



Adobe® Dreamweaver® CS6

CLASSROOM IN A BOOK®

The official training workbook from Adobe Systems

DVD-ROM Included for Windows and Mac OS

Bonus 2 hours
of tutorials from
Adobe Dreamweaver CS6:
Learn by Video
also from Adobe Press



Adobe® Dreamweaver® CS6
CLASSROOM IN A BOOK
The official training workbook from Adobe Systems



Adobe® Dreamweaver® CS6 Classroom in a Book®

© 2012 Adobe Systems Incorporated and its licensors. All rights reserved.

If this guide is distributed with software that includes an end user license agreement, this guide, as well as the software described in it, is furnished under license and may be used or copied only in accordance with the terms of such license. Except as permitted by any such license, no part of this guide may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, recording, or otherwise, without the prior written permission of Adobe Systems Incorporated. Please note that the content in this guide is protected under copyright law even if it is not distributed with software that includes an end user license agreement.

The content of this guide is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Adobe Systems Incorporated. Adobe Systems Incorporated assumes no responsibility or liability for any errors or inaccuracies that may appear in the informational content contained in this guide.

Please remember that existing artwork or images that you may want to include in your project may be protected under copyright law. The unauthorized incorporation of such material into your new work could be a violation of the rights of the copyright owner. Please be sure to obtain any permission required from the copyright owner.

Any references to company names in sample files are for demonstration purposes only and are not intended to refer to any actual organization.

Adobe, the Adobe logo, Classroom in a Book, Dreamweaver, Fireworks, Flash, and Photoshop are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Apple, Mac OS, and Macintosh are trademarks of Apple, registered in the U.S. and other countries. Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation registered in the U.S. and/or other countries. All other trademarks are the property of their respective owners.

Adobe Systems Incorporated, 345 Park Avenue, San Jose, California 95110-2704, USA

Notice to U.S. Government End Users. The Software and Documentation are “Commercial Items,” as that term is defined at 48 C.F.R. §2.101, consisting of “Commercial Computer Software” and “Commercial Computer Software Documentation,” as such terms are used in 48 C.F.R. §12.212 or 48 C.F.R. §227.7202, as applicable. Consistent with 48 C.F.R. §12.212 or 48 C.F.R. §§227.7202-1 through 227.7202-4, as applicable, the Commercial Computer Software and Commercial Computer Software Documentation are being licensed to U.S. Government end users (a) only as Commercial Items and (b) with only those rights as are granted to all other end users pursuant to the terms and conditions herein. Unpublished-rights reserved under the copyright laws of the United States. Adobe Systems Incorporated, 345 Park Avenue, San Jose, CA 95110-2704, USA. For U.S. Government End Users, Adobe agrees to comply with all applicable equal opportunity laws including, if appropriate, the provisions of Executive Order 11246, as amended, Section 402 of the Vietnam Era Veterans Readjustment Assistance Act of 1974 (38 USC 4212), and Section 503 of the Rehabilitation Act of 1973, as amended, and the regulations at 41 CFR Parts 60-1 through 60-60, 60-250, and 60-741. The affirmative action clause and regulations contained in the preceding sentence shall be incorporated by reference.

Adobe Press books are published by Peachpit, a division of Pearson Education located in Berkeley, California. For the latest on Adobe Press books, go to www.adobeypress.com. To report errors, please send a note to errata@peachpit.com. For information on getting permission for reprints and excerpts, contact permissions@peachpit.com.

Writer: James J. Maivald
Project Editor: Nancy Peterson
Production Editor: Cory Borman
Development Editor: Robyn G. Thomas
Technical Editors: Clint Funk, Catherine Palmer
Copyeditor: Scout Festa
Compositor: Kim Scott, Bumpy Design
Indexer: Joy Dean Lee
Media Producer: Eric Geoffroy
Cover Design: Eddie Yuen
Interior Design: Mimi Heft

Printed and bound in the United States of America

ISBN-13: 978-0-321-82245-1

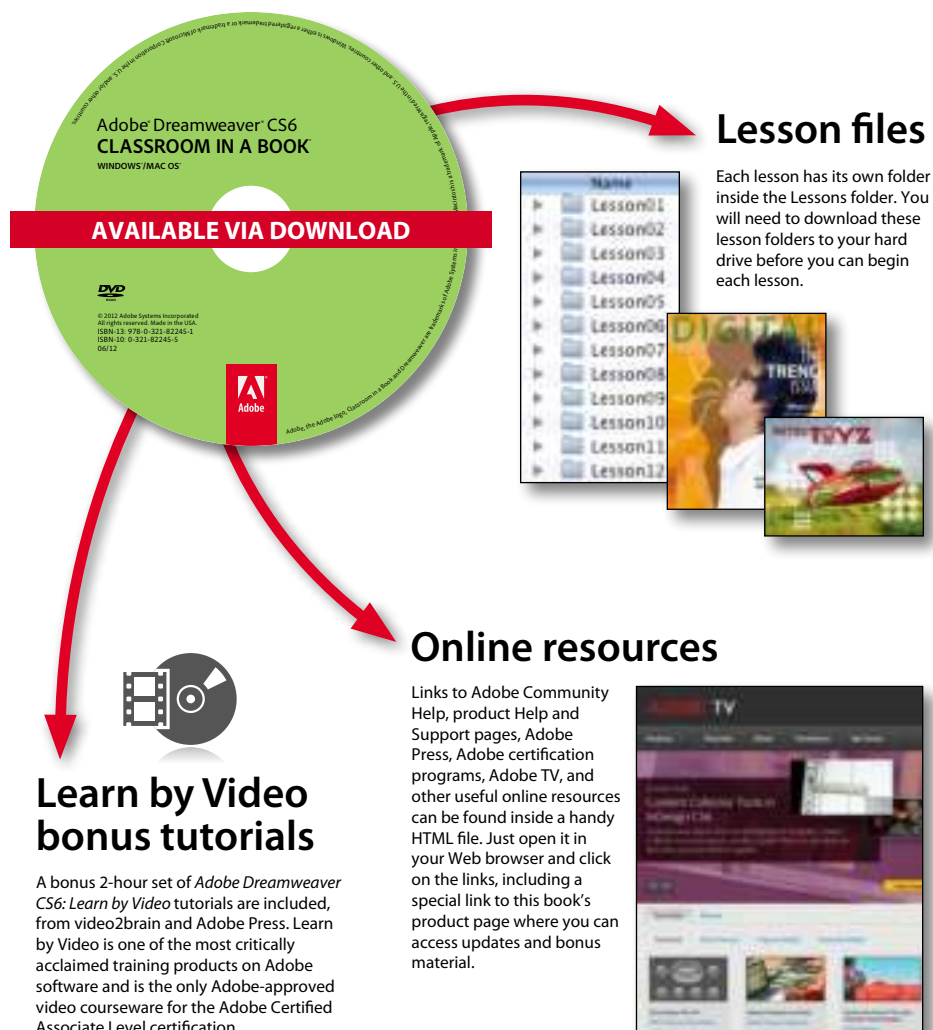
ISBN-10: 0-321-82245-5

9 8 7 6 5 4 3 2 1

EBOOK READERS: FIND YOUR LESSON FILES

Throughout this title you will see references to lesson or resource files on a disc. Please note that these files are available to eBook readers via high-speed download. Please [click here](#) to go to the last page in this eBook for the download location and instructions.

Adobe Dreamweaver CS6 Classroom in a Book includes the lesson files that you'll need to complete the exercises in this book, as well as other content to help you learn more about Adobe Dreamweaver CS6 and use it with greater efficiency and ease. The diagram below represents the contents of the lesson files directory, which should help you locate the files you need.



CONTENTS

GETTING STARTED ix

About <i>Classroom in a Book</i>	ix
TinyURLs	ix
Prerequisites	x
Installing the program	x
Copying the <i>Classroom in a Book</i> files	x
Recommended lesson order	xi
Jumpstart	xi
Setting up the workspace	xiii
Windows vs. Macintosh instructions	xiii
Finding Dreamweaver information	xiii
Checking for updates	xiv
Additional resources	xiv
Adobe certification	xv

1 CUSTOMIZING YOUR WORKSPACE 2






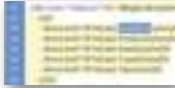


Lesson overview	2
Touring the workspace	4
Switching and splitting views	5
Working with panels	7
Selecting a workspace layout	11
Adjusting toolbars	13
Personalizing preferences	13
Creating custom keyboard shortcuts	14
Using the Property inspector	15

2 HTML BASICS 18



Lesson overview	18
What is HTML?	20
Where did HTML begin?	20

	Writing your own HTML code	22
	Frequently used HTML 4 codes	30
	Introducing HTML5	32
3	CSS BASICS	38
	Lesson overview	38
	What is CSS?	40
	HTML vs. CSS formatting	41
	HTML defaults	42
	CSS box model	45
	Formatting text	46
	Formatting objects	55
	Multiples, classes, and IDs, oh my!	64
	CSS3 overview and support	66
4	CREATING A PAGE LAYOUT	70
	Lesson overview	70
	Web design basics	72
	Working with thumbnails and wireframes	75
	Defining a Dreamweaver site	78
	Using the Welcome screen	79
	Previewing your completed file	80
	Modifying an existing CSS layout	81
	Adding a background image to the header	84
	Inserting new components	87
	Changing element alignment	92
	Modifying the page width and background color	93
	Modifying existing content and formatting	95
	Inserting an image placeholder	100
	Inserting placeholder text	101
	Modifying the footer	101
	Checking browser compatibility	102
5	WORKING WITH CASCADING STYLE SHEETS	104
	Lesson overview	104
	Previewing the completed file	106
	Working with the CSS Styles panel	106

	Working with type	110
	Using images for graphical effects.....	113
	Creating new CSS rules	115
	Creating an interactive menu	122
	Modifying hyperlink behavior	125
	Creating faux columns	128
	Moving rules to an external style sheet	130
	Creating style sheets for other media.....	132
6	WORKING WITH TEMPLATES	140
	Lesson overview.....	140
	Previewing completed files	142
	Creating a template from an existing layout	143
	Inserting editable regions.....	144
	Producing child pages	146
	Updating a template.....	149
	Using library items.....	152
	Using server-side includes	157
7	WORKING WITH TEXT, LISTS, AND TABLES	166
	Lesson overview.....	166
	Previewing the completed file.....	168
	Creating and styling text.....	170
	Creating and styling tables.....	182
	Spell checking webpages	195
	Finding and replacing text	197
8	WORKING WITH IMAGES	202
	Lesson overview.....	202
	Reviewing web image basics.....	204
	Previewing the completed file.....	209
	Inserting an image.....	210
	Adjusting image positions with CSS classes	212
	Working with the Insert panel.....	213
	Using Adobe Bridge to insert images	215
	Inserting non-web file types	218
	Working with Photoshop Smart Objects.....	221

