After modifying the update page to retrieve a record from the database and store it in a DataSet, you must add a form to let users modify the record data. You must also add the logic needed to update the database. You can accomplish these tasks in one operation or block by block. See "Completing the update page in one operation (ASP.NET)" on page 690 or "Completing the update page block by block (ASP.NET)" on page 691.

Completing the update page in one operation (ASP.NET)

An update page has three building blocks:

- A filtered DataSet to retrieve the record from a database table (see "Retrieving the record to update (ASP.NET)" on page 689)
- 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 final two building blocks of an update page in a single operation using the Record Update Form application object. The application object adds both an HTML form and an Update Record server behavior to the page.

Before you can use the application object, your web application must be able to identify the record to update, and your update page must be able to retrieve it. See "Searching for the record to update (ASP.NET)" on page 686, "Opening the update page and passing the record ID (ASP.NET)" on page 686, and "Retrieving the record to update (ASP.NET)" on page 689.

After the application 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.

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.

To build the update page with the Record Update Form application object:

1 Open the page in Design view, then select Insert > Application Objects > Update Record > Record Update Form Wizard.

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 application 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 (ASP.NET)

An update page has three building blocks:

- A filtered DataSet to retrieve the record from a database table (see "Retrieving the record to update (ASP.NET)" on page 689)
- 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 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 "Searching for the record to update (ASP.NET)" on page 686, "Opening the update page and passing the record ID (ASP.NET)" on page 686, and "Retrieving the record to update (ASP.NET)" on page 689.

Completing the update page consists of three tasks:

- Adding an HTML form to the page to let users modify the data
- Displaying the record in the form by binding the form objects to database table columns
- Adding the Update Record server behavior to update the database table after the user modifies the record

To add an HTML form to an update page:

1 Create a new ASP.NET page.

This will become your update page.

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 selecting 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 such as a text field (Insert > Form > Text Field) for each column you want to update in the database table.

The form objects let users modify the data. 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 DataSet 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 "Inserting HTML form objects" on page 613.

6 Add a Submit button to the form (Insert > Form > Button).

You can 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 text box.

To display the record in the form:

1 Make sure you defined a DataSet to hold the record the user wants to update.

For more information, see "Retrieving the record to update (ASP.NET)" on page 689.

2 Bind each form object to data in the recordset.

For more information, see the following sections:

- "Displaying dynamic content in HTML text fields" on page 618
- "Dynamically preselecting HTML checkboxes" on page 618
- "Dynamically preselecting HTML radio buttons" on page 619
- "Creating a dynamic HTML form menu" on page 617
- "Making existing HTML form menus dynamic" on page 618

To add a server behavior to update the database table:

1 In the Server Behaviors panel (Window > Server Behaviors), click the Plus (+) button and select 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 clicking the Submit button on the form.

To edit the server behavior, open the Server Behaviors panel (Window > Server Behaviors) and double-click the Update Record behavior.

Related topics

- "Completing the update page in one operation (ASP.NET)" on page 690

Building pages to delete a record (ASP.NET)

Your application can contain a set of pages that lets users delete records in a database. The pages normally consist of a search page and a delete page. The search page let users retrieve the record and the delete page lets users delete the record.

This section describes the steps to build ASP.NET pages that delete records:

- "Searching for the record to delete (ASP.NET)" on page 692
- "Creating delete links on the search page (ASP.NET)" on page 693
- "Displaying the record on the confirmation page (ASP.NET)" on page 695
- "Adding logic to delete the record (ASP.NET)" on page 698

Searching for the record to delete (ASP.NET)

When users want to delete a record, they must first find that record in the database. Accordingly, you need a search page to work with the delete page. The user enters search criteria in the search page and selects the record from the search results displayed in a DataGrid. When the user clicks the record, the delete page opens and displays the record in an HTML form.

For instructions on creating a page to search for the record to delete, see "Building a database search page (ASP.NET)" on page 681.

After creating the search page, the next step is to add delete buttons or hyperlinks to the DataGrid on the search page.

Creating delete links on the search page (ASP.NET)

After building the search page with a DataGrid, you must create links that open the delete confirmation page and pass the ID of the record that the user selected. The confirmation page will use this ID to find the requested record in the database and display it.

The record ID is passed to the confirmation page in a URL parameter. For more information, see "URL parameters" on page 518.

In a DataGrid, you create these links by adding a hyperlink column and setting its attributes, as described in this section.

To create delete links in the DataGrid on the search page:

1 Open the DataGrid dialog box you created in the previous section.

To open the dialog box, double-click your DataGrid in the Server Behaviors panel. Make sure you double-click the DataGrid listed in the panel, not the DataGrid item that appears in the pop-up menu when you click the Plus (+) button.

The DataGrid dialog box opens.

2 Add a column of delete links by clicking the Plus (+) button and selecting Hyperlink.

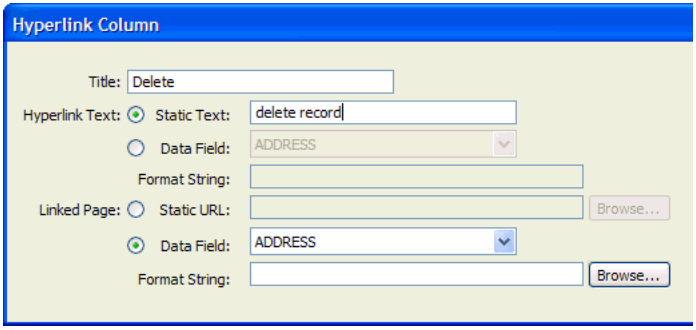
3 In the Title text box, enter a column a title such as Delete.

The title will appear in the column heading.

4 Select the Static Text option and enter the text of the link such as **delete record**.

Each row in the DataGrid will display the same text in the hyperlink column.

The DataGrid dialog box should look as follows:



5 In the Linked Page area, build the URL to apply to the text in the hyperlink column.

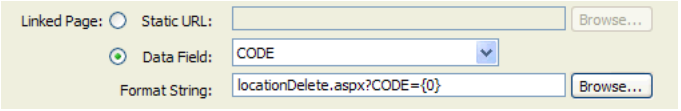
The URL not only has to open the confirmation page, it must uniquely identify the record to display on that page.

To identify the record to display on the confirmation page, select the Data Field option and select a field in your DataSet that uniquely identifies each record. In most cases, the field will consist of a record ID number.

6 In the Format String text box in the Linked Page area, click the Browse button then locate and select your confirmation page.

Dreamweaver creates a URL to the confirmation page that includes a URL parameter identifying the record the confirmation page should display. Make a note of the name of the URL parameter because you'll use it for the confirmation page later.

For example, if you select locationDelete.aspx as your detail page and you selected CODE as the field in your DataSet that uniquely identifies each record, then Dreamweaver creates the following URL.



In this case, Dreamweaver creates a URL parameter called `CODE`. Dreamweaver copies the name of the data field, but you don't have to use that name. You can change it to something more descriptive, such as recordID, as in the following example.

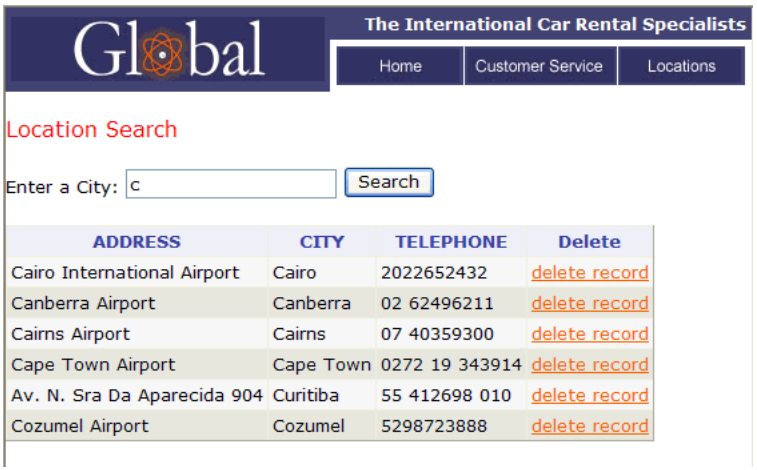
```
locationDelete.aspx?recordID={0}
```

The `{0}` element is a placeholder corresponding to the data field's value. When the page runs, the values of the DataSet's CODE field are inserted in the corresponding rows in the DataGrid. For example, if the Canberra, Australia, rental location has the code CBR, then the following URL will be used in the Canberra row in the DataGrid:

```
locationDelete.aspx?recordID=CBR
```

7 Click OK to close the Hyperlink dialog box; then click OK to close the DataGrid dialog box.

Dreamweaver updates the DataGrid on your page. The following is a DataGrid viewed in a browser after searching for all the cities that start with the letter *c*.



After creating the delete links, the next step is to display the record on the confirmation page.

Displaying the record on the confirmation page (ASP.NET)

After completing the page listing the records, switch to the delete confirmation page. The confirmation page shows the record and asks the user if they're sure they want to delete it. When the user confirms the operation by clicking the form button, the web application deletes the record from the database.

Building this page consists of the following tasks:

- Creating an HTML form with a button to click to confirm the deletion
- Retrieving the record to display on the page
- Displaying the record on the page

Retrieving and displaying the record consists of defining a DataSet to hold a single record—the record the user wants to delete—and binding the DataSet columns to the page. The steps are outlined in more detail below.

To create an HTML form to confirm the deletion:

1. Create a new ASP.NET page and save it as the confirmation page you specified in the previous section.

   You specified a confirmation page when you created the Delete links in the previous section. Use this name when saving the file for the first time (for example, locationDelete.aspx).
2. Insert an HTML form on the page (Insert > Form > Form).
3. Add a hidden form field to the form.

   The hidden form field is required to store the record ID passed by the URL parameter. To add a hidden field, place the insertion point in the form and select Insert > Form > Hidden Field.
4. Add a button to the form.

   The user will click the button to confirm and delete the displayed record. To add a button, place the insertion point in the form and select Insert > Form > Button.
5. Enhance the design of the page as desired and save it.

To retrieve the record the user wants to delete:

1. In the Bindings panel (Window > Bindings), click the Plus (+) button and select DataSet from the pop-up menu.

   The simple DataSet dialog box appears. If the advanced DataSet dialog box appears instead, click Simple to switch to the simple DataSet dialog box.
2. Name the DataSet, then select the database table that contains the records that could be displayed.
3. In the Columns area, select the table columns (record fields) you want to display on the page.

   To display only some of the record's fields, click Selected and select the desired fields by Control-clicking (Windows) or Command-clicking (Macintosh) them in the list.

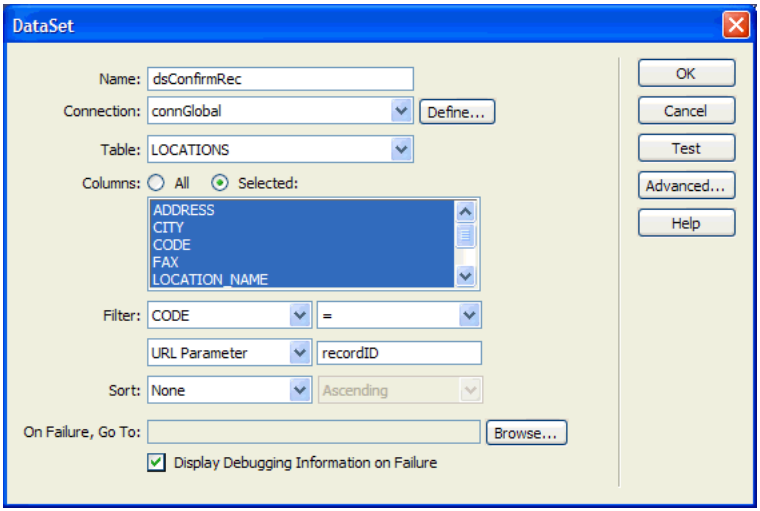
   Make sure to include the record ID field even if you won't be displaying it.

4 Complete the Filter section as follows to find and display the record specified in the URL parameter passed by the search page:

- From the first pop-up menu in the Filter area, select the column in the DataSet containing values that match the value of the URL parameter passed by the page with the Delete links. For example, if the URL parameter contains a record ID number, select the column containing record ID numbers. In the example discussed in the previous section, the DataSet column called CODE contains the values that match the value of the URL parameter passed by the page with the Delete links.
- From the pop-up menu beside the first menu, select the equal sign, if not already selected.
- From the third pop-up menu, select URL Parameter. The page with the Delete links uses a URL parameter to pass information to the confirmation page.
- In the fourth box, enter the name of the URL parameter passed by the page with the Delete links.

  For example, if the URL used to open the confirmation page included the suffix locationDelete.aspx?recordID=CBR, then enter recordID.

  The DataSet dialog box should look as follows.



5 Click OK.

The DataSet appears in the Bindings panel.

To display the record the user wants to delete:

1 Select the DataSet columns (record fields) in the Bindings panel and drag them to the confirmation page.

Make sure you insert this read-only dynamic content within the form boundaries. For more information on inserting dynamic content in pages, see "Making text dynamic" on page 547.

Next, you must bind the record ID column to the hidden form field.

2 Make sure Invisible Elements are turned on (View > Visual Aids > Invisible Elements), then click the yellow shield icon representing the hidden form field.

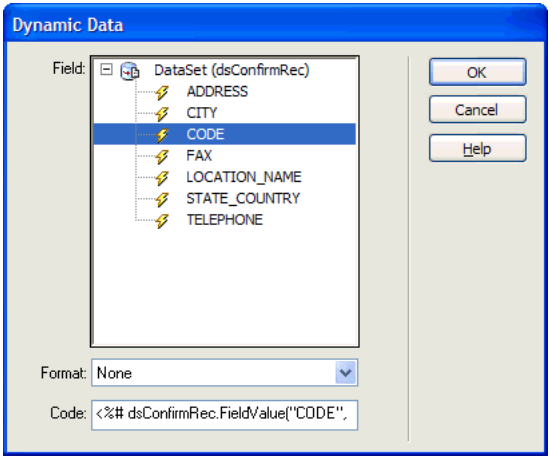
The hidden form field is selected.

3 In the Property inspector, click the lightning bolt icon beside the Value text box.

The Dynamic Data dialog box appears.

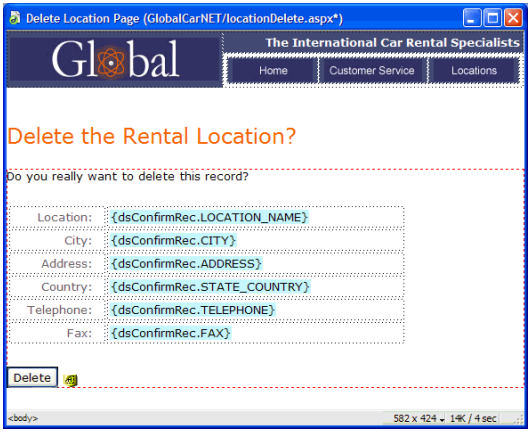
4 In the Dynamic Data dialog box, select the record ID column in the DataSet and click OK.

In the following example, the record ID column selected is CODE.



5 Save the page.

The completed confirmation page should look as follows.



After creating a confirmation page, the next step is to add logic to delete the record.

Adding logic to delete the record (ASP.NET)

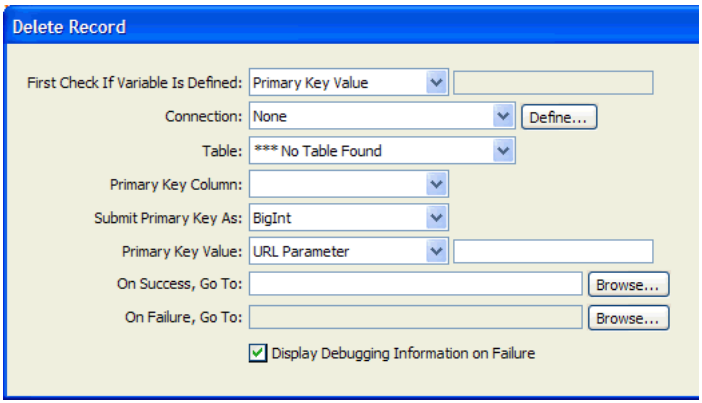
After creating the confirmation page, the next step is to add logic to the page that deletes the record from the database when the user clicks the Confirm button. You can add this logic quickly and easily in Dreamweaver with the Delete Record server behavior.