Copying and pasting images from Photoshop and Fireworks	223
Inserting images by drag and drop	227
Optimizing images with the Property inspector.....	227

9 WORKING WITH NAVIGATION 232



Lesson overview	232
Hyperlink basics	234
Previewing the completed file.....	235
Creating internal hyperlinks.....	237
Creating an image-based link	240
Creating an external link	241
Setting up email links	243
Targeting page elements.....	244
Inserting Spry menu bars	248
Checking your page	265

10 ADDING INTERACTIVITY 268



Lesson overview	268
Learning about Dreamweaver behaviors	270
Previewing the completed file.....	272
Working with Dreamweaver behaviors	273
Working with Spry Accordion widgets.....	279

11 WORKING WITH WEB ANIMATION AND VIDEO 286






Lesson overview	286
Understanding web animation and video	288
Previewing the completed file.....	289
Adding web animation to a page.....	290
Adding web video to a page	293

12 WORKING WITH FORMS 300



Lesson overview	300
Previewing the completed file.....	302
Learning about forms	303
Adding a form to a page	305
Inserting text form elements	306
Inserting checkboxes	315

	Creating radio buttons	317
	Incorporating text areas	319
	Working with lists	320
	Adding a Submit button	322
	Specifying a form action	324
	Emailing form data	325
	Styling forms	329
13	WORKING WITH ONLINE DATA	334
	Lesson overview	334
	Working with dynamic content	336
	Previewing the completed file	336
	Using HTML and XML data	338
	Choosing a server model	348
	Configuring a local web server	349
	Setting up a testing server	350
	Building database applications	352
14	BUILDING DYNAMIC PAGES WITH DATA	360
	Lesson overview	360
	Building pages with PHP	362
	Creating a master/detail page set	372
	Creating a detail page	384
15	PUBLISHING TO THE WEB	388
	Lesson overview	388
	Defining a remote site	390
	Cloaking folders and files	396
	Wrapping things up	398
	Putting your site online	402
	Synchronizing local and remote sites	404
	APPENDIX: TinyURLs	408
	INDEX	411

GETTING STARTED

Adobe® Dreamweaver® CS6 is the industry-leading web authoring program. Whether you create websites for a living or plan to create one for your own business, Dreamweaver offers all the tools you need to get professional-quality results.

About *Classroom in a Book*

Adobe Dreamweaver CS6 Classroom in a Book® is part of the official training series for graphics and publishing software developed with the support of Adobe product experts.

The lessons are designed so that you can learn at your own pace. If you're new to Dreamweaver, you'll learn the fundamentals of putting the program to work. If you are an experienced user, you'll find that *Classroom in a Book* teaches many advanced features, including tips and techniques for using the latest version of Dreamweaver.

Although each lesson includes step-by-step instructions for creating a specific project, you'll have room for exploration and experimentation. You can follow the book from start to finish, or complete only those lessons that correspond to your interests and needs. Each lesson concludes with a Review section containing questions and answers on the subjects you've covered.

TinyURLs

In several points in the book, we reference external information available on the Internet. The uniform resource locators (URLs) for this information are often long and unwieldy, so we have provided custom TinyURLs in their place for your convenience. Unfortunately, the TinyURLs sometimes expire over time and no longer function. If you find that a TinyURL doesn't work, look up the actual URL provided in the Appendix.

Prerequisites

Before using *Adobe Dreamweaver CS6 Classroom in a Book*, you should have a working knowledge of your computer and its operating system. Be sure you know how to use the mouse, standard menus and commands, and also how to open, save, and close files. If you need to review these techniques, see the printed or online documentation that was included with your Microsoft Windows or Apple Macintosh operating system.

Installing the program

Before you perform any exercises in this book, verify that your computer system meets the hardware requirements for Dreamweaver CS6, that it's correctly configured, and that all required software is installed.

Adobe Dreamweaver CS6 software must be purchased separately; it is not included with the lesson files that accompany this book. For system requirements, go to www.adobe.com/products/dreamweaver/tech-specs.html.

Install Dreamweaver CS6 onto your hard drive from the Adobe Dreamweaver CS6 application DVD (you cannot run the program from the disc) or from the installation files you downloaded from Adobe. For complete instructions on installing the software, see the Adobe Dreamweaver CS6 Read Me file on the application DVD or on the web at www.adobe.com/support.

Make sure your serial number is accessible before installing the application.

Copying the *Classroom in a Book* files

The *Adobe Dreamweaver CS6 Classroom in a Book* DVD includes folders containing all the electronic files for the lessons in the book. Each lesson has its own folder; you must copy the folders to your hard drive to complete the lessons. It is recommended that you copy all lesson folders to your hard drive at once, but to conserve space on your hard disk, you can install individual folders for each lesson as you need them. It is vitally important that you store all lesson folders within a single folder on your hard drive. If you follow the recommended lesson order, this master folder will serve as the local site root folder, as described more fully in Lesson 4, "Creating a Page Layout."

To install the *Classroom in a Book* files:

- 1 Insert the *Adobe Dreamweaver CS6 Classroom in a Book* DVD into your computer's optical disc drive.

- 2 Navigate to the CD/DVD drive on your computer.
- 3 If you intend to complete all lessons in the book in order, drag the Lessons folder to your computer hard drive. Otherwise, skip to step 5.

The Lessons folder contains all the individual lesson folders and other assets needed for the training.

- 4 Rename the Lessons folder **DW-CS6**.

This folder will be used as the local site root folder.

- 5 If you desire to perform one or more lessons individually, copy each lesson folder to your hard drive separately, as needed. Then, proceed to the “Jumpstart” section for more instructions. Do not copy one lesson folder into any other lesson folder.

The files and folders for each lesson cannot be used interchangeably. For specific instructions, see the following section.

Recommended lesson order

The training herein is designed to take you from A to Z in basic to intermediate website design, development, and production. Each new lesson builds on previous exercises, using the files and assets you create to develop an entire website. It is recommended that you perform each lesson in sequential order to achieve a successful result and the most complete understanding of all aspects of web design.

The ideal training scenario will start in Lesson 1 and proceed through the entire book to Lesson 15. Since each lesson builds essential files and content for the next, once you start this scenario you shouldn't skip any lessons, or even individual exercises. While ideal, this method may not be a practicable scenario for every user. So, if desired, individual lessons can be accomplished using the jumpstart method described in the next section.

Jumpstart

For users who don't have the time or inclination to perform each lesson in the book in sequence, or who are having difficulty with a particular lesson, a jumpstart method is included to facilitate the performance of individual lessons in or out of sequence. Once you start using the jumpstart method, you will have to use this method for all subsequent lessons. For example, if you want to jumpstart Lesson 6, you will have to jumpstart Lesson 7, too. In many instances, essential files needed for subsequent exercises were built in earlier lessons and exercises and may not be present in a jumpstart environment.

Each lesson folder includes all the files and assets needed to complete the exercises contained within that lesson. Each folder contains finished files, staged files, and customized Template and Library files, but not always a complete set of files that may have been used or completed in other lessons. You may think these folders contain seemingly duplicative materials. But these duplicate files and assets, in most cases, cannot be used interchangeably in other lessons and exercises. Doing so will probably cause you to fail to achieve the goal of the exercise.

The jumpstart method for completing individual lessons treats each folder as a stand-alone website. To jumpstart a lesson, copy the lesson folder to your hard drive and create a new site for that lesson using the Site Setup dialog box. Do not define sites using subfolders of existing sites. Keep your jumpstart sites and assets in their original folders to avoid conflicts. One suggestion is to organize the lesson folders, as well as your own site folders, in a single *web* or *sites* master folder near the root of your hard drive. But avoid using the Dreamweaver application folder or any folders that contain a web server, like Apache, ColdFusion, or Internet Information Services (IIS) (which are described more fully in Lessons 13 and 14).

Feel free to use the jumpstart method for all lessons, if you prefer.

To set up a jumpstart site, do the following:

- 1 Choose Site > New Site.
The Site Setup dialog box appears.
- 2 In the Site Name field, enter the name of the lesson, such as **lesson06**.
- 3 Click the Browse (📁) icon next to the Local Site Folder field. Navigate to the desired lesson folder you copied from the *Adobe Dreamweaver CS6 Classroom in a Book DVD* and click Select/Choose.
- 4 Click the arrow (⌵) next to the Advanced Settings category to reveal the tabs listed there. Select the Local Info category.
- 5 Click the Browse icon next to the Default Images Folder field. When the dialog box opens, navigate to the Images folder contained within the lesson folder and click Select/Choose.
- 6 In the Site Setup dialog box, click Save.
- 7 The name of the currently active website will appear in the Files panel's site pop-up menu. If necessary, press F8/Cmd-Shift-F to display the Files panel, and select the desired website you wish to work on from the Show menu.

These steps will have to be repeated for each lesson you wish to jumpstart. For a more complete description of how to set up a site in Dreamweaver, see Lesson 4. Remember, if you use the jumpstart method for all lessons, you may not end up with a complete set of site files in any individual folder when you are finished.

Setting up the workspace

Dreamweaver includes a number of workspaces to accommodate various computer configurations and individual workflows. For this book the Designer workspace is recommended.

- 1 In Dreamweaver CS6, locate the Application bar. If necessary, choose Window > Application Bar to display it.
- 2 The default workspace is called Designer. If it is not displayed, use the pop-up menu in the Application bar to choose it.
- 3 If the default workspace has been modified and certain toolbars and panels are not visible (as they appear in the figures in the book), you can select Window > Workspace > Reset Designer to restore the default configuration.

Most of the figures in this book show the Designer workspace. When you finish the lessons in this book, experiment with various workspaces to find the one that you prefer.

For a more complete description of the Dreamweaver workspaces, see Lesson 1, “Customizing Your Workspace.”