To add logic to delete the record displayed in the HTML form:

1 Make sure the confirmation page is open in Dreamweaver.

2 In the Server Behaviors panel (Window > Server Behaviors), click the Plus (+) button and select Delete Record.

The Delete Record dialog box appears.



3 In the First Check If Variable Is Defined text box, make sure Primary Key Value is selected.

You specify the primary key value later in the dialog box.

4 In the Connection pop-up menu, select a connection to the database so that the page can find and connect to the database.

5 In the Table pop-up menu, select the database table that contains the records that will be deleted.

6 In the Primary Key Column pop-up menu, select the table column that contains record IDs.

The Delete Record server behavior will search this column for a match. The column should contain the same record ID data as the DataSet column you bound to the hidden form field on the page.

7 In the Submit Primary Key As pop-up menu, select the data type of your primary key column.

8 In the Primary Key Value pop-up menu, select the variable on your page that contains the record ID identifying the record to be deleted.

The variable is created by your hidden form field. It has the same name as the name attribute of the hidden field and is either a form or URL parameter, depending on the form's method attribute. In our example, the variable is a URL parameter called hiddenID.

9 In the On Success, Go To text box, specify a page to open after the record has been deleted from the database table.

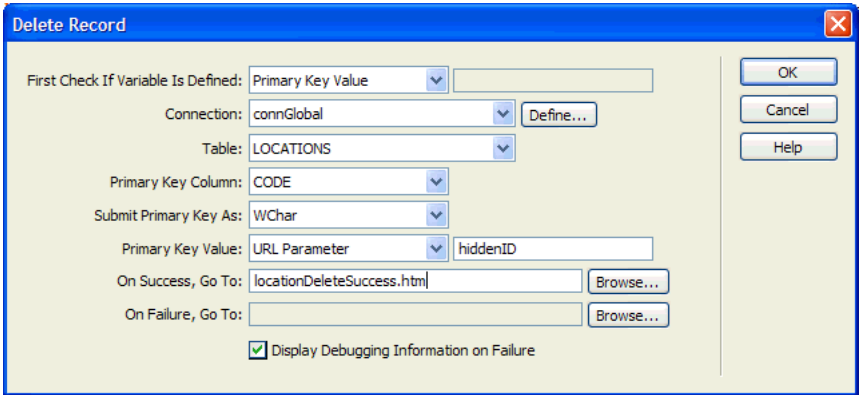
You can specify a page that contains a brief success message to the user, or a page listing the remaining records so that the user can verify that the record has been deleted.

10 If you want, select the Display Debugging Information on Failure option.

The debugging information will be generated by the server.

11 If you want to provide a custom error message, deselect the Display Debugging Information on Failure option and specify a page to open in the On Failure, Go To text box.

The completed Delete Record dialog box should look as follows.



12 Click OK and save your work.

Upload the pages to your web server (if required), open a browser and search for a record to delete. When you click a Delete link on the results page, the confirmation page appears. Click the form button to delete the record from the database. To verify that the record has been deleted, search for the record again.

Using stored procedures to modify databases (ASP.NET)

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.

Note: Microsoft Access and MySQL databases do not support stored procedures.

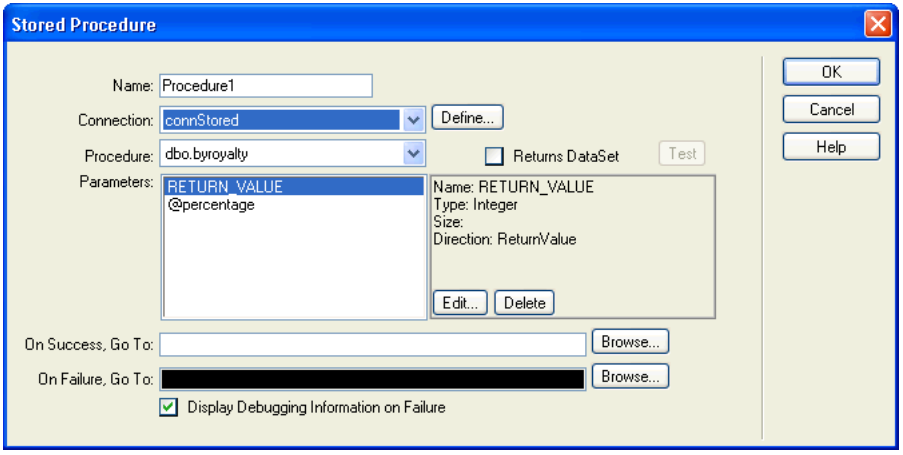
Before you use a stored procedure to modify a database, make sure the stored procedure contains SQL that modifies the database in some way. To create and store one in your database, consult your database documentation and a good Transact-SQL manual.

To add a stored procedure to an ASP.NET page:

1 In Dreamweaver, open the page that will run the stored procedure.

2 In the Bindings panel (Window > Bindings), click the Plus (+) button and 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.

Related topics

- "Stored procedures" on page 632

Building pages that restrict access to your site (ASP.NET)

You can use Dreamweaver to build a set of pages that restrict access to your site. Dreamweaver does not have authentication server behaviors for ASP.NET pages. However, since ASP and ASP.NET pages can run on the same site, you can use the ASP authentication server behaviors. The method you use to build these pages is identical to the one for ColdFusion. For instructions, see "Building pages that restrict access to your site (ColdFusion, ASP, JSP, PHP)" on page 666.

CHAPTER 42
Building ASP and JSP Applications Rapidly

You can use the tools in Macromedia Dreamweaver MX 2004 to build an ASP or JSP web application rapidly and with little or no coding.

Related topics

- "About rapid application development (all servers)" on page 629

Building master/detail pages (ASP and JSP)

With Dreamweaver, you can create sets of pages that present information in two levels of detail: a master page that lists records and a detail page that displays more detail about each record. This section describes how to build master/detail pages.

Building master/detail pages in one operation (ASP and JSP)

When developing ASP or JSP applications, you can quickly build master/detail pages using the Master/Detail Page Set application object. An application object lets you build a complete set of dynamic pages by completing only one or two dialog boxes.

The method you use is identical to the one for ColdFusion. For instructions, see "Building master/detail pages in one operation (ColdFusion, ASP, JSP, PHP)" on page 634.

You can also build master/detail pages using individual server behaviors.

Building master/detail pages block by block (ASP and JSP)

This section describes how to build a set of master/detail pages without using the Master/Detail Page Set application object. For instructions on using the application object, see "Building master/detail pages in one operation (ASP and JSP)" on page 703.

This section describes the steps to build a set of master/detail pages:

- "Creating a master page and defining a recordset for it (ASP and JSP)" on page 704
- "Displaying the records on the master page (ASP and JSP)" on page 704
- "Opening the detail page and passing a record ID to it (ASP and JSP)" on page 704
- "Finding and displaying the requested record on the detail page (ASP and JSP)" on page 705

Creating a master page and defining a recordset for it (ASP and JSP)

The first step in building master/detail pages is to create a blank master page and add a recordset to it.

You can define a recordset at design time (see "Defining a recordset" on page 530) or the user can define one at runtime (see "Building search/results pages (ASP and JSP)" on page 706).

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.

The next step in creating master/detail pages is to display the records on the master page.

Displaying the records on the master page (ASP and JSP)

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. For more information, see "Displaying multiple recordset results" on page 566.

The next step in creating master/detail pages is to open the detail page and pass a record ID to it.

Opening the detail page and passing a record ID to it (ASP and JSP)

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 the detail page.

To open the detail page and pass it a URL parameter identifying the record to display:

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 select 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, see "URL parameters" on page 518.

The next step in creating master/detail pages is to find and display the requested record on the detail page.

Finding and displaying the requested record on the detail page (ASP and JSP)

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, or create a new ASP or JSP page if it doesn't exist yet.
2. In the Bindings panel (Window > Bindings), click the Plus (+) button and select Recordset from the pop-up menu.

   The simple Recordset dialog box appears. If the advanced dialog box appears instead, click Simple to switch.
3. Name the recordset, then select 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 select 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, select 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.

The master/detail page set is done.

After creating master/detail page, you can use the Server Behaviors panel (Window > Server Behaviors) to modify the various server behaviors.

Building search/results pages (ASP and JSP)

You can use Dreamweaver to build a set of pages to let users search your database. The method you use is identical to the one for ColdFusion. For instructions, see "Building search/results pages (ColdFusion, ASP, JSP, PHP)" on page 641.

Building a record insert page (ASP and JSP)

You can use Dreamweaver to build a page that lets users insert new records in a database. The method you use is identical for all server technologies supported by Dreamweaver. For instructions, see "Building a record insert page (all servers)" on page 646.

Building pages to update a record (ASP and JSP)

Your application can contain a set of pages that lets users update existing records in a database table. The pages normally consist of a search page, a results page, and an update page. The search and results page let users retrieve the record and the update page lets users modify the record.

This section describes the steps to build a page to update records:

- "Searching for the record to update (ASP and JSP)" on page 707
- "Opening the update page and passing the record ID (ASP and JSP)" on page 707
- "Retrieving the record to update (ASP and JSP)" on page 708
- "Completing the update page in one operation (ASP and JSP)" on page 709
- "Completing the update page block by block (ASP and JSP)" on page 709

Related topics

- "Building a record insert page (all servers)" on page 646
- "Building pages to delete a record (ASP and JSP)" on page 711

Searching for the record to update (ASP and JSP)

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. The user enters search criteria in the search page and selects the record on the results page. When the user clicks the record, the update page opens and displays the record in an HTML form.

For instructions on creating pages to search for the record to update, see "Building search/results pages (ColdFusion, ASP, JSP, PHP)" on page 641.

After creating the search/results pages, the next step in creating a record update page is to create links on the results page that open the update page when clicked.

Opening the update page and passing the record ID (ASP and JSP)

After creating the search/results pages, you must create links on the results page that open the update page and pass the ID of the record that the user selected. The update page will use this ID to find the requested record in the database and display it.

The record ID is passed to the update page in a URL parameter. For more information, see "URL parameters" on page 518.

To create links to open the update page and pass the record ID:

1. In the results page, select the dynamic content placeholder for the text you want to be linked.
2. In the Server Behaviors panel (Window > Server Behaviors), click the Plus (+) button and select Go To Detail Page from the pop-up menu.

   The Go To Detail Page dialog box appears.
3. In the Detail Page text box, click Browse and locate the update page.
4. Name the URL parameter and specify the recordset and column containing the record ID you want to pass to the update page.

   Make a note of the name of the URL parameter because you'll use it in the update page later.
5. Click OK.

Dreamweaver inserts a special link in the page. When the user clicks the link, a URL parameter containing the record ID is passed to the update page.

The next step in creating a record update page is to retrieve the record to update. .

Retrieving the record to update (ASP and JSP)

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 Create a new ASP or JSP page in Dreamweaver and save it.

The page will become your update page.

2 In the Bindings panel (Window > Bindings), click the Plus (+) button and select Recordset.

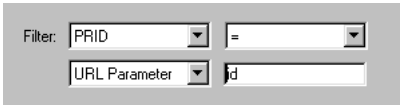
If the advanced dialog box appears, click the Simple button to open the simple dialog box.

3 Name the recordset and specify where the data you want to update is located using the Connection and Table pop-up menus.

4 Click the Selected option and select a key column (usually the record ID column) and the columns containing the data to be updated.

5 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:



6 Click OK.

When the user selects a record on the results page, the update page will generate a recordset containing only the selected record.

After modifying the update page to retrieve a record from the database and store it in a recordset, you must add a form to let users modify the record data, and the logic needed to update the database. You can accomplish these tasks in one operation or block by block. See "Completing the update page in one operation (ASP and JSP)" on page 709 or "Completing the update page block by block (ASP and JSP)" on page 709.

Completing the update page in one operation (ASP and JSP)

An update page has three building blocks:

- A filtered recordset to retrieve the record from a database table (see "Retrieving the record to update (ASP and JSP)" on page 708)
- 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 final two building blocks of an update page in a single operation using the Record Update Form application object. The application object adds both an HTML form and an Update Record server behavior to the page.

Before you can use the application object, your web application must be able to identify the record to update, and your update page must be able to retrieve it. See "Searching for the record to update (ASP and JSP)" on page 707, "Opening the update page and passing the record ID (ASP and JSP)" on page 707, and "Retrieving the record to update (ASP and JSP)" on page 708.

After the application 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.

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.

To build the update page with the Record Update Form application object:

1 Open the update page in Design view, then select Insert > Application Objects > Update Record > Record Update Form Wizard.

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 application 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 (ASP and JSP)

An update page has three building blocks:

- A filtered recordset to retrieve the record from a database table (see "Retrieving the record to update (ASP and JSP)" on page 708)
- 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 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 "Searching for the record to update (ASP and JSP)" on page 707, "Opening the update page and passing the record ID (ASP and JSP)" on page 707, and "Retrieving the record to update (ASP and JSP)" on page 708.

Completing the update page consists of three tasks:

- Adding an HTML form to the page to let users modify the data
- Displaying the record in the form by binding the form objects to database table columns
- Adding the Update Record server behavior to update the database table after the user modifies the record and clicks the form button

To add an HTML form to an update page:

1 Create a new ASP or JSP page (File > New).

The page will become your update page.

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 selecting Insert > Form > Form.

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 such as a text field (Insert > Form > Text Field) 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 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, see "Inserting HTML form objects" on page 613.

6 Add a Submit button to the form (Insert > Form > Button).

You can 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.

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 (ASP and JSP)" on page 708.

2 Bind each form object to data in the recordset, as described in the following sections:

- "Displaying dynamic content in HTML text fields" on page 618
- "Dynamically preselecting HTML checkboxes" on page 618
- "Dynamically preselecting HTML radio buttons" on page 619
- "Creating a dynamic HTML form menu" on page 617
- "Making existing HTML form menus dynamic" on page 618

To add a server behavior to update the database table:

1 In the Server Behaviors panel (Window > Server Behaviors), click the Plus (+) button and select 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 clicking the Submit button on the form.

To edit the server behavior, open the Server Behaviors panel (Window > Server Behaviors) and double-click the Update Record behavior.

Related topics

- "Completing the update page in one operation (ASP and JSP)" on page 709

Building pages to delete a record (ASP and JSP)

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 select 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 (see "Retrieving the record to delete (ASP and JSP)" on page 712)
- A read-only display of the data about to be deleted (see "Displaying the data to be deleted (ASP and JSP)" on page 713)
- A Submit button to send the delete command to the server (see "Sending the delete command to the server (ASP and JSP)" on page 713)
- A Delete Record server behavior to update the database table (see "Deleting the record from the database table (ASP and JSP)" on page 714)

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.

Related topics

- "Building a record insert page (ASP and JSP)" on page 706
- "Building pages to update a record (ASP and JSP)" on page 707

Identifying the record to delete (ASP and JSP)

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 search/results pages (ASP and JSP)" on page 706.

The results page tells the delete page which record to delete by passing it a URL parameter. The next step is to retrieve the record to delete.

Retrieving the record to delete (ASP and JSP)

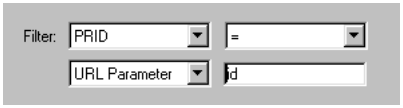
After the results page passes a URL parameter to the delete page identifying the record to delete, the delete page must read the parameter, retrieve the record from the database table, and temporarily store the record in a recordset.

To retrieve the record to delete:

1. In the Bindings panel (Window > Bindings), click the Plus (+) button and select Recordset.

   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:



5. Click OK.

When the user selects a record on the results page, the delete page generates a recordset containing only the selected record.

The next step in building a page to delete records is to display the record to be deleted.

Displaying the data to be deleted (ASP and JSP)

After retrieving the record to delete, it's 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 (ASP and JSP)" on page 712.
2. Drag a recordset 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 547.

The next step in building a page to delete records is to send the delete command to the server.