Windows vs. Macintosh instructions

In most cases, Dreamweaver performs identically in both Windows and Mac OS X. Minor differences exist between the two versions, mostly due to platform-specific issues out of the control of the program. Most of these are simply differences in keyboard shortcuts, how dialog boxes are displayed, and how buttons are named. Screen shots may alternate between platforms throughout the book. Where specific commands differ, they are noted within the text. Windows commands are listed first, followed by the Macintosh equivalent, such as Ctrl-C/Command-C. Common abbreviations are used for all commands whenever possible, as follows:

WINDOWS	MACINTOSH
Control = Ctrl	Command = Cmd
Alternate = Alt	Option = Opt

Finding Dreamweaver information

For complete, up-to-date information about Dreamweaver panels, tools, and other application features, visit the Adobe website. Choose Help > Dreamweaver Help. The Adobe Help application opens and downloads the latest Help files from the

Adobe Community Help website. These files are cached locally so you can access them even when you are not connected to the Internet. You can also download a PDF version of the Dreamweaver Help files from the Adobe Help application.

For additional information resources, such as tips, techniques, and the latest product information, visit www.adobe.com/support/dreamweaver to access the Adobe Community Help page.

Checking for updates

Adobe periodically provides software updates. You can obtain these updates using Adobe Updater if you have an active Internet connection.

- 1 In Dreamweaver, choose Help > Updates. The Adobe Updater automatically checks for updates for your Adobe software.
- 2 In the Adobe Updater dialog box, select the updates you want to install, and then click Download And Install Updates to install them.

For book updates and bonus material, visit the book's page on the web at www.peachpit.com/dwcs6cib.

Additional resources

Adobe Dreamweaver CS6 Classroom in a Book is not meant to replace documentation that comes with the program or to be a comprehensive reference for every feature. Only the commands and options used in the lessons are explained in this book. For comprehensive information about program features and tutorials, please refer to these resources:

Adobe Community Help: Community Help brings together active Adobe product users, Adobe product team members, authors, and experts to give you the most useful, relevant, and up-to-date information about Adobe products.

To Access Community Help: To invoke Help, press F1 or choose Help > Dreamweaver Help.

Adobe content is updated based on community feedback and contributions. You can add comments to content and forums (including links to web content), publish your own content using Community Publishing, or contribute Cookbook recipes. Find out how to contribute at www.adobe.com/community/publishing/download.html.

See community.adobe.com/help/profile/faq.html for answers to frequently asked questions about Community Help.

Adobe Dreamweaver CS6 Help and Support: www.adobe.com/support/dreamweaver is where you can find and browse Help and Support content on adobe.com.

Adobe Forums: forums.adobe.com lets you tap into peer-to-peer discussions, questions, and answers on Adobe products.

Adobe TV: tv.adobe.com is an online video resource for expert instruction and inspiration about Adobe products, including a How To channel to get you started with your product.

Adobe Design Center: www.adobe.com/designcenter offers thoughtful articles on design and design issues, a gallery showcasing the work of top-notch designers, tutorials, and more.

Adobe Developer Connection: www.adobe.com/devnet is your source for technical articles, code samples, and how-to videos that cover Adobe developer products and technologies.

Resources for educators: www.adobe.com/education offers a treasure trove of information for instructors who teach classes on Adobe software. Find solutions for education at all levels, including free curricula that use an integrated approach to teaching Adobe software and can be used to prepare for the Adobe Certified Associate exams.

Also check out these useful links:

Adobe Marketplace & Exchange: www.adobe.com/cfusion/exchange is a central resource for finding tools, services, extensions, code samples, and more to supplement and extend your Adobe products.

Adobe Dreamweaver CS6 product home page: www.adobe.com/products/dreamweaver

Adobe Labs: labs.adobe.com gives you access to early builds of cutting-edge technology, as well as forums where you can interact both with the Adobe development teams building that technology and with other like-minded members of the community.

Adobe certification

The Adobe training and certification programs are designed to help Adobe customers improve and promote their product-proficiency skills. There are four levels of certification:

- Adobe Certified Associate (ACA)
- Adobe Certified Expert (ACE)

- Adobe Certified Instructor (ACI)
- Adobe Authorized Training Center (AATC)

The Adobe Certified Associate (ACA) credential certifies that individuals have the entry-level skills to plan, design, build, and maintain effective communications using different forms of digital media.

The Adobe Certified Expert program is a way for expert users to upgrade their credentials. You can use Adobe certification as a catalyst for getting a raise, finding a job, or promoting your expertise.

If you are an ACE-level instructor, the Adobe Certified Instructor program takes your skills to the next level and gives you access to a wide range of Adobe resources.

Adobe Authorized Training Centers offer instructor-led courses and training on Adobe products, employing only Adobe Certified Instructors. A directory of AATCs is available at partners.adobe.com.

For information on the Adobe Certified programs, visit www.adobe.com/support/certification/index.html.

This page intentionally left blank

4

CREATING A PAGE LAYOUT

Lesson Overview

In this lesson, you'll learn the following:

- The basics of webpage design
- How to create design thumbnails and wireframes
- How to insert and format new components into a predefined CSS layout
- How to use the Code Navigator to identify CSS formatting
- How to check for browser compatibility



This lesson should take 1 hour to complete. Before beginning, make sure you have copied the files for Lesson 4 to your hard drive as described in the “Getting Started” section at the beginning of the book. If you are starting from scratch in this lesson, use the method described in the “Jumpstart” section of “Getting Started.”



Whether you use thumbnails and wireframes or just a vivid imagination, Dreamweaver can quickly turn design concepts into complete, standards-based CSS layouts.

Web design basics

Before you begin any web design project for yourself or for a client, there are three important questions you need to answer:

- What is the purpose of the website?
- Who is the customer?
- How do they get here?

What is the purpose of the website?

Will the website sell or support a product or service? Is your site for entertainment or games? Will you provide information or news? Will you need a shopping cart or database? Do you need to accept credit card payments or electronic transfers? Knowing the purpose of the website tells you what type of content you'll be developing and working with and what types of technologies you'll need to incorporate.

Who is the customer?

Are the customers adults, children, seniors, professionals, hobbyists, men, women, everyone? Knowing *who* your market will be is vital to the overall design and functionality of your site. A site intended for children probably needs more animation, interactivity, and bright engaging colors. Adults will want serious content and in-depth analysis. Seniors may need larger type and other accessibility enhancements.

A good first step is to check out the competition. Is there an existing website performing the same service or selling the same product? Are they successful? You don't have to mimic others just because they're doing the same thing. Look at Google and Yahoo. They perform the same basic service, but their site designs couldn't be more different from one another.

Could two sites be more different than Google and Yahoo? Yet they both perform the same service.



How do they get here?

This sounds like an odd question when speaking of the Internet. But, just as with a brick-and-mortar business, your online customers can come to you in a variety of ways. For example, are they accessing your site on a desktop computer, laptop, tablet, or cell phone? Are they using high-speed Internet, wireless, or dial-up service? What browser do they most like to use, and what is the size and resolution of the display? These answers will tell you a lot about what kind of experience your customers will expect. Dial-up and cell phone users may not want to see a lot of graphics or video, while users with large flat-panel displays and high-speed connections may demand as much bang and sizzle as you can send at them.

So, where do you get this information? Some you'll have to get through painstaking research and demographic analysis. Some you'll get from educated guesses based on your own tastes and understanding of your market. But a lot of it is actually available on the Internet itself. W3Schools, for one, keeps track of tons of statistics regarding access and usage, all updated regularly:

- **w3schools.com/browsers/browsers_stats.asp**: Provides more information about browser statistics.
- **w3schools.com/browsers/browsers_os.asp**: Gives the breakdown on operating systems. In 2011, they started to track the usage of mobile devices on the Internet.
- **w3schools.com/browsers/browsers_display.asp**: Lets you find out the latest information on the resolutions, or size, of screens using the Internet.

If you are redesigning an existing site, your web hosting service itself may provide valuable statistics on historical traffic patterns and even the visitors themselves. If you host your own site, third-party tools are available, like Google Analytics and Adobe Omniture, which you can incorporate into your code to do the tracking for you for free or for a small fee.

When you boil down all the statistics, this is what you will find as of the beginning of 2012. Windows (80 to 90 percent) dominates the Internet, with most users divided almost equally between Firefox (37 percent) and Google Chrome (33 percent), with various versions of Internet Explorer (22 percent) taking third position. The vast majority of browsers are set to a resolution *higher* than 1024 pixels by 768 pixels. If it weren't for the growth in usage of cell phones and smartphones for accessing the Internet, these statistics would be great news for most web designers and developers. Designing a website that can look good and work effectively for both flat-panel displays and cell phones is a tall order.

Each day, more people are using cell phones and other mobile devices to access the Internet. Some users may use them now to access the Internet more frequently than they use desktop computers. This presents a nagging problem to web

designers. For one thing, cell phone screens are a fraction of the size of even the smallest flat-panel display. How do you cram a two- or three-column page design into a meager 200 to 300 pixels? Another problem is that many device manufacturers have decided to follow Apple's decision to drop support for Flash-based content on their mobile devices. Keep all these statistics in mind as you go through the process of designing your site.

Many of the concepts of print design are not applicable to the web, because you are not in control of the user's experience. A page carefully designed for a typical flat panel is basically useless on a cell phone.



Scenario

For the purposes of this book you will be working to develop a website for Meridien GreenStart, a fictitious community-based organization dedicated to green investment and action. This website will offer a variety of products and services and require a broad range of webpage types, including dynamic pages using server-based technologies like PHP.

Your customers come from a broad demographic including all ages and education levels. They are people who are concerned about environmental conditions and who are dedicated to conservation, recycling, and the reuse of natural and human resources.

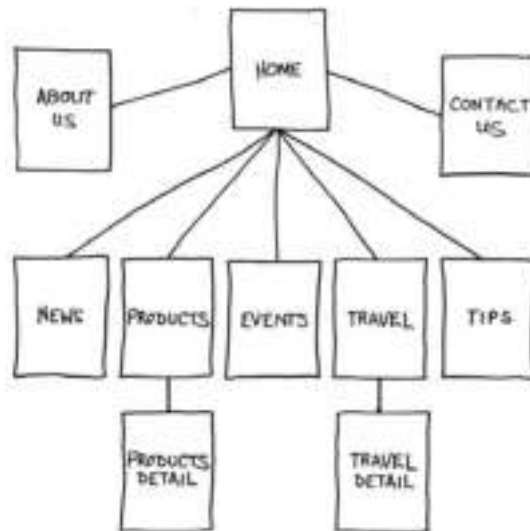
Your marketing research indicates that most of your customers use desktop computers or laptops, connecting via high-speed Internet services, but that you can expect 10 to 20 percent of your visitors via cell phone and other mobile devices.

Working with thumbnails and wireframes

The next step, after you have nailed down the answers to the three questions about your website purpose, customer demographic, and access model, is to determine how many pages you'll need, what those pages will do, and finally, what they will look like.