Sending the delete command to the server (ASP and JSP)

After retrieving and displaying the record to delete, you must add a button the user can click to delete the record. The delete page uses a form 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 select Insert > Form > Form.

   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 > 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.

The next step is to add the Delete Record server behavior to update the database table after the user clicks the Submit button.

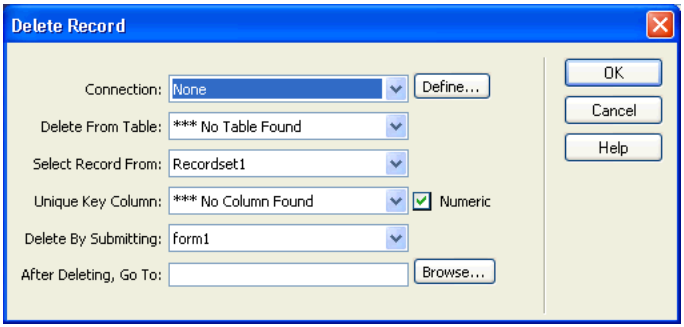
Deleting the record from the database table (ASP and JSP)

After adding a button the user can click to delete the record , 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 select 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.

Building pages with advanced data manipulation objects (ASP and JSP)

You can use Dreamweaver to build pages that use stored procedures, ASP command objects, and JSP prepared statements.

Using stored procedures to modify databases (ASP and JSP)

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.

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.

Related topics

- "Stored procedures" on page 632

Running a stored procedure (ASP)

With ASP pages, you must add a command object to a page to run a stored procedure. For more information on command objects, see "ASP command objects" on page 633.

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 (Window > Bindings), click the Plus (+) button, and then select Command (Stored Procedure).

   The Command dialog box appears.
3. Enter a name for the command, select a connection to the database containing the stored procedure, and then select Stored Procedure from the Type pop-up menu.
4. Select your stored procedure by expanding the Stored Procedures branch in the Database Items box, selecting 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 a stored procedure (JSP)

With JSP pages, 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 (Window > Bindings), click the Plus (+) button and select Callable (Stored Procedure).

   The Callable (Stored Procedure) dialog box appears.
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. A command object is a server object that performs some operation on a database. You supply the command object with the SQL statement that performs the operation on the database. For more information, see "ASP command objects" on page 633.

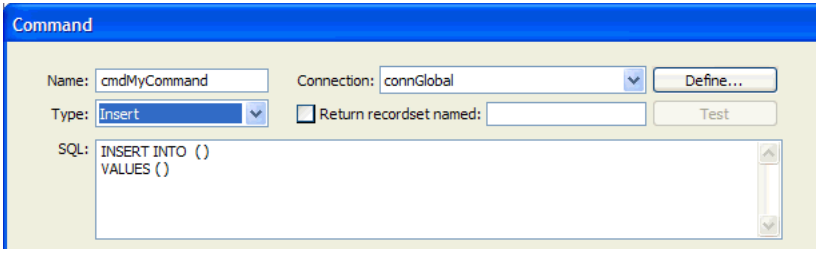
You can also supply the object with a stored procedure that performs the operation. For more information, see "Running a stored procedure (ASP)" on page 715.

To create the command object that uses SQL to edit a database:

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 select Command.

   The Command dialog box appears.
3. Enter a name for the command, select a connection to the database containing the records you want to edit, and select 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 select. For example, if you select Insert, the dialog looks as follows:

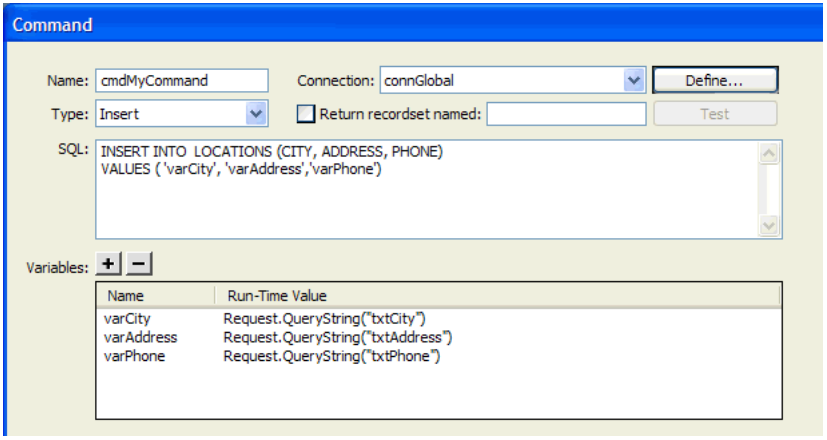


4 Complete the SQL statement.

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.



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 (txtCity, txtAddress, and txtPhone) and a submit button. 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. A JSP prepared statement is a reusable server object that contains a SQL statement. You supply the prepared statement with the SQL that performs the operation on the database. For more information, see "JSP prepared statements" on page 633.

To create a prepared statement that edits a database record:

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 select Prepared (Insert, Update, Delete).

   The Prepared (Insert, Update, Delete) dialog box appears.
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.

Building pages that restrict access to your site (ASP and JSP)

You can use Dreamweaver to build a set of pages that restrict access to your site. The method you use is identical to the one for ColdFusion. For instructions, see "Building pages that restrict access to your site (ColdFusion, ASP, JSP, PHP)" on page 666.

CHAPTER 43
Building PHP Applications Rapidly

You can use the tools in Macromedia Dreamweaver MX 2004 to build a PHP web application rapidly and with little or no coding.

Related topics

- "About rapid application development (all servers)" on page 629

Building master/detail pages (PHP)

With Dreamweaver, you can create sets of pages that present information in two levels of detail: a master page that lists records and a detail page that displays more detail about each record. This section describes how to build these kinds of master/detail pages.

Building master/detail pages in one operation (PHP)

When developing PHP applications, you can quickly build master/detail pages using the Master/Detail Page Set application object. An application object lets you build a complete set of dynamic pages by completing only one or two dialog boxes.

The method you use is identical to the one for ColdFusion. For instructions, see "Building master/detail pages in one operation (ColdFusion, ASP, JSP, PHP)" on page 634.

You can also build master/detail pages using individual server behaviors.

Building master/detail pages block by block (PHP)

You can add the basic building blocks of master/detail pages separately using the Server Behaviors panel.

You can also add the building blocks all at once using the Master/Detail Page Set application object. For more information, see "Building master/detail pages in one operation (PHP)" on page 719.

This section describes the steps to build a set of master/detail pages:

- "Creating the master page (PHP)" on page 720
- "Creating the links to the detail page (PHP)" on page 722
- "Creating a URL parameter for the links (PHP)" on page 723
- "Finding and displaying the requested record on the detail page (PHP)" on page 724

Creating the master page (PHP)

This section describes how to create a master page that lists database records. You can use a dynamic table to list the records on a PHP page.

Before you start, make sure you define a database connection for the site. For more information, see Chapter 28, "Database Connections for PHP Developers," on page 487.

To create a master page:

1 In Dreamweaver, create a new PHP page.

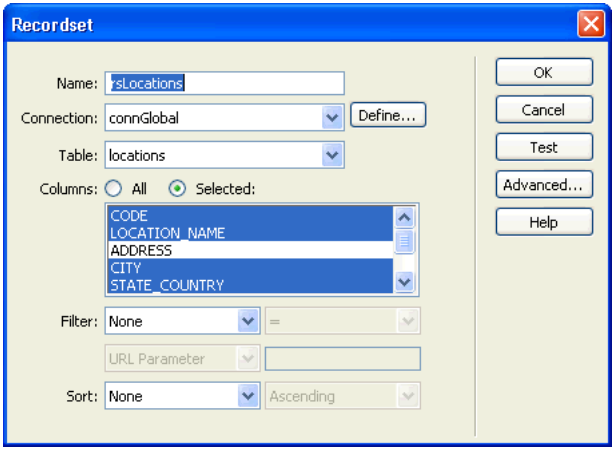
Select File > New > Dynamic, select PHP, and click Create.

A blank PHP page opens in Dreamweaver.

2 Define a recordset for the page.

In the Bindings panel (Window > Bindings), click the Plus (+) button, select Recordset (Query), and complete the Recordset dialog box. For more information, click the Help button on the dialog box. If you want to write your own SQL statement, click the Advanced button to open the advanced Recordset dialog box.

Make sure the recordset contains all the table columns you need to create your dynamic table. The recordset must also include the table column containing the unique key of each record—that is, the record ID column. In the following example, the CODE column contains information that uniquely identifies each record.



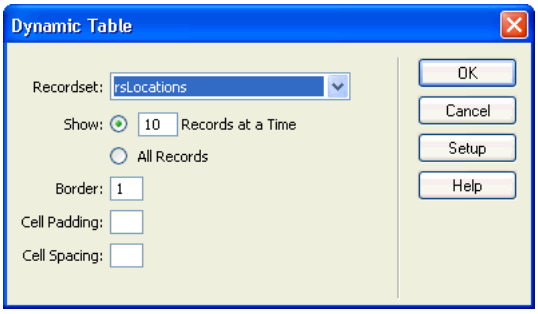
Typically, the recordset on the master page extracts a few columns from a database table while the recordset on the detail page extracts more columns from the same table to provide the extra detail.

The recordset can be defined by the user at runtime. For more information, see "Building search/results pages (PHP)" on page 725.

3 Insert a dynamic table to display the records on the page.

Place the insertion point where you want the dynamic table to appear on the page, and select Insert > Application Objects > Dynamic Data > Dynamic Table.

The Dynamic Table dialog box appears.



4 Complete the Dynamic Table dialog box and click OK.

Dreamweaver inserts a dynamic table on the page.

If you need help completing the Dynamic Table dialog box, click the Help button on the dialog box.

5 If you want, delete the dynamic table column containing the record IDs.

If you don't want to show record IDs to users, you can delete the column from the dynamic table. Click anywhere on the page to move the focus to the page. Move the pointer near the top of the column in the dynamic table until the column cells are outlined in red, then click to select the table column. Press Delete to delete the column from the table.

The next step in creating master/detail pages is to create links that open the detail page.

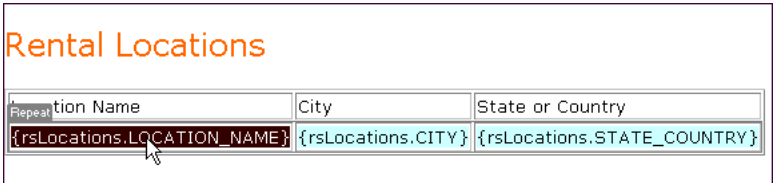
Creating the links to the detail page (PHP)

After creating the dynamic table, you must create links that open the detail page. This section describes how to create the links. The next section describes how to modify the link so that it also passes the ID of the record the user selects. The detail page will use this ID to find the requested record in the database and display it.

To create links to the detail page:

1 In the dynamic table, select the dynamic content placeholder for the text you want to be linked.

In the following example, the {rsLocations.LOCATION_NAME} placeholder is selected. The links will be applied to the location names in the column.



2 In the Property inspector, click the folder icon beside the Link field.

3 Browse and select the detail page.

The detail page appears in the Link field.

In the dynamic table, the placeholder for the dynamic content appears linked. If you run the page on a server, you can see that the link is applied to the text in every table row.

The next step in creating master/detail pages is to create a URL parameter to pass the ID of the record the user selected.

Creating a URL parameter for the links (PHP)

The links in the dynamic table on the master page not only have to open the detail page, they must pass the ID of the record the user selected. The detail page will use this ID to find the requested record in the database and display it.

The record ID is passed to the detail page in a URL parameter. For more information, see "URL parameters" on page 518.

This section describes how to create a URL parameter that passes a record ID to the detail page.

To create the URL parameter:

1 Select the link in the dynamic table.

If Live Data is turned on, select the link in the first row.

2 In the Link field in the Property inspector, add the following string at the end of the URL:

```
?recordID=<?php echo $row_recordsetName['fieldName']; ?>
```

The question mark tells the server that what follows is one or more URL parameters. The word `recordID` is the name of the URL parameter (you can use any name you like). Make a note of the name of the URL parameter because you'll use it in the detail page later.

The expression after the equal sign is the value of the parameter. In this case, the value is generated by a PHP expression that returns a record ID from the recordset. A different ID is generated for each row in the dynamic table. In the PHP expression, replace *recordsetName* with the name of your recordset, and replace *fieldName* with the name of the field in your recordset that uniquely identifies each record. In most cases, the field will consist of a record ID number. In the following example, the field consists of unique location codes.

```
locationDetail.php?recordID=<?php echo $row_rsLocations['CODE']; ?>
```

When the page runs, the values of the recordset's CODE field are inserted in the corresponding rows in the dynamic table. For example, if the Canberra, Australia, rental location has the code CBR, then the following URL will be used in the Canberra row in the dynamic table:

```
locationDetail.php?recordID=CBR
```

3 Save the page.

The next step in creating master/detail pages is to modify the detail page so that it can find the requested record in the database and display it on the page.

Finding and displaying the requested record on the detail page (PHP)

After completing the master page, switch to the detail page. You must find the requested record in the database and display it on the 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 or create a new PHP page if the page doesn't exist.
2. In the Bindings panel (Window > Bindings), click the Plus (+) button and select Recordset (Query) from the pop-up menu.

   The simple Recordset dialog box appears. If the advanced Recordset dialog box appears instead, click Simple to switch to the simple Recordset dialog box.
3. Name the recordset, and then select a connection and the database table that will provide data to your recordset.
4. In the Columns 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.

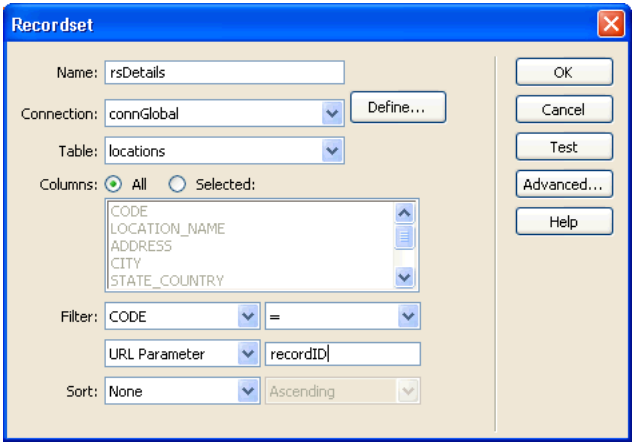
   If the recordsets are different, make sure the recordset on the detail page contains at least one column in common with the recordset on the master page. The common column is usually the record ID column, but it can also be the join field of related tables.

   To include only some of the table's columns in the recordset, click Selected and select 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 master page:
   - From the first pop-up menu in the Filter area, select the column in the recordset 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, select the column containing record ID numbers. In the example discussed in the previous section, the recordset column called CODE contains the values that match the value of the URL parameter passed by the master page.
   - Select the equal sign (if it is not already selected) from the pop-up menu beside the first menu.
   - Select URL Parameter from the third pop-up menu.

     The master page uses a URL parameter to pass information to the detail page.
   - In the fourth box, enter the name of the URL parameter passed by the master page.

     For example, if the URL that the master page used to open the detail page included the suffix locationDetail.php?recordID=CBR, then enter `recordID`.

The Recordset dialog box should look as follows:



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.

For more information, see "Making text dynamic" on page 547.

The PHP master/detail page set is done.

After creating master/detail pages, you can use the Server Behaviors panel (Window > Server Behaviors) to modify the various building blocks.

Building search/results pages (PHP)

You can use Dreamweaver to build a set of pages to let users search your database. The method you use is identical to the one for ColdFusion. For instructions, see "Building search/results pages (ColdFusion, ASP, JSP, PHP)" on page 641.

Building a record insert page (PHP)

You can use Dreamweaver to build a page that lets users insert new records in a database. The method you use is identical for all server technologies supported by Dreamweaver. For instructions, see "Building a record insert page (all servers)" on page 646.

Building pages to update a record (PHP)

Your application can contain a set of pages that lets users update existing records in a database table. The pages normally consist of a search page, a results page, and an update page. The search and results page let users retrieve the record and the update page lets users modify the record.

This section describes the steps to build pages to update a record:

- "Searching for the record to update (PHP)" on page 726
- "Creating the links to open the update page (PHP)" on page 727
- "Creating a URL parameter for the links (PHP)" on page 723
- "Retrieving the record to update (PHP)" on page 728
- "Completing the update page in one operation (PHP)" on page 729
- "Completing the update page block by block (PHP)" on page 730

Related topics

- "Building a record insert page (all servers)" on page 646
- "Building pages to delete a record (PHP)" on page 732

Searching for the record to update (PHP)

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. The user enters search criteria in the search page and selects the record on the results page. When the user clicks the record, the update page opens and displays the record in an HTML form.

For instructions on creating pages to search for the record to update, see "Building search/results pages (PHP)" on page 725.

After creating the search/results pages, the next step is to create links on the results page that open the update page when clicked.

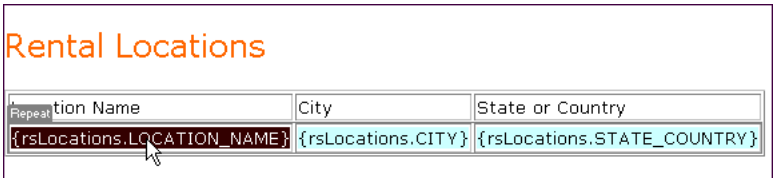
Creating the links to open the update page (PHP)

After creating the search/results pages, you must create links on the results page to open the update page and display the selected record in an HTML form. This section describes how to create the links. The next section describes how to modify the link so that it also passes the ID of the record the user selects. The update page will use this ID to find the requested record in the database and display it.

To create links to the update page:

1 In the results page, select the dynamic content placeholder for the text you want to be linked.

In the following example, the {rsLocations.LOCATION_NAME} placeholder is selected. The links will be applied to the location names in the column.



2 In the Property inspector, click the folder icon beside the Link field.

3 Browse and select the update page.

The update page appears in the Link field.

On the results page, the placeholder for the dynamic content appears linked. If you upload the pages to the server and run a search, you can see that the link is applied to every record on the results page.

The next step in creating a record update page is to create a URL parameter to pass the ID of the record the user selected.

Creating a URL parameter for update links (PHP)

The links on the results page not only have to open the update page, they must pass the ID of the record the user selected. The update page will use this ID to find the requested record in the database and display it.

The record ID is passed to the update page in a URL parameter. For more information, see "URL parameters" on page 518.

This section describes how to create a URL parameter that passes a record ID to the update page.

To create the URL parameter:

1 Select the link on the results page.

2 In the Link field in the Property inspector, add the following string at the end of the URL:

```
?recordID=<?php echo $row_recordsetName['fieldName']; ?>
```

The question mark tells the server that what follows is one or more URL parameters. The word `recordID` is the name of the URL parameter (you can make up any name you like). Make a note of the name of the URL parameter because you'll use it in the update page later.

The expression after the equal sign is the value of the parameter. In this case, the value is generated by a PHP expression that returns a record ID from the recordset. A different ID is generated for each row in the dynamic table. In the PHP expression, replace `recordsetName` with the name of your recordset, and replace `fieldName` with the name of the field in your recordset that uniquely identifies each record. In most cases, the field will consist of a record ID number. In the following example, the field consists of unique location codes.

```
locationDetail.php?recordID=<?php echo $row_rsLocations['CODE']; ?>
```

When the page runs, the values of the recordset's CODE field are inserted in the corresponding rows in the dynamic table. For example, if the Canberra, Australia, rental location has the code CBR, then the following URL will be used in the Canberra row in the dynamic table:

```
locationDetail.php?recordID=CBR
```

3 Save the page.

The next step is to modify the update page so that it can find the requested record in the database and display it on the page.

Retrieving the record to update (PHP)

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 Create a new PHP page in Dreamweaver and save it.

The page will become your update page.

2 In the Bindings panel (Window > Bindings), click the Plus (+) button and select Recordset.

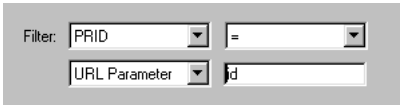
If the advanced dialog box appears, click the Simple button to open the simple dialog box.

3 Name the recordset and specify where the data you want to update is located using the Connection and Table pop-up menus.

4 Click the Selected option and select a key column (usually the record ID column) and the columns containing the data to be updated.

5 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:



6 Click OK.

When the user selects a record on the results page, the update page generates a recordset containing only the selected record.

After modifying the update page to retrieve a record from the database and store it in a recordset, you must add a form to let users modify the record data, and the logic needed to update the database. You can accomplish these tasks in one operation or block by block. See "Completing the update page in one operation (PHP)" on page 729 or "Completing the update page block by block (PHP)" on page 730.

Completing the update page in one operation (PHP)

An update page has three building blocks:

- A filtered recordset to retrieve the record from a database table (see "Retrieving the record to update (PHP)" on page 728)
- 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 final two building blocks of an update page in a single operation using the Record Update Form application object. The application object adds both an HTML form and an Update Record server behavior to the page.

Before you can use the application object, your web application must be able to identify the record to update, and your update page must be able to retrieve it. See "Searching for the record to update (PHP)" on page 726, "Creating the links to open the update page (PHP)" on page 727, "Creating a URL parameter for the links (PHP)" on page 723 and "Retrieving the record to update (PHP)" on page 728.

After the application 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.

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.

To build the update page with the Record Update Form application object:

1 Open the update page in Design view, then select Insert > Application Objects > Update Record > Record Update Form Wizard.

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 application 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 (PHP)

An update page has three building blocks:

- A filtered recordset to retrieve the record from a database table (see "Retrieving the record to update (PHP)" on page 728)
- 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 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 "Searching for the record to update (PHP)" on page 726, "Creating the links to open the update page (PHP)" on page 727, "Creating a URL parameter for the links (PHP)" on page 723 and "Retrieving the record to update (PHP)" on page 728.

Completing the update page consists of three tasks:

- Adding an HTML form to the page to let users modify the data
- Displaying the record in the form by binding the form objects to database table columns
- Adding the Update Record server behavior to update the database table after the user modifies the record

To add an HTML form to an update page:

1 Create a new PHP page (File > New).

The page will become your update page.

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 selecting 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 such as a text field (Insert > Form > Text Field) 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 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, see "Inserting HTML form objects" on page 613.

6 Add a Submit button to the form (Insert > Form > Button).

   You can 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.

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 (PHP)" on page 728.

2 Bind each form object to data in the recordset, as described in the following sections:

   - "Displaying dynamic content in HTML text fields" on page 618
   - "Dynamically preselecting HTML checkboxes" on page 618
   - "Dynamically preselecting HTML radio buttons" on page 619
   - "Creating a dynamic HTML form menu" on page 617
   - "Making existing HTML form menus dynamic" on page 618

To add a server behavior to update the database table:

1 In the Server Behaviors panel (Window > Server Behaviors), click the Plus (+) button and select 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 clicking the Submit button on the form.

To edit the server behavior, open the Server Behaviors panel (Window > Server Behaviors) and double-click the Update Record behavior.

Related topics

- "Completing the update page in one operation (PHP)" on page 729.

Building pages to delete a record (PHP)

Your application can contain a set of pages that lets users delete records in a database. The pages normally consist of a search page, a results page, and a delete page. The search and results page let users retrieve the record and the delete page lets users delete the record.

This section describes the following steps to build PHP pages to delete records:

- "Searching for the record to delete (PHP)" on page 732
- "Creating links to a confirmation page (PHP)" on page 732
- "Creating a URL parameter to pass to the confirmation page (PHP)" on page 733
- "Displaying the record on the confirmation page (PHP)" on page 734
- "Adding logic to delete the record (PHP)" on page 737

Searching for the record to delete (PHP)

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. The user enters search criteria in the search page and selects the record on the results page. When the user clicks the record, the delete page opens and displays the record.

For instructions on creating pages to search for the record to delete, see "Building search/results pages (PHP)" on page 725.

After creating the search/results pages, the next step is to create links on the results page to open the deletion confirmation page.

Creating links to a confirmation page (PHP)

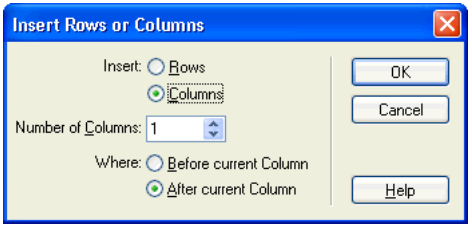
After creating the results page, you must create links that the user can click to open a page that asks the user to confirm the deletion. This section describes how to create the links. The next section describes how to modify the link so that it also passes the ID of the record the user wants to delete.

To create links to a confirmation page:

1 On the results page, create a new column in the table used to display records by clicking inside the last table column and selecting Modify > Table > Insert Rows or Columns.

The Insert Rows or Columns dialog box appears.

2 Select the Columns option, then select the After Current Column option.



After you click OK, Dreamweaver adds a column to the table.

3 In the newly created table column, enter the string `Delete` in the row containing the dynamic content placeholders. Make sure you enter the string inside the tabbed repeat region.

You can also insert an image with a word or symbol for delete.

If Live Data is turned on, enter the string in the first row of records and click the Refresh icon.

4 Select the `Delete` string to apply a link to it.

If Live Data is turned on, select the string in the first row of records.

5 In the Property inspector, enter the confirmation page in the Link field.

You can enter any filename you want.

After clicking outside the Link field, the `Delete` string appears linked in the table.

After creating the links, the next step in creating a record delete page is to create URL parameters for the links.

Creating a URL parameter to pass to the confirmation page (PHP)

The links on the results page not only have to open the confirmation page, they must pass the ID of the record the user wants to delete. The confirmation page will use this ID to find the record in the database and display it.

You must pass the record ID to the confirmation page with a URL parameter. This section describes how to create a URL parameter to pass the record ID to the confirmation page.

To create the URL parameter:

1 Select the delete link on the results page.

If Live Data is turned on, select the link in the first row.

2 In the Link field in the Property inspector, add the following string at the end of the URL:

```
?recordID=<?php echo $row_recordsetName['fieldName']; ?>
```

The question mark tells the server that what follows is one or more URL parameters. The word recordID is the name of the URL parameter (you can make up any name you like). Note the name of the URL parameter because you'll use it in the confirmation page later.

The expression after the equal sign is the value of the parameter. In this case, the value is generated by a PHP expression that returns a record ID from the recordset. A different ID is generated for each row in the dynamic table. In the PHP expression, replace `recordsetName` with the name of your recordset, and replace `fieldName` with the name of the field in your recordset that uniquely identifies each record. In most cases, the field will consist of a record ID number. In the following example, the field consists of unique location codes:

```
confirmDelete.php?recordID=<?php echo $row_rsLocations['CODE']; ?>
```

When the page runs, the values of the recordset's CODE field are inserted in the corresponding rows in the dynamic table. For example, if the Canberra, Australia, rental location has the code CBR, then the following URL will be used in the Canberra row in the dynamic table:

```
confirmDelete.php?recordID=CBR
```

3 Save the page.

After creating a dynamic URL parameter for the delete links, the next step is to display the record on the confirmation page.

Displaying the record on the confirmation page (PHP)

After completing the page listing the records, switch to the confirmation page. The confirmation page shows the record and asks the user if they're sure they want to delete it. When the user confirms the operation by clicking the form button, the web application deletes the record from the database.

Displaying the record consists of three tasks:

- Creating an HTML form
- Retrieving the record to display in the form
- Displaying the record in the form

Retrieving and displaying the record consists of defining a recordset to hold a single record—the record the user wants to delete—and binding the recordset columns to the form. The steps are outlined in more detail below.

To create an HTML form to display the record:

1. Create a new PHP page and save it as the confirmation page you specified in the previous section.

   You specified a confirmation page when you created the Delete link in the previous section. Use this name when saving the file for the first time (for example, deleteConfirm.php).
2. Insert an HTML form on the page (Insert > Form > Form).
3. Add a hidden form field to the form.

   The hidden form field is required to store the record ID passed by the URL parameter. To add a hidden field, place the insertion point in the form and select Insert > Form > Hidden Field.
4. Add a button to the form.

   The user will click the button to confirm and delete the displayed record. To add a button, place the insertion point in the form and select Insert > Form > Button.
5. Enhance the design of the page as desired and save it.

To retrieve the record the user wants to delete:

1. In the Bindings panel (Window > Bindings), click the Plus (+) button and select Recordset (Query) from the pop-up menu.

   The simple Recordset dialog box appears. If the advanced Recordset dialog box appears instead, click Simple to switch to the simple Recordset dialog box.
2. Name the recordset, and then select a connection and the database table that contains the records that could be displayed.
3. In the Columns area, select the table columns (record fields) you want to display on the page.

   To display only some of the record's fields, click Selected and select the desired fields by Control-clicking (Windows) or Command-clicking (Macintosh) them in the list.

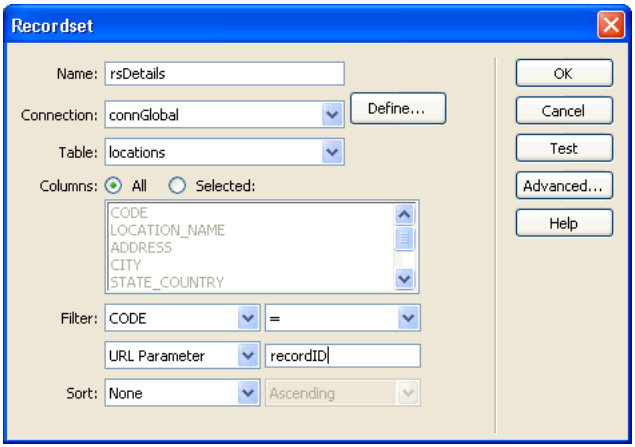
   Make sure to include the record ID field even if you won't be displaying it.

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 recordset containing values that match the value of the URL parameter passed by the page with the Delete links. For example, if the URL parameter contains a record ID number, select the column containing record ID numbers. In the example discussed in the previous section, the recordset column called CODE contains the values that match the value of the URL parameter passed by the page with the Delete links.
- From the pop-up menu beside the first menu, select the equal sign, if not already selected.
- From the third pop-up menu, select URL Parameter. The page with the Delete links uses a URL parameter to pass information to the confirmation page.
- In the fourth box, enter the name of the URL parameter passed by the page with the Delete links.

  For example, if the URL used to open the confirmation page included the suffix confirmDelete.php?recordID=CBR, then enter **recordID**.

  The Recordset dialog box should look as follows:



5 Click OK.

The recordset appears in the Bindings panel.

To display the record the user wants to delete:

1 Select the recordset columns (record fields) in the Bindings panel and drag them to the form you inserted on the page.

For more information, see "Making text dynamic" on page 547.

Next, you must bind the record ID column to the hidden form field.

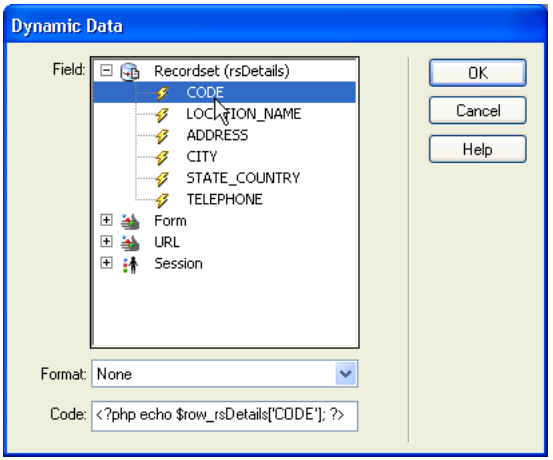
2 Make sure Invisible Elements are turned on (View > Visual Aids > Invisible Elements), then click the yellow shield icon representing the hidden form field.

The hidden form field is selected.