Creating thumbnails

Many web designers start by drawing thumbnails with pencil and paper. Think of thumbnails as a graphical shopping list of the pages you'll need to create for the website. Thumbnails can also help you work out the basic website navigation structure. Draw lines between the thumbnails showing how your navigation will connect them.



Thumbnails list the pages that need to be built and how they are connected to each other.

Most sites are divided into levels. Typically, the first level includes all the pages in your main navigation menu, the ones a visitor can reach directly from the home page. The second level includes pages you can reach only through specific actions or from specific locations, say from a shopping cart or product detail page.

Creating a page design

Once you've figured out what your site needs in terms of pages, products, and services, you can then turn to what those pages will look like. Make a list of components you want on each page, such as headers and footers, navigation, and areas for the main content and the sidebars (if any). Put aside any items that won't be needed on every page. What other factors do you need to consider?

Identifying the essential components for each page helps in creating an effective page design and structure that will meet your needs.

1. Header (includes banner and logo)
2. Footer (copyright info)
3. Horizontal navigation (for internal reference, i.e., Home, About Us, Contact Us)
4. Vertical navigation (links to products and services)
5. Main content (one-column with chance of two or more)

Do you have a company logo, business identity, graphic imagery, or color scheme you want to accent? Do you have publications, brochures, or current advertising campaigns you want to emulate? It helps to gather them all in one place so you can see everything all at once on a desk or conference table. If you're lucky, a theme will rise organically from this collage.

Once you've created your checklist of the components that you'll need on each page, sketch out several rough layouts that work for these components. Most designers settle on one basic page design that is a compromise between flexibility and sizzle. Some site designs may naturally lean toward using more than one basic layout. But resist the urge to design each page separately. Minimizing the number of page designs may sound like a major limitation, but it's key to producing a professional-looking site. It's the reason why some professionals, like doctors and airline pilots, wear uniforms. Using a consistent page design, or template, lends a sense of professionalism and gives confidence to your visitor.

Wireframes allow you to experiment with page designs quickly and easily without wasting time with code.



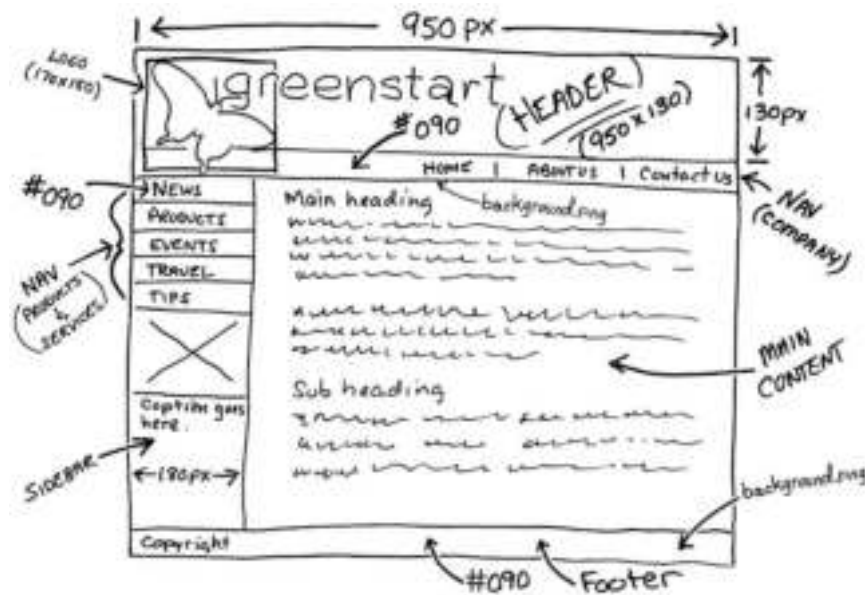
While you figure out what your pages will look like, you'll have to address the size and placement of the basic components. Where you put a component can drastically affect its impact and usefulness. In print, designers know that the upper-left corner of a layout is considered one of the "power positions," a place where you want to locate important aspects of a design, such as a logo or title. This is because in western culture we read from left to right, top to bottom. The second power position is the lower-right corner because this is the last thing your eyes will see when you're finished reading.

Unfortunately, in web design this theory doesn't work so well because of one simple reason: You can never be certain how the user is seeing your design. Are they on a 20-inch flat panel or a 2-inch cell phone?

In most instances, the only thing you can be certain of is that the user can see the upper-left corner of any page. Do you want to waste this position by slapping the company logo here? Or, make the site more useful by slipping in a navigation menu? This is one of the key predicaments of the web designer. Do you go for design sizzle, workable utility, or something in between?

Creating wireframes

After you pick the winning design, wireframing is a fast way to work out the structure of each page in the site. A wireframe is like a thumbnail, but bigger, that sketches out each page and fills in more details about the components, such as actual link names and main headings. This step helps to catch or anticipate problems before you smack into them when working in the code.



The wireframe for the final design should identify the components and feature markup for content, color, and dimensions.

Once the basic concepts are worked out, many designers take an extra step and create a full-size mockup or “proof of concept” using a program like Adobe Fireworks, Photoshop, or even Illustrator. It’s a handy thing to do because you’ll find that some clients just aren’t comfortable giving an approval based only on pencil sketches. The advantage here is that all these programs allow you to export the results to full-size images (JPEG, GIF, or PNG) that can be viewed in a browser. Such mockups are as good as seeing the real thing but may take only a fraction of the time to produce.

In some cases, creating a mockup in Photoshop, Fireworks, or Illustrator can save hours of tedious coding to receive a needed approval.

► **Tip:** For years, designers have started the design process in Fireworks, where they can create a fully functional mockup (with menus, links, and hotspots) that can then be exported to a CSS-based HTML layout and then edited in Dreamweaver.



Defining a Dreamweaver site


From this point forward, the lessons in this book function within a Dreamweaver site. You will create webpages from scratch and use existing files and resources that are stored on your hard drive, which combined make up what's called your *local* site. When you are ready to upload your site to the Internet (see Lesson 16, “Publishing to the Web”), you publish your completed files to a web host server, which then becomes your *remote* site. The folder structures and files of the local and remote sites are usually mirrors of each other.

First, let's set up your local site:

- 1 Launch Adobe Dreamweaver CS6, if necessary.
- 2 Choose Site > New Site. The Site Setup dialog box appears.

If you've used any previous version of Dreamweaver, you will notice that the Site Setup dialog box has been slightly redesigned. Along with the options for creating a standard Dreamweaver site, the dialog offers the ability to create a site based on the services offered by Adobe Business Catalyst. Business Catalyst is an online, hosted application that allows you to build and manage rich, dynamic web-based businesses. To learn more about the capabilities of Business Catalyst, check out www.BusinessCatalyst.com.


To create a standard website in Dreamweaver CS6, you need only name it and select the local site folder. Site names typically relate to a specific project or client and will appear in the Files panel. This name is intended for your own purposes, so there are no limitations to the name you can choose. It's a good idea to use a name that clearly describes the purpose of the website.

- 3 In the Site Name field, type **DW-CS6**.
- 4 Next to the Local Site Folder field, click the folder () icon. When the Choose Root Folder dialog box opens, navigate to the DW-CS6 folder containing the lesson files you copied from the *Adobe Dreamweaver CS6 Classroom in a Book* CD and click Select/Choose.

You could click Save at this time and begin working on your new website, but we'll add one more piece of handy information.

- 5 Click the arrow (▣) next to the Advanced Settings category to reveal the tabs listed there. Select the Local Info category.

Although it's not required, a good policy for site management is to store different file types in separate folders. For example, many websites provide individual folders for images, PDFs, video, and so on. Dreamweaver assists in this endeavor by including an option for a Default Images folder. Later, as you insert images from other places on your computer, Dreamweaver will use this setting to automatically move the images into the site structure.

- 6 Next to the Default Images Folder field, click the folder () icon. When the dialog box opens, navigate to the DW-CS6/images folder containing the files you copied from the *Adobe Dreamweaver CS6 Classroom in a Book* CD and click Select/Choose.

You've entered all the information required to begin your new site. In subsequent lessons, you'll add more information to enable you to upload files to your remote site and test dynamic webpages.

- 7 In the Site Setup dialog box, click Save.

The site name DW-CS6 now appears in the site list pop-up menu in the Files panel.

Setting up a site is a crucial first step in beginning any project in Dreamweaver. Knowing where the site root folder is located helps Dreamweaver determine link pathways and enables many site-wide options, such as Find and Replace.

Using the Welcome screen

The Dreamweaver Welcome screen provides quick access to recent pages, easy creation of a range of page types, and a direct connection to several key Help topics. The Welcome screen appears when you first start the program or when no other documents are open. Let's use the Welcome screen to explore ways to create and open documents.

- 1 In the Create New column of the Welcome screen, click HTML to create a new, blank HTML page instantly.

- 2 Choose File > Close.

The Welcome screen reappears.

- 3 In the Open A Recent Item section of the Welcome screen, click the Open button. This allows you to browse for files to open in Dreamweaver. Click Cancel.

The Welcome screen shows you a list of up to nine of your recently opened files; however, your installation may not display any used files at this point. Choosing a file from this list is a quick alternative to choosing File > Open when you want to edit an existing page you have recently opened or created.

You will use the Welcome screen several times in this book. When you've completed the lessons in this book, you may prefer not to use the Welcome screen or even see it. If so, you can disable it by selecting the Don't Show Again option in the lower left of the window. You can re-enable the Welcome screen in the General category of the Dreamweaver Preferences panel.

Previewing your completed file

To understand the layout you will work on in this lesson, preview the completed page in Dreamweaver.

- 1 In Dreamweaver CS6, press F8/Shift-Command-F to open the Files panel, and select DW-CS6 from the site list.
- 2 In the Files panel, expand the Lesson04 folder.
- 3 Double-click **layout_finished.html** to open it.



This page represents the completed layout you will create in this lesson. It is based on the wireframe drawings made earlier in this lesson and uses one of the new HTML5 CSS layouts available in Dreamweaver. Take a few moments to familiarize yourself with the design and components on the page. Can you determine what makes this layout different from existing HTML 4-based designs? You will learn the differences as you work through this lesson.

- 4 Choose File > Close.