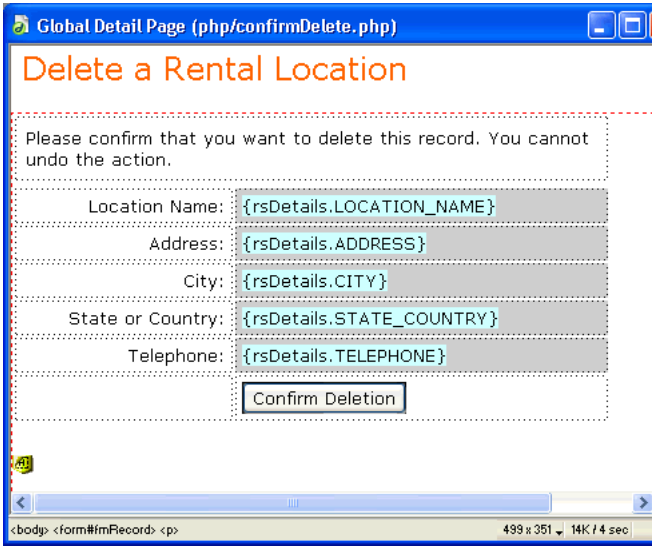
3 In the Property inspector, click the lightning bolt icon beside the Value text box.

The Dynamic Data dialog box appears.

4 In the Dynamic Data dialog box, select the record ID column in the recordset and click OK.
In the following example, the record ID column selected is CODE.



5 Save the page.
The completed confirmation page should look as follows:



After creating a confirmation page, the next step in creating a record delete page is to add logic to delete the record.

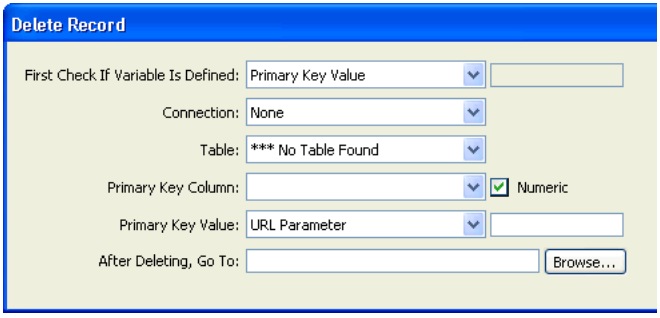
Adding logic to delete the record (PHP)

After creating a confirmation page, the next step in building a record delete page is to add logic to the confirmation page that deletes the record from the database when the user clicks the Confirm button. You can add this logic quickly and easily in Dreamweaver with the Delete Record server behavior.

To add logic to delete the record displayed in the HTML form:

1 In the Server Behaviors panel (Window > Server Behaviors), click the Plus (+) button and select Delete Record.

The Delete Record dialog box appears.



2 In the First Check If Variable Is Defined text box, make sure Primary Key Value is selected.

You specify the primary key value later in the dialog box.

3 In the Connection pop-up menu, select a connection to the database so that the server behavior can connect to the affected database.

4 In the Table pop-up menu, select the database table that contains the records that will be deleted.

5 In the Primary Key Column pop-up menu, select the table column that contains record IDs.

The Delete Record server behavior will search this column for a match. The column should contain the same record ID data as the recordset column you bound to the hidden form field on the page.

If the record ID is numeric, select the Numeric option.

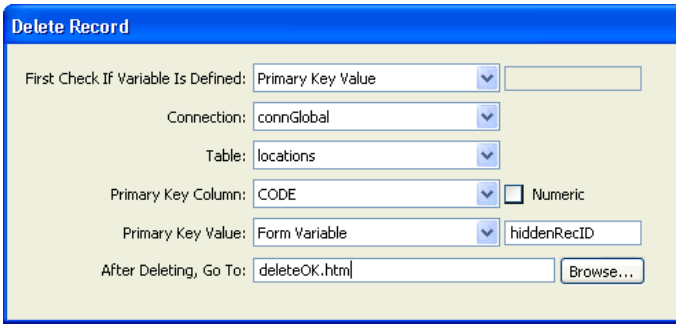
6 In the Primary Key Value pop-up menu, select the variable on your page that contains the record ID identifying the record to be deleted.

The variable is created by your hidden form field. It has the same name as the name attribute of the hidden field and is either a form or URL parameter, depending on the form's method attribute. In our example, the variable is a form variable called hiddenRecID.

7 In the After Deleting, Go To text box, specify a page to open after the record has been deleted from the database table.

You can specify a page that contains a brief success message to the user, or a page listing the remaining records so that the user can verify that the record has been deleted.

The completed Delete Record dialog box should look as follows:



8 Click OK and save your work.

Upload the pages to your web server (if required), open a browser and search for a record to delete. When you click a Delete link on the results page, the confirmation page appears. Click the Confirm button to delete the record from the database. To verify that the record has been deleted, open the page with the Delete links again. The record no longer appears in the list.

Note: Click Refresh if the record still appears on the page.

Building pages that restrict access to your site (PHP)

You can use Dreamweaver to build a set of pages that restrict access to your site. The methods you use are identical to the ones for ColdFusion. For instructions, see "Building pages that restrict access to your site (ColdFusion, ASP, JSP, PHP)" on page 666.

PART IX
Appendixes

Get more help to develop your web applications or to use Flash elements.

This part contains the following chapters:

Appendix A, “Beginner’s Guide to Databases,” on page 741

Appendix B, “SQL Primer,” on page 753

APPENDIX A
Beginner's Guide to Databases

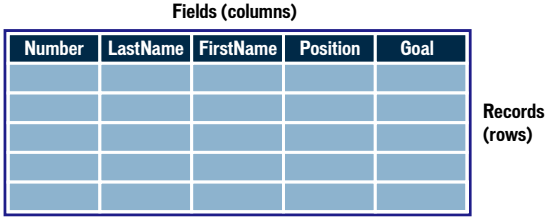
This appendix is intended for Macromedia Dreamweaver MX 2004 users who have little or no experience working with databases or database connections. It explains general concepts, not specific procedures. To see how these concepts apply in practice, see the rest of this guide.

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.

About databases

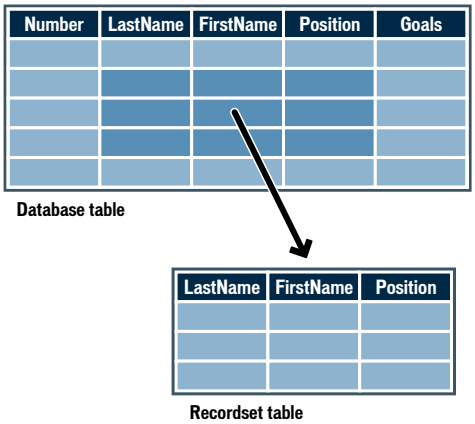
The building block of a database is the record. A record is a collection of related data treated as a single entity. For example, a hockey trading card could be called a record: it brings together the name, photograph, team, and statistics of one player. Using database terms, each of these related pieces of information is called a *field*: each hockey card *record* has a name field, a photograph field, a team field, and various player statistics 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.



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.



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 "Understanding recordsets" on page 526.

Related topics

- "Understanding database connections" on page 747

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.

Database design consists of a series of common steps. First, study the business rules and policies to be reflected in the web application. Second, consider the questions users will ask the database. Third, define the structure of the database. Finally, create the database.

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 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.

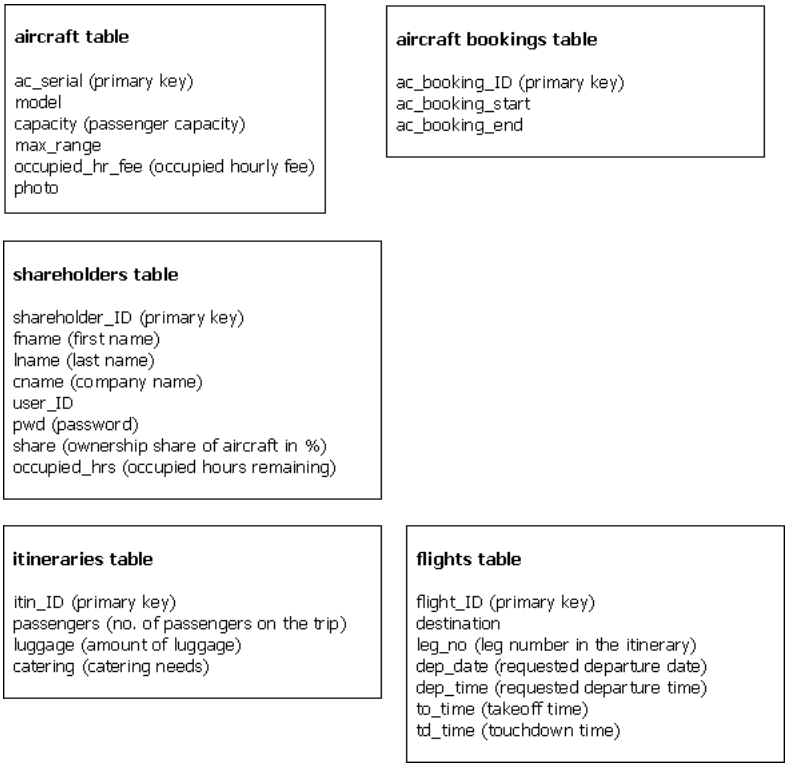
A good rule of thumb when choosing columns is to avoid duplicating data. For example, in the Arrow Aircraft application, you know shareholder names will have to be used in association with the itinerary data to answer the following Flight Ops question: "What are the shareholder's catering needs for a 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:



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 to combine data from two tables (see "Joining tables" on page 759).

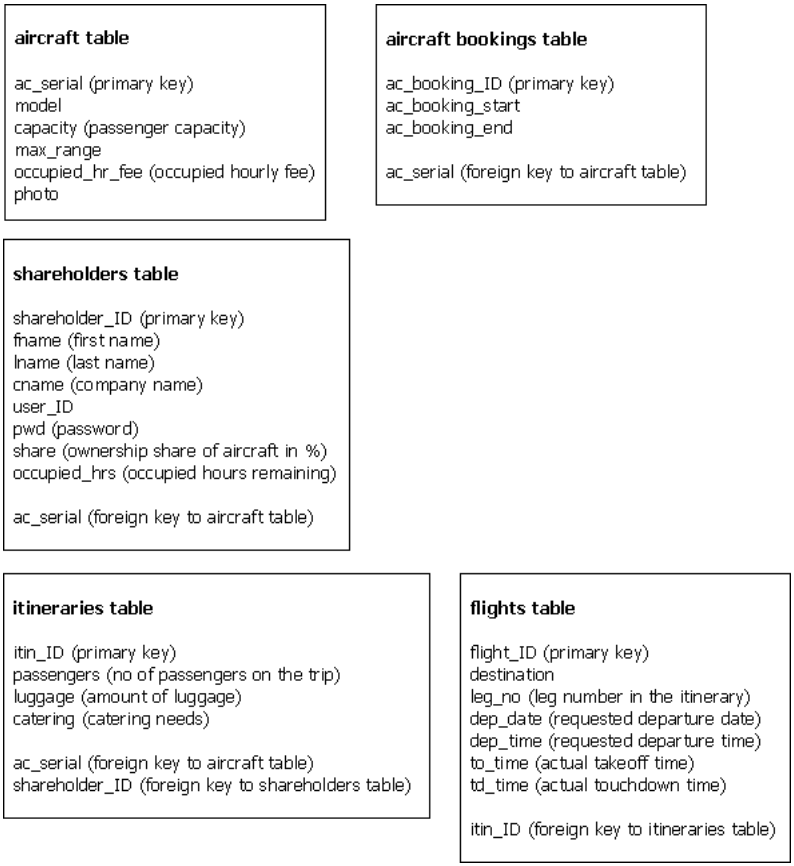
For example, each plane managed by Arrow Aircraft has several shareholders. You would like to establish a similar "one-to-many" relationship between each plane in your aircraft table and the shareholders in the shareholders table. This would save you from entering and tracking redundant aircraft data 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

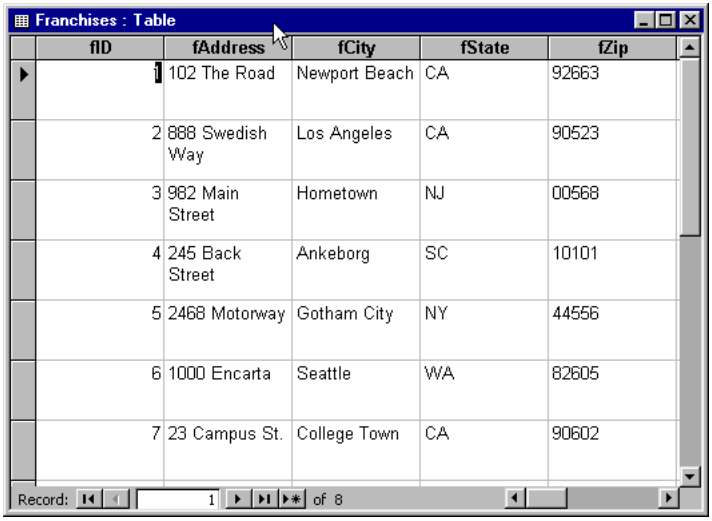
If you plan to use a database with your web application, you need to create at least one database connection. Without one, the application won't know where to find the database or how to connect to it. You create a database connection in Dreamweaver by providing the information—or the "parameters"—the application needs to establish contact with the database.

Note: You don't need to create database connections for ColdFusion pages in Dreamweaver. 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, see "Connecting to a database" on page 461.

Communicating with the database

Data stored in a database is normally in a proprietary format in the same way text in a word processor file is in a proprietary format. For example, here's what data looks like in Microsoft Access:



Here's what the same database looks like in Notepad:

Your web application faces the same problem as Notepad or any other application trying to access data in an unknown format: the application can't decipher the data. A software interface is needed between your web application and the database allowing the application and the database to talk to each other.

Three common interfaces let applications communicate with databases. The first is called Open Database Connectivity, or ODBC; the second is called OLE DB (object linking and embedding database); and the third is called Java Database Connectivity, or JDBC.

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 speak ODBC (through a built-in OLE DB/ODBC interpreter), ASP.NET applications speak OLE DB, and ColdFusion 4 or 5 applications speak ODBC and OLE DB.

Using database drivers to communicate 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 and Windows XP Professional. 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 www.microsoft.com/data/download.htm. MDAC 2.7 installs a number of OLE DB providers.

Note: Install MDAC 2.5 first, then install MDAC 2.7.

To find out which ODBC drivers are installed on your Windows system, see "Viewing the ODBC drivers installed on a Windows system" on page 749.

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	ColdFusion native drivers Sun JDBC-ODBC driver Oracle Thin JDBC driver
ASP	ODBC or OLE DB	Microsoft Access 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 98, select 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, select Start > Settings > Control Panel > Administrative Tools > Data Sources.
- In Windows XP, select 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
```

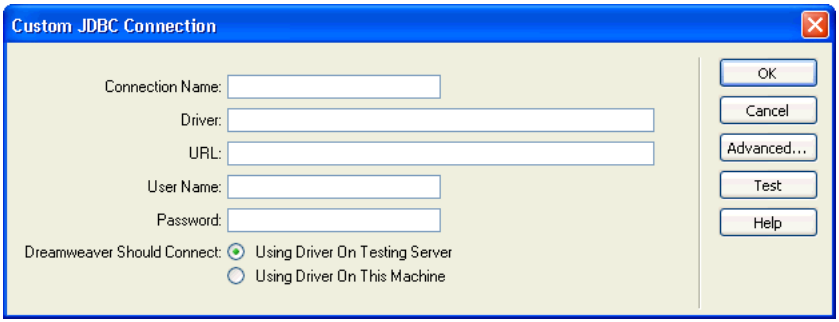
ASP connection strings can contain a Provider parameter specifying an OLE DB provider. If you omit this parameter, by default ASP uses the OLE DB provider for ODBC drivers. In the above example, the OLE DB provider for ODBC drivers would communicate with the ODBC driver, Microsoft Access Driver, which 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=Queequeg
```

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

You can specify data source names (DSNs) in some connections. A DSN is a type of shortcut you create in Windows to represent a connection string.

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 instead of all the other parameters.

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 JSP 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.