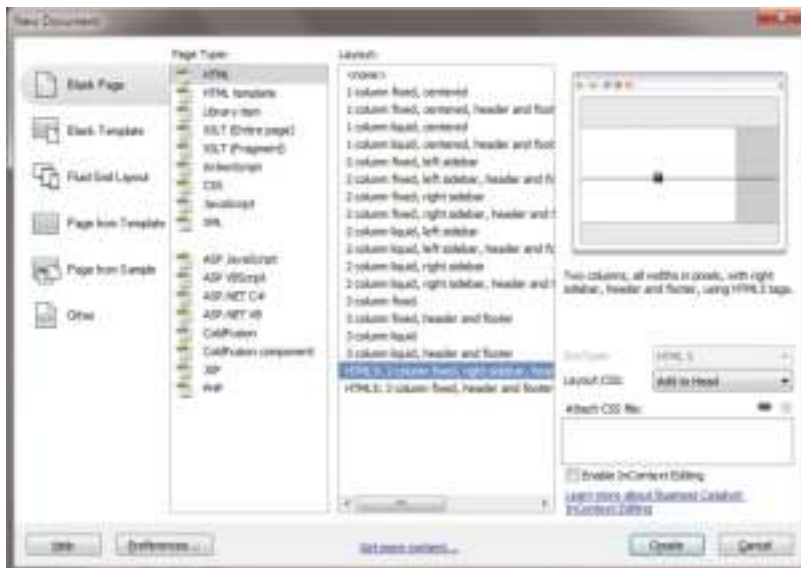
Modifying an existing CSS layout

The predefined CSS layouts provided by Dreamweaver are always a good starting point. They are easy to modify and adaptable to most projects. Using a Dreamweaver CSS layout, you will create a proof-of-concept page to match the final wireframe design. This page will then be used to create the main project template in subsequent lessons. Let's find the layout that best matches the wireframe.

- 1 Choose File > New.
- 2 Choose Blank Page > HTML in the New Page dialog box.

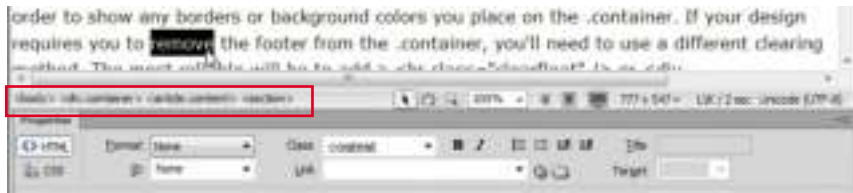
Dreamweaver CS6 offers 16 standard HTML 4 layouts and two HTML5 CSS layouts. Although the HTML5 layouts use some of the new semantic content elements, they are nearly identical to the HTML 4 designs. Unless you need to support an installed base of older browsers (like IE5 & 6), there's little to worry about using the newer layouts. Let's choose one of the HTML5 layouts that best fits the needs of the new site.

Note: If for some reason you can't or don't want to use an HTML5-based layout, see the sidebar "Alternate HTML 4 workflow" later in this lesson.



The layout “HTML5: 2 column fixed, right sidebar, header and footer” has the most in common with the target design. The only difference is that the sidebar element is aligned to the right of the layout instead of to the left. You will align this element to the left later in this lesson.

- 3 Select **HTML5: 2 column fixed, right sidebar, header and footer** from the layout list. Click Open/Create.
- 4 Switch to Design view, if necessary.
- 5 Insert the cursor anywhere in the page content. Observe the names and order of the tag selectors at the bottom of the document window.



The display order of elements in the tag selector directly correlates to the page’s code structure. Elements appearing to the left are parents, or containers, of all elements to the right. The element farthest to the left is the highest in the page structure. As you can see, the `<body>` element is highest and `<div.container>` is second.

As you click around the page sections, you will be able to determine the HTML structure without having to delve into the Code view window at all. In many ways, the tag selector interface makes the job of identifying the HTML skeleton much easier, especially in complex page designs.

Semantics is all in the name

You will see several elements that you may not be familiar with yet, such as `<section>`, `<article>`, `<aside>`, and `<nav>`. These are some of the new *semantic* elements being introduced in HTML5. In the past, you would have seen `<div>` elements identified and differentiated with `class` or `id` attributes, such as `<div class="header">` or `<div id="nav">`, to make it possible to apply CSS styling. HTML5 has simplified this construction down to `<header>` and `<nav>`. By using elements that are named for specific tasks or types of content, you can streamline code construction while achieving other benefits as well. For example, as search engines, such as Google and Yahoo, are optimized for HTML5, they will be able to locate and identify specific types of content on each page more quickly, making your site more useful and easier to browse.

The page consists of four main content elements, three subsections, and a single element that wraps around all the others. All but one of these are new HTML5 elements, including `<header>`, `<footer>`, `<nav>`, `<aside>`, `<article>`, and `<section>`. The only `<div>` elements in this layout are being used to hold the sidebar content and to hold everything together. Using these new elements means that you can apply complex CSS styling while reducing the complexity of the code overall. You can still use `class` and `id` attributes, but the new semantic elements reduce the need for this technique.

To understand exactly how much this design depends on CSS, it's sometimes a good idea to shut off CSS styling.

- 6 Choose View > Style Rendering > Display Styles to disable CSS styling in Design view.

Style display is typically on by default (showing a check mark in the menu). By clicking this option in the menu, you'll toggle CSS styling off temporarily.



- 7 Note the identity and order of each page component.

Without CSS, the HTML skeleton is exposed for all to see. It's instructive to know what the page will look like if somehow the cascading style sheet is disabled or not supported by a particular browser. Now it's easier to identify the page components and their structure. Although it is not strictly required, items that display higher on the page, like `<header>`, usually are inserted before other elements that appear lower, like `<footer>`.

Another important aspect you should notice is the navigation menu. Without the CSS styling, the navigation menu reverted back to a simple bulleted, or unordered, list with hyperlinks. Not too long ago this menu would have been built with images and complex rollover animation. If the images failed to load, the menu usually became a jumbled, unusable mess. The hyperlinks continued to work, but without the images there were no words to tell the user what they were clicking. But navigation built on text-based lists, on the other hand, will always be usable, even without styling.

- 8 Choose View > Style Rendering > Display Styles to turn on CSS styling again.

It's a good idea to get into the habit of saving files before you modify any settings or add content. Dreamweaver doesn't offer a backup or recovered-file feature; if it crashes before you save, all your work in any open, unsaved file will be lost. Get into the habit of saving your files on a regular basis. It will prevent the loss of data and important changes to your files.

- 9 Choose File > Save. In the Save As dialog box, navigate to the site root folder, if necessary. Name the file **mylayout.html** and click Save.

Dreamweaver normally saves HTML files to the default folder specified in the site definition, but it's a good idea to double-check the destination to make sure your files end up in the right place. All HTML pages created for the final site will be saved in the site root folder.

Adding a background image to the header

CSS styles are the current standard for all web styling and layout. In the following exercises, you'll apply background colors and a background image to a page section, adjust element alignment and the page width, and modify several text properties. All these changes are accomplished using Dreamweaver's CSS Styles panel.

If you start at the top of the page and work down, the first step would be to insert the graphical banner that appears in the final design. You could insert the banner directly into the header, but adding it as a background image has the advantage of leaving that element open for other content. It will also allow the design to be more adaptable to other applications, like cell phones and mobile devices.

- 1 Select the image placeholder *Insert_logo (180x90)* in the header. Press Delete.

When you delete the image placeholder, the empty header will collapse to a fraction of its former size because it has no CSS height specification.

- 2 Choose Window > CSS Styles to display the CSS Styles panel, if necessary.
- 3 In the CSS Styles panel, double-click the header rule to edit it.

The "CSS Rule Definition for header" dialog box appears.

Alternate HTML 4 workflow

HTML5 is coming on strong all over the Internet and for most applications the suggested workflow will work perfectly well. But HTML5 is not the current web standard and some pages or components may not display properly on certain older browsers and devices. If you'd rather work with code and structures that are more tried and true, feel free to substitute one of the HTML 4-based layouts still offered by Dreamweaver. We suggest the layout "2-Column fixed, left sidebar, header and footer."

However, if you use this layout you'll have to adapt the steps within all the following lessons and exercises to this layout. For example, HTML 4 doesn't use the new semantic elements, such as the following:

```
<header>...</header>
<footer>...</footer>
<section>...</section>
<article>...</article>
<nav>...</nav>
```

Instead, substitute a generic `<div>` element and use a `class` attribute that identifies the component:

```
<div class="header">...</div>
<div class="footer">...</div>
<div class="section">...</div>
<div class="article">...</div>
<div class="nav">...</div>
```

You'll also have to adapt the CSS styling for the HTML 4 elements. In most cases, Dreamweaver will supply many of the rules you'll need with the layout or will build the appropriate selector name for you automatically by using the class name in place of the HTML5 component (header, footer, nav, and so on).

That way, the CSS rule `header {color:#090}` becomes instead `.header {color: #090}`.

With all the caveats out of the way, the ugly truth is that even when you use standard HTML 4 code and components, older browsers and certain devices will still fail to render them properly. Some web designers believe that the longer we persist in using the older code, the longer the older software and devices will hang around making our lives difficult and delaying the inevitable adoption of HTML5. These designers say we should abandon the older standards and force users to upgrade as soon as possible.

The final decision is yours or your company's to make. In most cases, the problems you experience with HTML5 will be minor flaws—a font that's too big or too small—not a complete meltdown.

For more information about the differences between HTML 4 and HTML5, check out the following links:

<http://tinyurl.com/html-differences>

<http://tinyurl.com/html-differences-1>

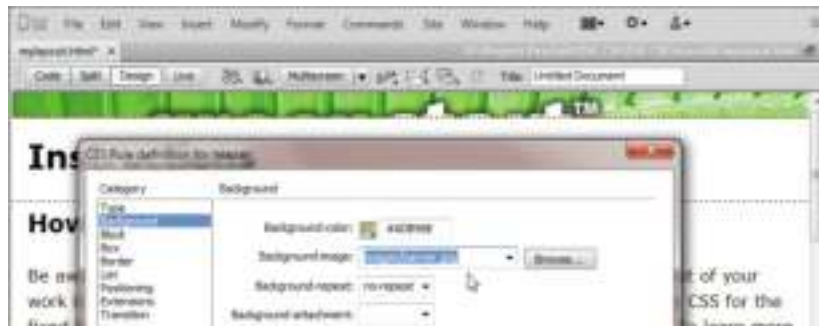
<http://tinyurl.com/html-differences-2>

- 4 In the Background category of the “CSS Rule Definition for *header*” dialog box, click the Browse button next to the Background-image field.
- 5 Select **banner.jpg** and note the dimensions of the image in the preview.



The image is 950 by 130 pixels.

- 6 Click OK/Choose to select the background image.
Background images repeat both vertically and horizontally by default. This isn't a problem at the moment, but to ensure that this setting doesn't cause any undesirable effects in the future, you'll need to change the repeat behavior.
- 7 Choose no-repeat from the Background-repeat menu.



- 8 Click Apply to view the results.
The header is wide enough but not tall enough to display the entire background image. Since background images aren't truly inserted into an element, they have no effect (positive or negative) on the size of a container. To ensure that the `<div>` is large enough to display the entire image, you need to add a height specification to the header rule.

- 9 In the Box category, type **130** in the Height field and choose px from the pop-up menu to the right of the Height field. Click Apply.



The background appears fully in the `<header>` element. Note that the image is slightly narrower than the container. We'll adjust the width of the layout later. We don't want to set the width on the `<header>` element itself. You learned in Lesson 3 that the width of block elements, like `<header>`, defaults to the entire width of their parent element. Before clicking OK, we need to add some finishing touches to the element.

The `<header>` element contains a background color that doesn't match your site color scheme. Let's apply one that does.

- 10 In the Background category, type **#090** in the Background-color field. Click OK.
You won't see the background color unless the background image fails to load.
- 11 Choose File > Save.

Inserting new components

The wireframe design shows two new elements that don't exist in the current layout. The first contains the butterfly image, the second the horizontal navigation bar. Did you notice that the butterfly actually overlaps both the header and the horizontal navigation bar? There are several ways to achieve this effect. In this case, an absolutely positioned (AP) `<div>` will work nicely.

- 1 Insert the cursor into the header, if necessary. Select the `<header>` tag selector. Press the Left Arrow key.

This procedure should insert the cursor before the opening `<header>` tag. If you had pressed the Right Arrow key, the cursor would move outside the closing `</header>` tag instead. Remember this technique—you'll use it

Note: To better understand how this technique works, try this step in Split view.

● **Note:** AP divs were used extensively in the past to create highly structured, fixed-layout web designs. This technique has declined dramatically in recent years as the need to support cell phones and other mobile devices has increased. For certain applications, AP divs are still handy.

frequently in Dreamweaver when you want to insert the cursor in a specific location before or after a code element without resorting to Code view.

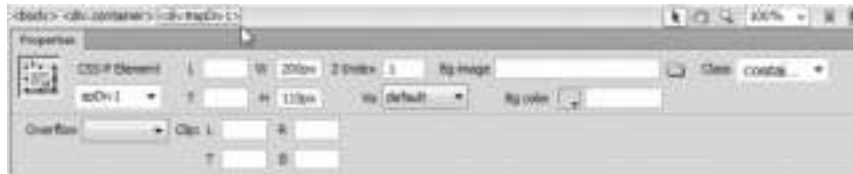
- 2 Choose Insert > Layout Objects > AP Div.



An AP div will appear at the top left of the header. Note the ID (`#apDiv1`) assigned to the new div in the tag selector. A corresponding rule has been added to the CSS Styles panel.

In previous versions of HTML, an AP div would have been assigned its size and position using inline HTML attributes. In a concession to new CSS-based web standards, these specifications are now applied by CSS via a unique ID created by Dreamweaver at the moment you insert the element.

- 3 Click the `<div#apDiv1>` tag selector.



The Property inspector displays the specifications for `<div#apDiv1>`. Note the element's width and height. Another property to be aware of is the z-index. This setting determines whether the element displays above or below another object. By default, all elements have a z-index of zero (0). On the other hand, the AP div has a z-index of 1, which means it will appear above all the other elements on the page. All the values displayed in the Property inspector are actually stored in the `#apDiv1` rule that was generated automatically by Dreamweaver.

- 4 Insert the cursor into `<div#apDiv1>`.
- 5 Choose Insert > Image. Select **butterfly-ovr.png** from the images folder. Observe the dimensions of the image: 170 pixels by 158 pixels.

- 6 Click OK/Choose.

The Image Tag Accessibility Attributes dialog box appears.

- 7 Type **GreenStart Logo** in the Alternate text field in the Image Tag Accessibility Attributes dialog box. Click OK.

The butterfly appears in the AP `div`.

Testing special image effects

You may have noticed that the butterfly displays a partially translucent drop shadow. The PNG file type supports *alpha* transparency, making this 3D effect possible. Unfortunately, older browsers may not support alpha transparency and other PNG features. If you use PNG files with such effects, always test them in your target browsers to make sure the desired effects are supported. If not, remove the effect or replace the file with a similar image saved as a different file type (GIF or JPEG).

You'll notice that the AP `div` is slightly wider than the butterfly image. Although the extra space shouldn't cause any trouble, it's a good idea to match the dimensions of the container to the image.

- 8 Double-click the `#apDiv1` rule in the CSS Styles panel.

The "CSS Rule definition for *#apDiv1*" dialog box appears.

- 9 In the Box category, change the width to **170 px**. Change the height to **158 px**.

The `<div>` dimensions now match the height and width of the image.

- 10 Deselect the Same For All check boxes for Margins.

- 11 Type **10 px** in the Top margin field. Type **30 px** in the Left margin field. Click OK.

When the rule definition dialog box vanishes, `<div#apDiv1>` appears floating over the header, 10 pixels from the top and 30 pixels to the left of its original position.

An AP `div` acts like a free agent. It ignores the other page components and can even be positioned above or below other `<div>` elements and content.

The `<div#apDiv1>` is complete. Now, let's add another new component that will hold the horizontal navigation shown in the site design specs. The vertical navigation menu will hold links to the organization's products and services. The horizontal navigation will be used to link back to the organization's home page, mission statement, and contact information.

In HTML 4, you probably would have inserted the links into another `<div>` element and used a `class` or `id` attribute to differentiate it from the other `<div>` elements in the file. Instead, HTML5 provides a new element geared specifically toward such components: `<nav>`.

- 12 Insert the cursor back into the header. Click the `<header>` tag selector. Press the Right Arrow key.

The cursor should now appear after the ending `</header>` tag.

- 13 Press `Ctrl-T/Cmd-T` to activate the Tag Editor.



The Tag Editor appears, allowing you to add HTML components or edit existing tags without having to switch to Code view.

- 14 Type `nav`. Press `Return/Enter` as necessary to complete the tag.

An empty `<nav>` element has been inserted into the code, and the cursor has been inserted within this element—ready for you to enter any content.

- 15 In the Property inspector, select Paragraph from the Format pop-up menu. Type **Home | About Us | Contact Us** in the new `<p>` element as placeholders for navigation links.



You will convert these to actual hyperlinks in Lesson 9, “Working with Navigation.” For now, let’s create a new CSS rule to format this element.

- 16 Select the `<p>` tag selector at the bottom of the document window.
- 17 In the CSS Styles panel, click the New CSS Rule button.



● **Note:** The `<nav>` element is new in HTML5. If you need to use HTML 4 code and structures, see the sidebar “Alternate HTML 4 workflow” earlier in this lesson.

- 18 If necessary, select Compound from the Selector Type menu.
The Selector Name window should display `.container nav p`.
- 19 Click the Less Specific button to remove the notation `.container` from the selector name.
The Selector Name window should display `nav p`.
- 20 Click OK to create the rule.
The “CSS Rule Definition for *nav p*” dialog box should appear.
- 21 In the Type category, type **90** in the Font-size field and select the percentage sign (%) from the pop-up list. Type **#FFC** in the Color field. Select Bold from the Font-weight field.
- 22 In the Background category, type **#090** in the Background-color field.
- 23 In the Block category, select Right from the Text-align field.
- 24 In the Box category, deselect the Same For All check box for Padding. Enter **5 px** in the Top padding field. Enter **20 px** in the Right padding field. Enter **5 px** in the Bottom padding field.
- 25 In the Border category, deselect the Same For All check boxes for Style, Width, and Color. Enter the following values only in the corresponding Bottom border fields: **solid, 2 px, #060**.
- 26 Click OK in the CSS Rule Definition dialog box. Press Ctrl-S/Cmd-S to save the file.


► **Tip:** To enter separate values in the Bottom field, remember to deselect the Same For All check boxes in each section first.



The new `<nav>` element appears below the header fully formatted and filled with your placeholder text aligned to the right.

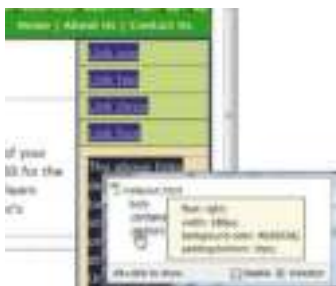
Changing element alignment

The proposed design calls for the sidebar to appear on the left side of the layout, but this layout puts it on the right. Dreamweaver does offer HTML 4 layouts that match the design criteria much more closely, but it was hard to resist working with an HTML5 layout and all the new semantic elements. Besides, adjusting the layout is a lot easier than you may think.

- 1 In Design view, insert the cursor anywhere in the right sidebar.
- 2 Click the Code Navigator () icon, or right-click in the sidebar and select the Code Navigator option from the context menu.

The Code Navigator window opens, displaying all CSS rule names that may affect the selected element.

- 3 Identify the rule that aligns the sidebar to the right.



The obvious culprit is the `.sidebar1` rule, which contains the declaration `float: right`.

- 4 In the CSS Styles panel, double-click the `.sidebar1` rule to edit it.
- 5 In the Box category, change the `float` property from `right` to `left`.
- 6 Click Apply to preview the change.



The sidebar moves to the left side of the layout.

- 7 Click OK and save the file.

Modifying the page width and background color

Before you convert this file into the project template, let's tighten up the formatting and the placeholder content. For example, the overall width has to be modified to match the banner image. Before you can adjust the width, you'll first have to identify the CSS rule that controls the page width.

- 1 If necessary, select View > Rulers > Show or press Alt-Ctrl-R/Opt-Command-R to display the rulers in the Design window.

You can use the rulers to measure the width and height of HTML elements or images. The orientation of the rulers defaults to the upper-left corner of the Design window. To give you more flexibility, you can set this zero point anywhere in the Design window.

- 2 Position the cursor over the axis point of the horizontal and vertical rulers. Drag the crosshairs to the upper-left corner of the header element in the current layout. Note the width of the layout.



Using the ruler, you can see that the layout is between 960 and 970 pixels wide.

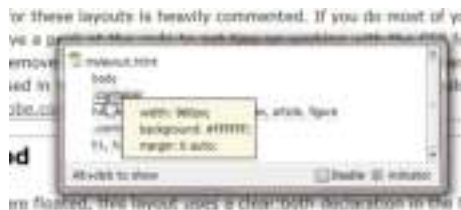
- 3 Insert the cursor into any content area of the layout.