To set up a DSN in Windows, see the following articles on the Microsoft website:

- Windows 98 users, see Microsoft Knowledge Base Article 300595 at http://support.microsoft.com/default.aspx?scid=kb;en-us;300595
- Windows 2000 users, see Microsoft Knowledge Base Article 300596 at http://support.microsoft.com/default.aspx?scid=kb;en-us;300596
- Windows XP users, see Microsoft Knowledge Base Article 305599 at http://support.microsoft.com/default.aspx?scid=kb;en-us;305599

Related topics

- "Database Connections for ASP Developers" on page 469
- "Database Connections for JSP Developers" on page 479

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.

Related topics

- "About databases" on page 741
- "Database design basics" on page 742
- "Understanding recordsets" on page 526

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 741.

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 755
- "Filtering records based on the likeness of two values" on page 756
- "Filtering records based on a range of values" on page 757
- "Filtering records based on a combination of search conditions" on page 758

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 745.) 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.

Related topics

- "Defining relationships between the tables" on page 745

INDEX

Symbols

? in field names 493
@import 297

A

Absolute Bottom alignment 308
Absolute Middle alignment 308
absolute paths 316
access privileges
  adding to pages 672
  example 671
  storing in a database 672
accessibility 26–27
  dialog box, activating 53
  Federal Rehabilitation Act 26
  frames 205, 210
  images 306
  keyboard-only navigation 50
  media objects 359
  objects, inserting 622
  operating system features 49
  screen reader 49
  Web Accessibility Initiative (W3C) 26
accessing files
  on a Dreamweaver site 80
  on a server 80
  on local drive or desktop 81
accounts, troubleshooting account names 492
actions
  browser compatibility 376
  changing in behaviors 375
  choosing in Behaviors panel 373
  creating 375
  defined 371
  included with Dreamweaver 376–398
  See also individual actions by name
Active Links color option (Page Properties) 261
ActiveX controls
  inserting 368
  overview 368
ActiveX objects, making dynamic 551
Add Object to Library command 126
adding
  ASP.NET form controls 623
  dynamic content 545
  extensions to Dreamweaver 57
  links 120
  rows and columns 176
Adjust Position command 162
administrator passwords for Contribute sites 139
Align with Selection command 162
aligning
  images 283
  layers 158
  options 308
  page elements 308
  text 283
  tracing images 162
Always Nest When Created Within Existing Layer option 152
Anchor object (Insert bar) 324
anchors, named 324
applets. *See* Java applets
application objects
  Master/Detail Page Set 634, 675, 703, 719
  Record Insertion Form 647
  Record Update Form 652, 690, 709, 729
application server
  choosing 454
  setting up 454
application variables 537
applications, other, using with Dreamweaver 25
Apply Source Formatting command 403
Apply Template to Page command 248

applying
behaviors to images 313
colors to text 120
CSS styles 295
custom styles 295
ASP
application servers 456
command objects 716
command objects, defined 633
connecting to ISP 475
database connections 469
delete pages 711
DSN connections 471
DSN-less connections 474
insert pages 646
login pages 666
master/detail pages 634, 703
OLE DB connections 470
search pages 641, 706
stored procedures 715
update pages 707
ASP.NET
CheckBox 625
database connections 465
DataGrid 561, 570
DataGrid Delete Buttons 561, 570
DataGrid Edit, Update, Cancel Buttons 561, 570
DataGrid Free Form 561, 570
DataGrid Hyperlink 561, 570
DataGrid Simple Data Field 561, 570
DataList 570
delete pages 692
DropDownList 624
form controls, adding 623
form controls, modifying 623
importing tags 410
insert pages 646
installing the .NET Framework 456
ListBox 624
master/detail pages 675
RadioButton 626
RadioButtonList 626
search pages 681
stored procedures 700
TextBox 625
update pages 686
writing SQL for 527
assets
categories 118
colors, applying to text 120
colors, creating 123
editing 121
Favorites folder, creating 125
Favorites list 117, 118, 123
inserting 119
managing 116
opening the Assets panel 117
reusing in another site 122
selecting multiple 121
Site list 117, 118
URLs, creating 123
viewing 117
working with 116
Assets panel
opening 117
Templates category 228
Attach Style Sheet icon 297
attributes
editing with the Tag inspector 442
making dynamic 549
searching for 428
See also code
audio. *See* sound
authorization levels 671
Auto Refresh 504
auto-number fields, troubleshooting 494
automating tasks 265
autostretch (Layout mode) 186

B

background
page image and color 260
transparency in 260
balancing braces 434
Baseline alignment 308
basics of Dreamweaver 20
BBEdit integration (Macintosh only) 412
behaviors
actions, creating 375
attaching 373
changing 375
compatibility with browsers 376
defined 371
deleting 375
forms and 620
frames and 214
images and 313

JavaScript 371–398
library items and 130
links and 336
media and 369
third-party 376
triggering 373
Behaviors panel 372
Bindings panel
adding dynamic text 547
creating a record counter 570
deleting data sources 543
Format column 562
making forms dynamic 610, 617
making HTML attributes dynamic 549
bitmaps
resizing 309
bitmaps, resizing
See also images
blockquote, applying 283
bold text 286
Bottom alignment 308
braces, balancing 434
broken links 336
Browser Default alignment 308
browsers
colors, safe 255
compatibility, testing 434
compatibility, with frames 214
previewing in 270
primary, defining 271
targeting 434
versions, checking 378
buttons
about 609
Go buttons 384
inserting 615

C

caching data sources 542
Call JavaScript action 377
callable, JSP 715
cascading styles. *See* style sheets
case, changing 403, 430
catalogs 506
categories
assets 118
preferences 54
cells
highlighting 184
See also layout cells, table cells
CGI reference material 29
Change Link Sitewide command 330
Change Property action 377
changing
Files panel column order 84
Files panel viewing area 83
site views in Files panel 84
characters, valid in account names 492
Check Browser action 378
Check In/Check Out
checking files in and out 99
finding checked out files 88
overview 78
setting up 98
undoing a file check-out 100
using 97
Check Links Sitewide command 337
Check Plugin action 379
Check Spelling command 301
Check Target Browsers command 434
CheckBox control 625
checkboxes 609, 613
checking links 336
Clean Up HTML command 433
Clean Up Word HTML command 74
clearing
column widths 175
table widths and heights 175
cloaking
enabling and disabling 105
files 106
overview 78
site folders 105
uncloaking all 107
code
case, changing 403
cleaning up 433
coding environment, customizing 401
coding guidelines 596
coloring preferences 405
comments 427
copying and pasting 427
external editors 412
in external files 447
formatting 403
formatting preferences 403
head section of a document 449
indenting 403, 408, 426, 430
invalid 416
languages, supported 416

line numbers 403
navigation 428
printing 432
reference 431
rewriting automatically 417
rewriting preferences 404
searching 428
selecting in Contribute 553
server behaviors 423
snippets 424
tag editors 426
tag libraries 406
validation preferences 405
viewing 401
viewing preferences 403
word wrap 403
wrapping selection using the Quick Tag Editor 444
writing and editing 423
XHTML 417
code blocks
coding guidelines 596
parameter markers 603
positioning 602
writing 598
Code Coloring preferences for templates 228
code hints
overview 423
preferences 404
Quick Tag Editor 445
tag libraries and 406
Code inspector 401
code snippets, adding keyboard shortcuts for 57
Code view
and Design view, specifying which view appears on top 35
displaying 401
opening non-HTML files 405
options 403
toggling to Design view 39
viewing template-based documents 219
viewing templates 218
writing and editing code 423
code, editing
with BBEdit 412
with HomeSite 412
with the Property inspector 441
with the Quick Tag Editor 444
with the Tag inspector 442
code, inserting
in Design view 446
with the Insert bar 425
with the Quick Tag Editor 443
with the Tag Chooser 425
coding guidelines 596
ColdFusion
client variables 529, 539
ColdFusion variables 539
components 573–577
Components panel 577
data sources, creating 464
database connections 463
delete pages 655
insert pages 646
installing 455
login pages 666
master/detail pages 634, 635
pages that use components 577
pages, debugging 439
search pages 641
stored procedures 665
update pages 649
ColdFusion Markup Language (CFML)
debugging 176
ColdFusion Markup Language (CFML), debugging 439
collapsing the Files panel 83
color box option 260
Color Cubes color palette 261
color picker
Dreamweaver 260
system 261
Color Wheel (System Color) button 261
colors
as assets. *See* assets
choosing 260
code, preferences 405
default for page text 261
eyedropper, using 260
frame background 212
page background 260
swatches 260
text, changing 120, 288
web-safe 255
column header menu
about 164
Add Spacer Image 196
Clear Column Width 175
displaying 176, 185

Insert Column Left 176
Insert Column Right 176
Make Column Autostretch 195
Make Column Fixed Width 195
Remove Spacer Image 196
Select Column 169
combining floating panels 46
command objects, ASP 716
commands
accessing from context menu 46
creating, from history steps 268
recording 269
Revert 257
Save All 257
Save As 257
comments
commenting out code 430
inserting and editing 427
Components panel 577
connecting to databases 461
connection keys, exporting 138
connection parameters 479
connection string, OLE DB 467
connection strings 750
connections to Contribute sites, troubleshooting 140
content blocks
adding content to 149
editing 149
highlighting preference 148
inserting 147
content, adding to tables 165
context menus 46
Continuous Tone color palette 261
Contribute
administration settings, changing 138
compatibility, enabling 138
CSS styles 300
Dreamweaver integration and 133
dynamic content 553
files, managing 141
files, transferring 136
files, unlocking 142
passwords, administrator 139
permission groups, changing 138
permissions on servers 136
site definitions, exporting 138
site structure, creating 134
sites, managing with Dreamweaver 133, 134, 138
special files, working with 137
templates 229, 244
troubleshooting 140
Control Shockwave or Flash action 380
conventions 29
Convert Layers to Tables command 160
copying
assets to another site 122
history steps 268
copying and pasting
Fireworks HTML 347
recordsets 543
table cells 179
Create Web Photo Album command 349
creating
colors and URLs 123
document based on design file 70
document based on template 71
Favorites list for assets 123
frames 204
library items 125
links 320
links to a Microsoft document 281
new blank documents 70
new CSS style 294
null and script links 326
rollover images 312
templates 226
temporary files 491
XHTML pages 438
CSS styles
applying custom (class) 295
class attribute 294
creating 294
exporting 296
removing 292
removing style from a selection 296
updating in a Contribute site 300
CSS Styles panel 291
custom styles. *See* style sheets
custom tags, importing 410
customizing
coding environment 401
Dreamweaver, basics 54
panel layout 46

D

data formats
  applying 562
  creating 563
  editing 562
Data Link Properties 466
Data Source Name. *See* DSN
data sources
  about 526
  application variables 537
  caching 542
  ColdFusion variables 539
  defining in ColdFusion 464
  deleting 543
  editing or deleting 543
  form parameters, about 517
  JavaBeans 540
  JSP variables 540
  recordsets, about 526
  recordsets, creating (simple) 531
  session variables 536
  session variables, about 520
  submitted by users 516
  URL parameters, about 518
data types, mismatched 493, 494
database connections
  ASP 469
  ASP.NET 465
  ColdFusion 463
  connecting 461
  JDBC 479
  JSP 479
  MySQL 487
  OLE DB 465, 470
  OLE DB parameters 467
  PHP 487
Database Items tree 533
databases
  about 741
  connecting to 461
  connection strings 750
  connections, about 747
  content storage 515
  delete pages 711
  design 742
  drivers 747
  insert pages 646
  locked 491
  login pages 669
  MySQL 487
  permissions, changing 490
  providers 747
  records 741
  relational 745
  results pages 641, 681
  schemas and catalogs 506
  search pages 641, 681
  SQL 753
  stored procedures 665, 700, 714
  tables 741
  update pages 649
dates
  inserting 289
  troubleshooting 493
debugging ColdFusion pages 439
Default Color button 261
delete pages
  ASP 711
  ASP.NET 692
  ColdFusion 655
  JSP 711
  PHP 732
Delete Record behavior 714
deleting
  data sources 543
  dynamic content 513, 552
  files and folders 87
  files in a Contribute site 141
  library items 128
  recordsets 543
  rows and columns 176
dependent files, showing and hiding in site map 94
design files 70
design frames 201
Design Notes
  about 108
  adding status choices 110
  deleting unassociated 110
  enabling and disabling 108
  for Fireworks files 79
  for Flash files 79
  for media objects 360
  opening 110
  overview 79
  saving file information 109
  setting up 108
design schemes, to format tables 172
Design view
  displaying with Code view 402
  JavaScript, inserting 446

refreshing 402
toggling to Code view 39
VBScript, inserting 446
viewing template-based documents 218
viewing templates 217
Design-Time style sheets, using 300
desktop, accessing files on 81
destroying temporary files, permissions for 491
Detach from Original option 130
Detach from Template command 248
detail pages 634, 675, 703, 719
Director, creating Shockwave movies with 365
disabling
cloaking 105
Design Notes 108
Display Dependent Files option 95
display options
floating panels 48
fonts 56
displaying
code hints 423
FTP log 102
layers 156
multiple records 566
table and column widths 176
tracing images 162
div tag, inserting 147
docking panels and panel groups 46
Document toolbar 34, 35
Document Type Definition 410
Document window
about 34
basics 34, 39
page size and download time 36
playing Navigator plug-ins 367
resizing 40
searching for text 301
selecting elements 262
status bar 36
tag selector 36
title bar 35
viewing code 39
Window Size pop-up menu 36
document-relative paths
about 316
setting 321
documents
applying a template 247
checking links 336
creating blank 70
creating, based on design file 70
creating, based on template 71
Design Notes, using with 108
detaching a template 248
download size, time 272
importing Microsoft Word 74
opening 73
page title 259
previewing in browsers 270
saving 72
saving as a template 226
searching 301
setting a default type for new 72
setting properties 258
downloading
behaviors 376
files 101
size, time estimates 272
time 36
Drag Layer action 380
drawing
layers 150
layout cells and tables 187
nested layers 151
Dreamweaver
basics 20
Contribute integration and 133
discussion group 28
Help 27
site. *See* sites 60
Dreamweaver Support Center 28
Dreamweaver, Fireworks integration and
creating web photo albums 349
inserting Fireworks files 342
Launch and Edit preferences 347
launching and editing Fireworks images 343, 351
launching and optimizing Fireworks images 344
Optimize Image in Fireworks command 344
updating Fireworks HTML 348
drivers, not specified 491
dropdown menus 609, 614
DropDownList control 624
DSN
connections without 474
creating a connection 471
troubleshooting 491
using 751
DTD files 410

dynamic content
about 526
adding to pages 545
advanced recordset, creating 532
attributes 549
deleting from a page 513
form checkboxes 618
form menus 617, 618
form text fields 618
forms 610, 617
images 548
objects 551
radio buttons 619
recordset, creating 531
recordset, defining 526
removing 552
replacing 546
selecting in Contribute 553
text 547
Dynamic Data dialog box 550

E

e-mail files 73
e-mail links
changing 329
creating 325
Edit Font List command 287
Edit Format List dialog box 562
Edit NoFrames Content command 214
Edit Style Sheet dialog box 299
Edit with BBEdit command 412
Edit with HomeSite command 412
editable regions (templates)
changing the name of 234
controlling in nested templates 241
creating 232
defined 216
removing 233
selecting 233
editable tag attributes (templates)
defined 217
making uneditable 240
modifying in template-based documents 249
setting 240
editing
ASP.NET form controls 623
assets 121
code 423
CSS style sheet 299
data sources 543
Dreamweaver site settings 67
files in a Dreamweaver site 80
files on a server 80, 82
Flash button objects 362
keyboard shortcuts 57
library items 127
recordsets 543
server behaviors 605
style sheets, external 298
templates 242
websites, existing local 67
websites, existing remote 68
editors. *See* external editors
elements, aligning 308
enabling
cloaking 105
Design Notes 108
encodings 56
error messages
file already in use 491
Microsoft, troubleshooting 491
See also troubleshooting
event handlers. *See* events
events
available for different browsers and objects 372
changing in behaviors 375
defined 371
triggering actions 373
Excel. *See* Microsoft Excel files
Expanded Tables mode 170
expanding the Files panel 83
Export Table command 167
exporting
connection keys for Contribute 138
sites 96
styles, to create a CSS style sheet 296
table data 166
Expression Object Model (templates) 225
extensibility, third-party behaviors 376
Extension Manager 57
extensions
adding 57
creating 590
installing 598
external editors
BBEdit (Macintosh only) integration 412
HomeSite (Windows only) integration 412
HTML 412
images 313

media 359
text 412
external links 336
external style sheets
creating 296
editing 298
linking to 296
eyedropper 260