Observe the tag selector display to locate any elements that may control the width of the entire page; it would have to be an element that contains all the other elements. The only elements that fit this criterion are `<body>` and `<div.container>`.

- 4 Click the Code Navigator (🔍) icon, or right-click in the sidebar and select Code Navigator from the context menu.

The Code Navigator window opens, displaying all CSS rule names that may affect the selected element.

- 5 Move the cursor over the `body` and `.container` rules to locate the specification that may control the page width.



Can you identify the rule that controls the width of the entire page? The `.container` rule contains the declaration `width: 960px`. By now you should be getting good at CSS forensics using the tag selector interface and the Code Navigator.

- 6 Double-click the `.container` rule in the CSS Styles panel.
- 7 In the Box category, change the width to **950 px**. Click OK.



The `<div.container>` element now matches the width of the banner image, but you may have experienced an unintended consequence when you changed the overall width. In our example, the main content area shifted down below the sidebar. To understand what happened, you'll have to do a quick investigation.

- 8 In the CSS Styles panel, click the `.content` rule and check its properties. Note its width: 780 pixels.
- 9 Click the `.sidebar1` rule and check its width: 180 pixels.

Combined, the two `<div>` elements total 960 pixels, the same as the original layout width. The elements are too wide to sit side by side in the main container and thereby prompted the unexpected shift. This type of error is common in web design and is easily fixed by adjusting the width of either of the two child elements.

- 10 Click the `.content` rule in the CSS Styles panel. In the Properties section of the panel, change the width to **770 px**.



The `<div.content>` element returns to its intended position. This was a good reminder that the size, placement, and specifications of page elements have important interactions that can affect the final design and display of your elements and of the entire page.

The current background color of the page detracts from the overall design. Let's remove it.

- 11 Double-click the `body` rule. In the Background category, change the Background-color to `#FFF`. Click OK.

Note that the absence of the background color gives the impression that the page's content area drifts off into the wide expanse. You could give `<div.container>` a different background color, or you could simply add a border to give the content elements a definitive edge. For this application, a thin border makes the most sense.

- 12 Double-click the `.container` rule. In the Border category, if necessary select the Same For All check boxes and enter the following values for all border fields: **solid**, **2px**, and **#060**. Click OK.



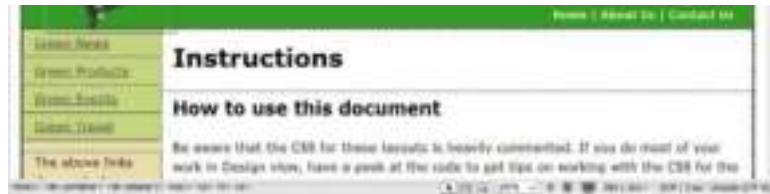
A dark green border appears around the page edge.

- 13 Save the file.

Modifying existing content and formatting

As you can see, the CSS layout comes equipped with a vertical navigation menu. The generic hyperlinks are simply placeholders waiting for your final content. Let's change the placeholder text in the menu to match the pages outlined in the thumbnails created earlier, and modify the colors to match the site color scheme.

- 1 Select the placeholder text *Link one* in the first menu button. Type **Green News**. Change *Link two* to read **Green Products**. Change *Link three* to read **Green Events**. Change *Link four* to read **Green Travel**.



One of the advantages of using bulleted lists as navigational menus is that it's easy to insert new links.

- 2 With the cursor still at the end of the words *Green Travel*, press Enter/Return. Type **Green Tips**.



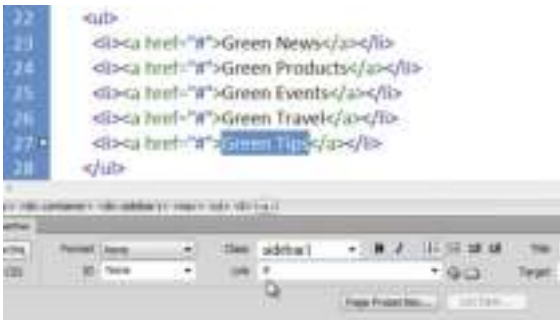
The new text appears in what looks like a button structure, but the background color doesn't match and the text doesn't align with other menu items. You could probably figure out what's wrong in Design view, but in this case, the problem may be identified faster in Code view.

- 3 Click the `` tag selector for the new link item, select Code view. Observe the menu items and compare the first four with the last. Can you see the difference?



The difference is obvious in Code view. The last item is formatted with the `` element like the others—as part of the bulleted list—but it doesn't feature the markup `` used in the other items to create the hyperlink placeholder. For *Green Tips* to look like the other menu items, you have to add a hyperlink, or at least a similar placeholder.

- 4 Select the text *Green Tips*. In the Link field of the HTML Property inspector, type # and press Enter/Return.




The code in all the items is identical now.

- 5 Select Design view.

All the menu items are identically formatted now. You'll learn more about how to format text with CSS to create a dynamic HTML menu in Lesson 5, "Working with Cascading Style Sheets."

The current menu color doesn't match the site color scheme. To change the color, you'll have to use the Code Navigator to find which CSS rule controls this formatting.

- 6 Insert the cursor into any of the menu items.
- 7 Click the Code Navigator () icon, or right-click in the menu and select Code Navigator from the context menu.



The Code Navigator window opens, displaying a list of 12 CSS rules that affect the targeted element. In some cases, the rules listed may only affect the element in a roundabout way, as in the `body` rule, which affects *all* HTML elements on the page.

- 8 Check each of the rules and note which ones format aspects of the menu items.

Remember, more than one rule can, and probably will, format each page element. You're looking for a background color that's applied to the menu items themselves. Be careful. There's probably more than one background color in the listed rules, so if you find one, it's important to determine whether it actually affects the menu or something else.

- 9 Examine each rule until you find the pertinent styling.



The `.sidebar1` rule specifies a background color, but it affects the whole `<div>` element and not just the menu. The one you're actually looking for is applied by the rule `nav ul a, nav ul a:visited`. If you parsed the meaning of the rule selector, it tells you that it specifies the background color for an `<a>` (hyperlink) element that is contained in a `` (unordered list) element that is contained in a `<nav>` element. The second half of the selector then styles the *visited* state of the hyperlink, too. Sound right?

- 10 Locate the rule `nav ul a, nav ul a:visited` in the CSS Styles panel and select it. In the Properties section of the panel, change the existing background color to `#090`.



The background color of the menu items now matches the horizontal `<nav>` element. But the black text is difficult to read against the green background color. As you see in the horizontal menu, a lighter color would be more appropriate. You can use the Properties section of the CSS Styles panel to add, as well as edit, element properties.

- 11 Click the Add Property link in the Properties section of the CSS Styles panel.



A new property field appears.

- 12 Choose Color from the Property field menu. Type #FFC in the Value field.



The link text does not change color as expected. Unfortunately, Dreamweaver missed a problem in the style sheet. The text display in Design view is currently honoring the formatting in the `a:link` rule, which applies default formatting to hyperlinks on the page. According to the Code Navigator, the `nav ul a`, `nav ul a:visited` rule is more specific, and that's why it's listed lower in the Code Navigator display. The selectors `a` and `a:link` are supposed to be equivalent. They both format the default state of hyperlinks, but in the case of this display, `a:link` is being given precedence. Luckily, there's an easy fix.

- 13 In the CSS Styles panel, right-click the `nav ul a`, `nav ul a:visited` rule and select Edit Selector from the context menu, or simply click the name and edit it in the panel list.



Note: No matter how you create or modify the formatting, Dreamweaver updates the CSS markup wherever needed.

Note: The CSS notation `a:link` is one of four pseudo-selectors that are used to format various default hyperlink behaviors. You will learn more about these pseudo-selectors in Lesson 9, "Working with Navigation."

- 14 Change the first half of the selector name to `nav ul a:link`, `nav ul a:visited`, and press Enter/Return to complete the selector.



The link text in the vertical menu now displays in the desired color.

- 15 Save the file.

Inserting an image placeholder

The sidebar will feature photos, captions, and short blurbs on environmental topics. Let's insert a placeholder image and caption below the vertical menu.

- 1 Insert the cursor into the text directly below the vertical menu. Click the `<p>` tag selector.

The placeholder image should not be inserted within the `<p>` element. If it were, it would inherit any margins, padding, and other formatting applied to the paragraph, which could cause it to disrupt the layout.

- 2 Press the Left Arrow key.

As you have seen in earlier exercises, the cursor moves to the left of the opening `<p>` tag in the code but stays within the `<aside>` element.

- 3 Choose Insert > Image Objects > Image Placeholder.

The Image Placeholder dialog box appears.

- 4 In the Image Placeholder dialog box, type **Sidebar** in the Name field.

Type **180** in the Width field.

Type **150** in the Height field.

Click OK.



► **Tip:** Use Split view whenever you're unsure where the cursor is inserted.

● **Note:** The alternate HTML 4 layout doesn't use the `<aside>` element. The `<p>` element is inserted directly following the vertical menu.

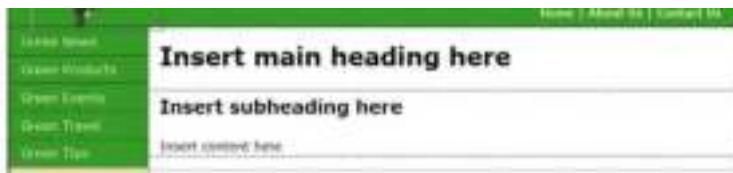
An image placeholder appears in `<div.sidebar1>` below the vertical menu.

- 5 Select all the text below the image placeholder. Type **Insert caption here**.
The caption placeholder replaces the text.
- 6 Press Ctrl-S/Cmd-S to save.

Inserting placeholder text

Let's simplify the layout by replacing the existing headings and text in the main content area.

- 1 Double-click to select the heading *Instructions*. Type **Insert main heading here** to replace the text.
- 2 Select the heading *How to use this document*. Type **Insert subheading here** to replace the text.
- 3 Select the placeholder text in that same `<section>` element. Type **Insert content here** to replace it.



- 4 Select and delete the remaining three `<section>` elements and all their content.
- 5 Press Ctrl-S/Cmd-S to save.

Modifying the footer

Let's reformat the footer and insert the copyright information.