F

Favorites folder 125
Favorites list (Assets panel)
about 117
adding a new color 123
adding a new URL 123
adding and removing assets 123
creating a Favorites folder 125
creating a nickname for 124
viewing 118
features, new in Dreamweaver 18
"file already in use" error message 491
file formats, image 303
file server-side includes 419
file transfer preferences 86
file types
external editor preferences and 413
Flash files 356
File Types/Editors preferences 360
file-upload fields 616
files
cloaking and uncloaking 106
creating 70
deleting 87
Design Notes, using with 108
downloading 101
locating checked out files 88
locating newer in local or remote 88
locating recently modified 89
managing in Contribute sites 141
moving 87
opening 73, 87
opening in Code view 405
putting 102
renaming 87
saving 72
searching 301
synchronizing local and remote sites 103
text 73
unlocking in Contribute sites 142
uploading 102
Files panel
changing column display 84
changing the display 84
changing the site view 84
changing view area size 83
collapsing 83
creating files and folders 87
deleting files and folders 87
expanding 83
moving files and folders 87
opening and closing 83
opening files 87
preferences 86
refreshing 88
renaming files and folders 87
searching for files 88
site maps 89
viewing files in 83
finding and replacing. *See* searching
Fireworks
Design Notes in 79
Dreamweaver integration and 342
Flash button dialog box 361
Flash button objects
modifying 362
previewing 362
Flash components
downloading new 364
editing properties 365
inserting 364
installing 364
Flash content 356
Flash objects, making dynamic 551
Flash SWF files
as assets. *See* assets
controlling 380
inserting 363
Flash Text dialog box 363
Flash text objects
inserting 363
previewing 362
Flash, Design Notes in 79
floating panels, combining 46
folders
cloaking and uncloaking 105
creating 87
deleting 87
Favorites, for assets 125
moving 87
renaming 87

searching 301
searching text 301
font characteristics, changing 286
fonts
changing characteristics 286
changing combinations 287
encodings, setting fonts for 56
Fonts/Encoding preferences 56
form controls, ASP.NET 623
Format column 562
Format Table command 172
formatting
code, applying formatting 403
code, preferences 403
tables and cells 170
forms
accessibility 622
adding to a document 611
ASP.NET 623
buttons 609, 615
checkboxes 609, 613
client-side scripting 621
creating 611
dynamic checkboxes 618
dynamic menus 617, 618
dynamic objects 610, 617
dynamic radio buttons 619
dynamic text fields 618
fields, validating 397
file fields 610, 616
hidden fields 609, 616
image fields 610, 615
inserting objects 613
JavaScript, using with 620
jump menus, creating 331
menus 609, 614
overview 607
password fields 613
radio buttons 609, 614
server-side scripting 608
text fields 608, 613
using to gather data 641, 681
validation 620
frames 200–214
adding accessibility attributes 205
browser compatibility 214
changing background color 212
changing content 213
creating 204
defined 200
deleting 206
editing accessibility attributes 210
Frames panel 207
nested 202
overview 200
planning 201
properties 210
resizing 206
saving 209
selecting 207
steps in creating 203
targeting 213
using with behaviors 214
Frames panel
opening and closing 207
selecting in 207
framesets 204–214
creating 205
defined 200
naming 211
nested 202
predefined, inserting 204
properties 212
saving 209
selecting 207
targeting links in 320
See also frames
FTP
getting and putting files 101
log 101
log, displaying 102
preferences 86
troubleshooting 66
functions, viewing 428

G

gathering data from users 641, 681
General preferences 56
Generator objects, making dynamic 551
Get command 101
Get More Behaviors command 376
getting and putting files
getting files from remote server 101
putting files on remote server 102
GIF images
as tracing image 161
uses for 303
Go button, associating with a jump menu 384
Go To URL action 383
graphics. *See* images

Grayscale color palette 261
grid
  as guide 54
  showing 160
  snapping layers to 160

H

head section, viewing and editing 449
Help 27
hidden fields 616
hidden files, showing and hiding 94
hidden form fields 609
Hide Pop-up Menu 395
hiding
  invisible elements 263
  links 566
Highlighting preferences
  content blocks 148
  layout cells 189
  library items 129
  tables 167
  template regions 229
hints menus for code 423
History panel
  automating tasks with 265
  commands, creating from history steps 268
  history list, clearing 264
  maximum steps, setting 264
  overview 255
  steps, applying to other objects 266
  steps, copying and pasting 268
  steps, repeating 265
home page, setting for site map 90, 93
HomeSite 412
horizontal rules, inserting and modifying 284
hotspots
  applying behaviors to 313
  in image maps 335
  resizing 336
  selecting multiple in an image map 336
HTML
  attribute reports 112
  attributes, making dynamic 549
  converting to XHTML 438
  formatting 273, 280
  inserting 273, 280
  non-breaking space 290
  Roundtrip 417
  searching 301
  source code, searching 301
  source code, tag styles 290
  tag styles 290
  See also code
HTML forms. *See* forms
HTTP server 454
Hyperlink dialog box 323
hypertext links 320

I

IBM WebSphere 457
IIS, server-side includes 419
illegal characters in account names 492
image buttons 615
image maps
  creating client-side 335
  hotspots 335
  overview 335
  selecting multiple hotspots 336
images
  about 303
  aligning 283
  applying behaviors to 313
  as assets. *See* assets
  brightness 311
  contrast 311
  cropping 310
  editing 313
  external image editors 313
  formats, supported 303
  in forms 610
  image maps 335
  inserting 305
  inserting in Layout mode 190
  making dynamic 548
  optimizing using Fireworks 310
  preloading (behavior) 387
  resampling 309
  restoring swapped (behavior) 397
  scalability 309
  sharpening 311
  swapping (behavior) 396
Import Table command 166
Import Word HTML command 74
importing
  ASP.NET tags 410
  custom tags 410
  external CSS style sheet 297
  JRun tags 411
  JSP tags 411
  Microsoft Word files 74

sites 96
tabular data 166, 281
text from other documents 281
includes, server-side 419
indenting code 403, 408, 426, 430
Insert bar
about 34
categories 37
docking 46
inserting code 425
preferences 44
showing categories as tabs 43
using 42
Insert E-Mail Link dialog box 325
Insert Jump Menu dialog box 331
Insert Named Anchor dialog box 324
Insert Navigation Bar dialog box 333
insert pages 646
Insert Record behavior 648
inserting
ActiveX controls 368
assets 119
comments 427
dates 289
div tags 147
Fireworks images into Dreamweaver 342
Flash button objects 361
Flash SWF files 363
Flash text objects 363
image placeholder 307
images 305
Java applets 368
library items 126
media elements 358
nested layers 151
rollover images 312
server-side includes 448
Shockwave movies 365
special characters 289
inspectors
Property inspector 45
See also panels
integrating Dreamweaver with other applications 25
invalid tags, displaying 417
invisible elements
comments 427
scripts 446
selecting 262
showing and hiding 263
ISP 475
italics 286

J

Jakarta Tomcat 457
jar files 411
Java applets
inserting 358, 368
making dynamic 551
overview 368
JavaScript
actions 372
alerts 386
attaching to form objects 620
behaviors 371
executing 377
files 73
inserting code in Design view 446
processing form data 621
JavaServer Pages. *See* JSP
JDBC
connection parameters 479
drivers 481, 747
JPEG images
as tracing image 161
uses for 303
JRun 457
JRun tags 411
JSP
application servers 457
callable 715
database connections 479
delete pages 711
importing tags 411
insert pages 646
JavaBeans 540
JDBC drivers 481
JSP variables 540
login pages 666
master/detail pages 634, 703
ODBC connections 482
prepared statements 718
prepared statements, defined 633
resultset, defined 508
search pages 641, 706
stored procedures 715
update pages 707
Jump Menu action 384
Jump Menu Go action 384

jump menus
about 609
adding menu items 331
changing menu items 332
creating a selection prompt for 331
editing 384
Go buttons 331, 384

K

Keyboard Shortcut Editor 57
keyboard shortcuts, editing 57

L

languages
reference 431
supported in Dreamweaver 416
Launch and Edit preferences 347
Launcher bar, customizing 48
launching an external image editor 313
layers
aligning 158
changing stacking order of 155
changing visibility with behaviors 391
changing visibility with Layers panel 156
converting to tables 160
creating 150
displaying markers 150
draggable 382
drawing 150
drawing nested 151
inserting 150
inserting nested 151
manipulating 157
moving 158
nesting 151
preferences 152
preventing overlap 159
properties for multiple 153
properties for single 152
resizing 157
selecting 154
selecting multiple 153
showing and hiding borders 150
snapping to grid 160
in table design 160
visibility 156
Layers panel 153
layout cells
about 184
alignment 194
bg color 194
clearing heights 192
drawing 187
formatting 194
highlighting preference 189
No Wrap option 194
preferences 197
Layout mode
about layout cells and tables 184
adding content to 190
autostretch 186
cell padding 194
cell spacing 194
clearing cell heights 192
column header menu 195
drawing layout cells and tables 187
fixed width 186
formatting layout cells 194
formatting layout tables 194
Make Widths Consistent option 194
moving layout cells and tables 192
nested layout tables 189
overview 184
preferences 197
Remove All Spacers option 194
Remove Nesting option 194
resizing layout cells and tables 192
selecting layout cells and tables 192
setting width 186
spacer images, preferences 186
switching into 187
switching out of 187
layout tables
about 184
alignment 194
bg color 194
cell padding 194
cell spacing 194
clearing heights 194
drawing 187
formatting 194
Make Widths Consistent option 194
nested 189
preferences 197
Remove all Spacers option 194
Remove Nesting option 194
layout, planning. *See* Layout mode

learning Dreamweaver 20
Left alignment 308
library items
about 115
adding to pages 126
as assets. *See* assets
changing highlight color 129
creating 125
deleting 128
editing 127
editing behaviors in 130
making editable in documents 130
overview 115
preferences 129
properties 129
recreating 128
renaming 128
working with 125
line numbers 403
lines, wrapping in Code view 403
Link Checker dialog box 338
Link Color option (Page Properties) 261
Link External Style Sheet dialog box 297
link href tag 297
Link to Existing File command 93
Link to New File command 93
linked documents, opening 339
linking
anchors 324
to document using Point-to-File icon 322
documents 320
to external CSS style sheet 297
Microsoft Word or Microsoft Excel documents 281
to named anchor using Point-to-File icon 325
navigation and 315
links
to anchors 324
applying to selection 120
broken 336
cache file 327
changing frames with 213
changing sitewide 329
checking 336
creating 320
to document 320
fixing 338
hiding 566
to Microsoft documents 281
opening source 328
Relative to Document option 321
Relative to Site Root option 321
removing 328
to script files 447
site map 328
to style sheets 296
targeting 320
in templates 221
updating 327
list menus 609, 614
ListBox control 624
lists, creating 285
Live Data Settings dialog box 503
Live Data window
about 512
Auto Refresh 504
described 500
missing files 502
providing expected parameters 503
URL parameters on toolbar 503, 512
live objects
Recordset Navigation Bar 563
Recordset Navigation Status 569
local drive, accessing files on 81
local folder
setting up 63
structure 60
local sites. *See* sites 62
locked database files 491
locked regions, clicking in 248
logging out users 673
login pages 669
logon failures 492

M

Mac OS color palette 261
Macromedia Director, creating Shockwave movies with 365
Macromedia Exchange 57, 598
Macromedia HomeSite 412
Macromedia JRun 411, 457
macros, creating commands 268
managing
assets. *See* assets
extensions 57
links 327
panel groups 46
map tag 335
markers for invisible elements 263
markup. *See* code

Master/Detail Page Set application object 634, 675, 703, 719
master/detail pages
  ASP 634, 703
  ASP.NET 675
  ColdFusion 634, 635
  JSP 634, 703
  PHP 634, 719
MDAC (Microsoft Data Access Components) 466
media elements, inserting 358
menus 614
Merge Cells command 178
Microsoft Access, locked database files 491
Microsoft error messages, troubleshooting 491
Microsoft Excel documents, dragging content to a page 281
Microsoft Excel files, importing 166
Microsoft IIS 456
Microsoft Knowledge Base articles 492
Microsoft OLE DB providers 466
Microsoft technical support 491
Microsoft Word documents
  dragging content to a page 281
  importing 74
Middle alignment 308
modes
  Expanded Tables mode 170
  Layout mode 183
modifying
  databases, using stored procedures 665, 700, 714
  page properties 259
monitor size, resizing pages to fit 40
Move to Record behavior 565
movies, inserting 358
moving files and folders 87
MPEG movies, as assets. *See* assets
multimedia. *See* media elements
multiuser systems 38
MySQL 487

N

Named Anchor command 324
named anchors, creating 324
navigating code 428
navigation bar
  adding images to 333
  creating 319
  image states 319
  modifying elements 334
navigation links for records 563
nested layers
  drawing 151
  inserting 151
nested templates
  controlling editable regions 241
  creating 241
  overview 222
nesting
  frames 202
  layers 151
  setting layer preferences 152
  tables 181
Netscape Navigator 4 layer compatibility 152
Netscape Navigator plug-ins
  playing in Document window 367
  troubleshooting 368
new document default type 72
New Document preferences 72
new features in Dreamweaver 18
New From Template command 71, 72
non-breaking space, inserting 290
non-HTML files, opening 405
null links
  changing 329
  creating 326
numbering lines of code 403

O

objects
  inserting in forms 613
  inserting with Insert bar 42
  making dynamic 551
ODBC drivers 747, 749
ODBC errors 494
OLE DB
  connections 470
  obtaining a provider 465
  providers 747
OLE DB Connection 466
onBlur event 397
online forums 28
Open Attached Template command 243
Open Browser Window action 384
Open command 73
Open Linked Page command 339
opening
  Assets panel 117
  Dreamweaver site 80
  existing documents 73
  files 87

linked documents 339
non-HTML files 405
text files 73
operating systems, multiuser 38
Optimize Image in Fireworks command 344
optional regions (templates)
defined 216
inserting 237
modifying 239
Oracle OLE DB providers 466
Oracle Thin Driver 479
orphaned files 336
overlapping tags 417

P

page layout. *See* Layout mode
page properties, changing titles 259
pages
background image 260
changing title 259
color 260
default text colors 261
delete 711
Design Notes, using with 108
download time, estimated 272
insert 646
login 669
previewing in browsers 270
resizing to fit monitor 40
restricting access to 671
size 272
update 649
user registration 666
XHTML, creating 438
palettes, color 260
panel groups 34
panels
Assets, Templates category 228
Behaviors 372
docking 46
Frames 207
History 255
setting floating preferences 48
paragraphs
adding a line break 284
formatting 283
parameters
markers 603
templates 224
too few 493
Parameters dialog box 551
passwords
checking during login 670
Contribute administrator 139
fields 613
incorrect 492
letting users choose 667
storing 667
pasting history steps 268
paths
absolute 316
document-relative 316
physical 475
root-relative 318
Perl 416
permissions
creating and destroying temporary files 491
folder containing database 492
groups in Contribute 138
security 490
on servers 136
troubleshooting 489
write permissions on servers 136
photographs 303
PHP
database connections 487
delete pages 732
insert pages 646
installing PHP 457
login pages 666
Mac OS X 457
master/detail pages 634, 719
search pages 641
update pages 726
pixels, transparent, in background 260
planning frames 201
Play Sound action 386
playing Flash objects 362
plug-ins
checking 379
making dynamic 551
playing in Document window 367
troubleshooting 368
PNG images
as tracing image 161
uses for 303
pop-up menus 609, 614
Popup Message action 386
positioning code blocks 602

preferences
changing highlighting 148
code coloring 405
code formatting 403
code hints 404
code rewriting 404
Code view 403
dictionary for spell checking 301
external editors 360
file types and editors 413
File Types/Editors 360
floating panels 48
Fonts/Encoding 56
General 56
Highlighting, content blocks 148
Highlighting, layout cells 189
Highlighting, libraries 129
Highlighting, tables 167
Highlighting, template regions 229
Insert bar 44
Launch and Edit 347
Layers 152
Layout mode 197
New Document 72
overview 54
Panels 48
Site 86
Status Bar 41
template 229
updating links 327
Validator 405
Preload Images action 387
prepared statements, JSP 718
Prevent Layer Overlaps command 159
Preview in Browser command 270
previewing in browsers 270
printing code 432
properties
changing with behaviors 377
displaying 45
document, setting 258
frame 210
frameset 212
layout cell 194
layout table 194
library item 129
multiple layer 153
single layer 152
table 171
template-based document 249
Property inspector
about 34
displaying 45
editing a recordset 513, 552
editing code 441
expanding 45
fixing broken links 338
making HTML attributes dynamic 549
Standard mode 549
proportion, maintaining 309
Put command 102
putting and getting files 101
putting files on a remote server 102

Q

queries, troubleshooting 493
question marks 493
Quick Tag Editor
hints menus 445
opening 443
QuickTime movies
as assets. *See* assets
inserting 367

R

radio buttons 609, 614
RadioButton control 626
RadioButtonList control 626
read permissions on servers 136
record counter, building 569
Record Insertion Form application object 647
record navigation bar
creating 563
hiding 566
Record Update Form application object 652, 690, 709, 729
recording commands 269
records
building a counter 569
deleting 711
displaying more than one 566
inserting 646
navigation links 563
updating 649
Recordset dialog box
advanced 532
simple 531
Recordset Navigation Bar live object 563
Recordset Navigation Status live object 569

recordsets
about 526
caching 542
columns, defining (SQL) 755
copying and pasting 543
Database Items tree 533
defining without SQL 531
editing or deleting 543
empty, troubleshooting 494
filtering records (SQL) 755, 758
joining tables (SQL) 759
limiting records returned (SQL) 755
Recordset dialog box (simple) 531
simple, creating 531
SQL, writing 753
SQL, writing custom statements 532
Reference panel 431
Refresh Local command 94
refreshing
Design view 402
Files panel 88
Site list (Assets panel) 118
regions
clicking in locked 248
hiding 566
registration page 666
regular expressions 420
remote folder
setting up 65
structure 60
troubleshooting 66
removing
empty tags 433
rows and columns 176
sites from Site list 97
Rename Panel Groups command 48
renaming
files and folders 87
library items 128
Repeat Region behavior 566
repeating regions (template-based documents) 250
repeating regions (templates)
alternating colors 236
creating 234
defined 216
repeating table 236
in template-based documents 250
repeating steps 265
repeating tables (templates)
alternating colors 236
inserting 236
replacing an image placeholder 307
Replay button 265
reports
saving 114
for sites 112
validating code 437
viewing 114
requirements, web applications 453
Reset Origin command 160
Reset Position command 162
resizing
frames 206
handles 309
images 309
layers 157
layout cells and tables 192
table cells 173
resources for information on web technologies 29
restricting site access 666
restricting tables 506
results pages 641, 681
Results panel group
Target Browser Check panel 434
Validation panel 437
resultset, JSP 508
return, adding a paragraph 284
reusing
assets 122
code 424
library items 126
searches 429
reverting to last saved version 257
rewriting code 417
Right alignment 308
roadmap, where to start 20
rollovers
about 312
creating 312
root-relative paths
about 318
setting 321
Roundtrip HTML 417
rulers 160

S

Save All command 257
Save All Frames command 210
Save command 72
Save Frame As command 209
Save Frame command 209
Save Frameset As command 209
Save Frameset command 209
saving
  documents 72
  documents with a different name 257
  files in frames and framesets 209
  reports 114
  reverting to last saved version 257
  searches 429
  site maps 96
schemas 410, 506
screen readers
  JAWS for Windows 49
  Window Eyes 49
script links
  changing 329
  creating 326
scripts
  as assets. *See* assets
  balanced braces 434
  editing external 446
  editing in Design view 447
  inserting 446
  linking external files 447
  viewing functions 428
search pages
  ASP 641, 706
  ASP.NET 681
  ColdFusion 641
  JSP 641, 706
  PHP 641
searching
  code 428
  for files 301
  finding and replacing 301
  regular expressions 420
  saving search patterns 429
  tags and attributes 428
  text within files 301
security 666
security and permissions 490
Select Newer Remote command 104
selecting
  frames and framesets 207
  layers 154
  layout cells and tables 192
  multiple assets 121
  objects in the Document window 262
  table elements 167
  tags 446
Server Behavior Builder 590
server behaviors
  code for 423
  coding guidelines 596
  creating 590
  creating dialog boxes for 603
  Delete Record 714
  editing custom behaviors 605
  hiding regions 566
  inserting records 648
  installing more 598
  moving to records 565
  repeating regions 566
  testing 597
  Update Record 655, 692, 711, 731
server objects
  application objects 537
  ColdFusion variables 529, 539
  session objects 536
server scripts, in templates 223
server-side image maps 335
server-side includes
  editing 448
  inserting 448
  overview 419
servers
  access options 65
  connecting to edit files 82
  opening an existing connection 80
  setup troubleshooting 66
  uploading pages 616
session variables
  about 520
  data, retrieving 524
  data, storing 522
  defining 536
  form parameters and 522
  URL parameters and 522
Set as Home Page command 93
Set Color Scheme command 261
Set Nav Bar Image action 387
Set Text of Frame action 388

Set Text of Layer action 389
Set Text of Status Bar action 390
Set Text of Text Field action 390
setting
  code coloring 228
  default new document type 72
  document properties 258
  fonts and changing characteristics 286
shared settings files in Contribute sites 137
Shockwave movies
  as assets. *See* assets
  controlling 380
  inserting 365
  overview 365
Shockwave objects, making dynamic 551
shortcut menus. *See* context menus
Show Region behavior 566
Show-Hide Layers action 391
showing invisible elements 263
Site list (Assets panel)
  about 117
  refreshing 118
  viewing 117
site maps
  adding files to a site 93
  changing links in 328
  hiding files 94
  Link to New File command 93
  links 328
  modifying layout of 91
  removing links in 329
  saving as an image file 96
  setting home page for 90
  showing dependent files in 95
  showing files 94
  viewing 90
  viewing a branch of 95
  working with 89
  working with pages in 92
Site panel
  now Files panel 34, 86
  searching for text and/or HTML within documents 301
site root-relative paths. *See* root-relative paths
Site window, searching for text 301
sites
  assets, reusing 122
  broken links, fixing 338
  browser compatibility 434
  cache 327
  cache file 328
  Check In/Check Out, using 97
  cloaking 78, 104
  creating new 62
  creating new, Advanced settings 63
  creating new, Site Definition Wizard 62
  Design Notes, using with 108
  editing existing websites 67, 68
  editing site settings 67
  Files panel display, changing 84
  files, working with 87
  folder structure 60
  importing and exporting 96
  large, assets in 123
  links 315
  links, changing sitewide 330
  links, checking 336
  local folder, setting up 63
  local versus remote 59
  locating files in 88
  locating recently modified files 89
  opening for viewing 80
  previewing in browsers 270
  reference material 29
  remote folder, access options 65
  remote folder, setting up 65
  remote folder, setup troubleshooting 66
  removing from site list 97
  reports 112
  searching for files in 301
  security 666
  site map, working with 89
  synchronizing local and remote 103
  testing guidelines 111
  testing server 461
  viewing in Files panel 83
slider controls, creating 380
Snap to Web-Safe color palette 261
Snippets panel 424
snippets, adding keyboard shortcuts for 57
SOAP and web services 582
sound
  adding to a page 366
  embedding 366
  linking to an audio file 366
  playing 386
source code
  copying and pasting from Fireworks to Dreamweaver 347
  reference material 29

selecting in the Document window 262
tag styles 290
updating Fireworks HTML placed in Dreamweaver 348
See also code
source control 78, 97
spaces
converting to tabs 430
inserting non-breaking 290
special characters
inserting 289
line break 284
question marks in field names 493
in SQL account names 492
spell check, with Check Spelling command 301
spelling
checking 301
dictionaries 301
Split Cells command 178
Split Frame commands 204
SQL
about 753
ASP.NET 527
Database Items tree 533
defining columns 755
DELETE 753
filtering records 755, 758
FROM 754
GROUP BY 754
INSERT 753
joining tables 759
limiting records 755
operators 754
ORDER BY 754
recordset, defining with SQL 532
SELECT 753
SELECT statement 753
UPDATE 753
WHERE 754
SQL Server Connection 466
SQL Server, troubleshooting dynamic pages 492
stacking order, layers 155
Standard mode in Property inspector 549
Standard toolbar 34, 36
Start page 33
status bar
about 36
preferences 41
setting text (behavior) 390
Window Size pop-up menu 36
stored procedures
ASP 715
ASP.NET 700
ColdFusion 665
defined 632
JSP 715
modifying databases 665, 700, 714
Strikethrough (Default Color) button 261
style sheets
Edit Style Sheet dialog box 299
editing 298
external 296
overview 290
See also styles
Style submenu 286
styles
applying custom styles 295
conflicting styles 276
See also style sheets
submit buttons 615
Sun JDBC-ODBC Bridge driver 483
Sun One Active Server Pages 456
supported languages 416
Swap Image action 396
Swap Image Restore action 397
swatches, color 260
synchronizing local and remote sites 103
syntax errors in insert statements 493
System Color button 261
system color picker 261

T

table cells
cutting, copying, and pasting 179
highlighting preference 167
merging 165
splitting 165
See also layout cells, layout tables
table header menu
about 164
Clear All Heights 175, 192
Clear All Widths 175
displaying 176, 185
Make Widths Consistent 175, 197
Remove All Spacer Images 196
Select Table 168
tables
cells, clearing width and height 175
cells, highlighting 167, 184
cells, merging 178

cells, splitting 178
column widths 164
column widths, making consistent 197
columns, clearing widths 175
columns, making widths consistent 175, 197
content, adding to 165
creating 165
data, exporting 166
editing 170
elements, selecting 167
Expanded Tables mode 170
formatting 170
highlighting preference 167
importing 166
layers, converting from 160
nesting 181
overview 164
preset designs for 172
properties 171
resizing 173
restricting 506
rows and columns, adding and removing 176
rows and columns, adjusting 173
sorting 181
tabular data, importing 281
widths 164
widths, displaying 176
See also columns, rows, and cells
tabs, converting to spaces 430
tabular data, importing 166, 281
tag
editors 426
heading, applying 283
paragraph, applying 283
syntax, templates 220, 244
Tag Chooser, inserting tags 425
Tag inspector 442
Tag Library Editor 406
tag selector 34, 446
Tag Validator 437
tags
ASP.NET, importing 410
custom, importing 410
editing with a Tag editor 426
editing with the Quick Tag Editor 443
empty, removing 433
invalid 417
JRun, importing 411
JSP, importing 411
libraries 406
nested, combining 433
overlapping 417
removing 430, 446
searching for 428
selecting 262, 446
See also code
Target Browser Check panel 434
targeting frames 213
targeting links
in documents 320
opening document in a new window 321
template expressions 224
template parameters
modifying in template-based documents 249
overview 224
Template Properties dialog box 249
template-based documents
attaching templates 247
in Code view 219
in Design view 218
detaching templates 248
modifying editable tag attributes 249
modifying properties in 249
modifying repeating regions 250
modifying template parameters 249
templates
applying to existing document 247
as assets. *See* assets
Assets panel 228
changing region highlight colors 229
checking syntax 244
clicking in locked regions 248
in Code view 218
creating 226
creating for Contribute sites 229
creating new documents with 71
deleting 245
in Design view 217
detaching document from 248
editable regions 216, 232, 233, 234
editable tag attributes 217, 240
editing 242
editing code outside of HTML tags 223
editing server scripts in documents 223
expressions 224
finding editable regions 233
links 221
making region noneditable 233
modifying properties in template-based documents 249

nested 241
nesting 222
optional region 216, 237, 239
overview 216
parameters 224
preferences 228, 229
renaming 245
repeating region 216, 234, 250
repeating table 236
tag syntax 220
types of regions 216
undoing application 248
updating Contribute sites 244
updating documents 243
XML 246
temporary files, permissions for destroying 491
testing server
behaviors 597
setting up 454
web applications 461
testing your site
about 111
guidelines 111
using reports 112
text
adding to a document 280
alignment 283
changing color of 120, 288
changing font combinations 287
default color in pages 261
editors, files created by 73
editors. *See also* external editors
fields 608, 613
fields, setting text with behaviors 390
files, opening 73
formatting 273, 280, 285
importing from other documents 281
importing tabular data 281
indenting 283
inserting 273, 280
making dynamic 547
non-breaking space 290
outdenting 283
searching within documents 301
Text Color option 261
text-based HTML editors. *See* external editors
TextBox controls 625
TextTop alignment 308
titles, changing 259
tld files 411
toolbar, changing document titles 259
Top alignment 308
tracing images 161
transferring files 101
transferring files to and from Contribute sites 136
transparent pixels in background 260
troubleshooting
BOF 494
checking in and out of Contribute sites 142
clicking in locked regions 248
Contribute 140, 142
Contribute compatibility, enabling 135
COUNT field incorrect 493
data type mismatch 493
DSN 491
EOF 494
file in use 491
links, broken, in Contribute 142
locked files in Contribute sites 142
logon failed 492
Microsoft error messages 491
Netscape Navigator plug-ins 368
ODBC errors 494
parameters, too few 493
permissions 489
styles, not displaying correctly in Contribute 300
syntax errors 493
updateable queries 492
typographical conventions 29

U

UDDI
public directories 581
site list, editing 588
uncloaking
all folders and files 107
file types 106
site folders 105
underline 286
underscores in SQL account names 492
undocking panels and panel groups 46
undoing file check-out 100
Unmark Editable Region command 233
Update Current Page command 127, 243
Update HTML command 348
update pages
ASP 707
ASP.NET 686
ColdFusion 649

JSP 707
PHP 726
Update Pages command 243
Update Record behavior 655, 692, 711, 731
updating
links 327
records 649
templates 243
uploading
files 102
files to a server 616
URLs
applying to selection 120
as assets. *See* assets
creating URL assets 123
See also paths
usemap attribute 335
user interface, customizing panel layout 46
user names
checking during login 670
checking for uniqueness 668
letting users choose 667
storing 667

V

Validate Form action 397
validating forms 620
Validator
preferences 405
using 437
VBScript, inserting code in Design view 446
viewing
assets 117
code 401
files on a Dreamweaver site 80
files on a drive or desktop 81
files on a server 80
head section code 449
invisible elements 263
template regions in Code view 219
template regions in Design view 218
templates in Code view 218
templates in Design view 217
virtual path 475
virtual server-side includes 419
Visited Links color option 261
visual guides
overview 54
rulers 160
tracing images 161

W

web
design, levels of experience in 20
hosting services 475
photo albums, creating 349
server, setting up 454
web applications
creating a root folder 458
database connections 461
defining a Dreamweaver site 459
requirements 453
setting up a web server 454
setting up an application server 454
testing server 461
web services
about 579
adding to a page 586
AXIS proxy generator 582
proxy generators, additional 582
proxy generators, configuring 583
proxy generators, installing 582
SOAP 582
UDDI directories 581
UDDI site list, editing 588
workflow 580
web.xml file 411
Window Size pop-up menu 36
Windows OS color palette 261
word processors, files created by 73
word wrapping 403
workflow reports 112
workflow, for dynamic pages 507–513
working environments, Live Data window 500
workspace
about 32
customizing panel layout 46
floating layout 32
layout 55
wrapping lines in Code view 403
write permissions on servers 136
writing code 423
writing code blocks 598

X

XHTML
code 417
converting from HTML to XHTML 438
creating pages 438
XML
DTD files 410
in templates 246

Z

Z-Index option (for layers), changing stacking order
155