- 1 Double-click the `<footer>` rule in the CSS Styles panel.
- 2 In the Type category, enter **90%** in the Font-size field and **#FFC** in the Color field.
- 3 In the Background category, type **#090** into the Background-color field. Click OK.
- 4 Select the placeholder text in the footer. Type **Copyright 2012 Meridien GreenStart. All rights reserved.**



- 5 Press Ctrl-S/Cmd-S to save.

The basic page layout is complete.

Checking browser compatibility

The CSS layouts included with Dreamweaver have been thoroughly tested to work flawlessly in all modern browsers. However, during the lesson you made major modifications to the original layout. These changes could have ramifications in the compatibility of the code in certain browsers. Before you use this page as your project template, you should check its compatibility.

- 1 If necessary, open **mylayout.html** in Dreamweaver.
- 2 Choose File > Check Page > Browser Compatibility.

When the Report box opens, there should be no issues listed.



- 3 To collapse the report, double-click the Browser Compatibility tab in the Report panel; or to close the entire tab group, right-click the tab and choose Close Tab Group from the context menu.

Congratulations. You created a workable basic page layout for your project template and learned how to insert additional components, image placeholders, text, and headings; adjust CSS formatting; and check for browser compatibility. In the upcoming lessons, you will work further on this file to complete the site template, tweak the CSS formatting and set up the template structure.

Review questions

- 1 What three questions should you ask before starting any web design project?
- 2 What is the purpose of using thumbnails and wireframes?
- 3 What is the advantage of inserting the banner as a background image?
- 4 How can you insert the cursor before or after an element without using Code view?
- 5 How does the Code Navigator assist in designing your website layout?
- 6 What advantages can be had by using HTML5-based markup?

Review answers

- 1 What is the purpose of the website? Who is the customer? How did they get here? These questions, and their answers, are essential in helping you develop the design, content, and strategy of your site.
- 2 Thumbnails and wireframes are quick techniques for roughing out the design and structure of your site without having to waste lots of time coding sample pages.
- 3 By inserting the banner or other large graphics as a background image, you leave the container free for other content.
- 4 Select an element using its tag selector, and press the Left Arrow or Right Arrow key to move the cursor before or after the selected element.
- 5 The Code Navigator serves the role of a CSS detective. It allows you to investigate what CSS rules are formatting a selected element and how they are applied.
- 6 HTML5 has introduced new semantic elements that help to streamline code creation and styling. These elements also allow search engines, like Google and Yahoo, to index your pages more quickly and effectively.

INDEX

<!-- . . . --> (HTML comments), 30, 107
< > (angle brackets), 21
/* and */ (CSS comments), 52, 109

A

<a> (anchor) tags, 30
 hyperlinks, 234
 pseudoclasses, 123
a:active pseudoclass, 123–127
absolute hyperlinks, 234–235
Access databases, 348
accessibility preferences, forms,
 307–308
action attributes, 304
Adams, Cameron, 68
Adobe and Ajax's Spry framework
 Spry accordion widgets, 279–280
 customizing, 282–284
 inserting, 280–281
 panels, adding, 281–282
 Spry data, 249
 date, 340
 html, 340
 HTML table-based data, 338–344
 number, 340
 string, 340
 Spry effects, 249
 Spry form widgets, 249
 checkboxes, 315–317
 lists/menus, 320–322
 passwords, 313–315
 radio buttons, 317–318
 text areas, 319–320
 text fields, 309–315
 Spry layout widgets, 249
 Spry menu bars, 248–254
 customizing, 257–263
 modifying, 254–257
Adobe Bridge insertions
 by dragging and dropping, 227
 of images, 215–217
 of non-web file types, 218
Adobe Business Catalyst, 78
Adobe Dreamweaver CS3 with ASP, ColdFusion, and PHP: Training from the Source, 328
Adobe Dreamweaver CS5 with PHP: Training from the Source, 328
Adobe Edge, 290–292
Adobe Fireworks
 copying/pasting images from,
 223–226
 website mockups, 77–78
Adobe Flash
 lack of support for, 74
 web animation, 288–289
 web video, 288–289, 294–297
Adobe Illustrator
 vector graphics, 204
 website mockups, 77–78
Adobe Omniture, 73
Adobe Photoshop
 copying/pasting images from,
 223–226
 inserting Smart Objects, 221–223
 PSD (Photoshop Document) file
 format, 218, 221
 website mockups, 77–78
a: hover pseudoclass, 123–127, 125
AI file format, 204, 215

- Ajax (Asynchronous JavaScript and XML),
 - dynamic content, 336, 347
 - Ajax and Adobe's Spry framework, 248–249
 - Spry accordion widgets, 279–280
 - customizing, 282–284
 - inserting, 280–281
 - panels, adding, 281–282
 - Spry data, 249
 - date, 340
 - html, 340
 - HTML table-based data, 338–344
 - number, 340
 - string, 340
 - Spry effects, 249
 - Spry form widgets, 249
 - checkboxes, 315–317
 - lists/menus, 320–322
 - passwords, 313–315
 - radio buttons, 317–318
 - text areas, 319–320
 - text fields, 309–315
 - Spry layout widgets, 249
 - Spry menu bars, 248–254
 - customizing, 257–263
 - modifying, 254–257
 - alignment
 - page elements, 92
 - tables, 193
 - a:link pseudoclass, 99–100, 123–125
 - alpha transparencies
 - PNG files, 89
 - raster graphics, 207
 - alphanumeric characters, forms, 308
 - alt (text) attribute, 210
 - alternate workflows, HTML 4, 85, 171
 - anchor (<a>) tags, 30
 - hyperlinks, 234
 - pseudoclasses, 123
 - angle brackets (< >), 21
 - animation
 - with Adobe Flash, 288
 - basics, 288
 - with HTML5, 288, 290–293
 - AP divs, 87–89
 - AP divs *versus* <div> elements, 120
 - Application bar, 4
 - The Art & Science of CSS*, 68
 - <article> (self-contained content) tags, 82–83, 85
 - <h1> in each semantic structure, 173
 - semantic design, 33–35
 - <article.content> tags, 115–117
 - template editable regions, 145
 - <aside> (sidebar) tags, 33, 82–83
 - ASP (Active Server Pages)
 - Dreamweaver support, 4
 - .html extension, 357
 - web form mail scripts, 328
 - web server models, 348–349
 - Assets panel, 211
 - Asynchronous JavaScript and XML. *See* Ajax
 - <audio> (multimedia) tags, 33–34
 - a:visited pseudoclass, 123–126
- ## B
- (break) tags, 30
 - (bold) tags *versus* tags, 24–25
 - backgrounds
 - color, 84–85, 87, 93–95
 - CSS, 59–60
 - HTML, 44
 - images, headers, 84, 86–87
 - Bardzell, Jeffrey, 328, 349
 - behaviors
 - applying, 275–276
 - basics, 270–271
 - displaying
 - multiple items, 376
 - record counts, 379–380
 - go-to-detail-page, 383
 - hiding paging controls, 378–379
 - hyperlinks
 - adding to, 277–279
 - modifying, 123–127
 - record paging, 377–378
 - removing, 277
 - repeat regions, 369–370
 - source, Adobe Marketplace & Exchange, 270

- Spry accordion widgets, 279–280
 - customizing, 282–284
 - inserting, 280–281
 - panels, adding, 281–282
 - Swap Image, 271
 - Swap Image Restore, 276–277
 - uses, 270–271
 - Berners-Lee, Tim, 20–21
 - bit depth, 206
 - block tags, 24
 - `<blockquote>` (quotation) tags, 30
 - text indents, 178
 - BMP file format, 215
 - `<body>` tags, 21–22, 25, 30, 82
 - body text, HTML, 44
 - boldface (``) tags, 31
 - versus `` tags, 24–25
 - Bolton, Jina, 68
 - book resources
 - Adobe Dreamweaver CS3 with ASP, ColdFusion, and PHP: Training from the Source*, 328, 349
 - Adobe Dreamweaver CS5 with PHP: Training from the Source*, 328
 - The Art & Science of CSS*, 68
 - Build Your Own Database Driven Web Site, 4th Edition*, 328
 - Bulletproof Web Design: Improving Flexibility and Protecting Against Worst-Case Scenarios with HTML5 and CSS3 (3rd edition)*, 68
 - CSS: The Missing Manual*, 68
 - Dreamweaver Marketplace & Exchange, behaviors, 270
 - Head First HTML with CSS & XHTML*, 29
 - HTML, XHTML, and CSS3: Visual QuickStart Guide, Seventh Edition*, 29
 - HTML and XHTML Pocket Reference*, 29
 - Introduction to Adobe Edge Preview 6 e-book*, 290
 - Stylin' with CSS: A Designer's Guide (2nd edition)*, 68
 - border-collapse property, 186
 - borders, 59–60, 66–67, 91, 95
 - box model, CSS, 45–46
 - box shadows, 66–67
 - break (``) tags, 30
 - Bridge (Adobe) insertions
 - by dragging and dropping, 227
 - of images, 215–217
 - of non-web file types, 218
 - brightness/contrast of images, 230
 - browsers
 - alpha transparency support, 89
 - hyperlink destinations, 236
 - most popular, 44
 - opening with behaviors, 270
 - testing webpages on, 44–45, 102
 - Build Your Own Database Driven Web Site, 4th Edition*, 328
 - bullet entity, 32
 - Bulletproof Web Design: Improving Flexibility and Protecting Against Worst-Case Scenarios with HTML5 and CSS3 (3rd edition)*, 68
 - Business Catalyst (Adobe), 78
- ## C
- `<canvas>` (graphics) tags, 33
 - captions, tables, 194–195
 - cascade theory, CSS text formatting, 47–48
 - Castro, Elizabeth, 29
 - Cederholm, Dan, 68
 - cells, tables, 44, 187–189
 - CERN (French for European Council for Nuclear Research), 20
 - checkboxes, forms, 304, 306, 315–317
 - classes, `<class>` attributes, 65, 82, 85
 - custom classes, 117–119
 - Classic workspace layout, 11
 - client-based email, hyperlinks, 244
 - code hinting, 27
 - Code Navigator, 53–55, 92–94
 - editing CSS, 263–265
 - Code view, 4–5, 26–29
 - Coding toolbar, 13
 - Split Vertically, 41–42
 - Coder workspace layout, 12
 - Coding toolbar, 4, 13

- ColdFusion
 - Dreamweaver support, 4
 - .html extension, 357
 - RDS (Remote Development Services), 390
 - web form mail scripts, 328
 - web server models, 348–349
- color
 - alpha transparencies, 207
 - backgrounds, 95
 - bit depth, 206
 - dithering, 206
 - high color, 207
 - images, 84, 86–87
 - 3D, 113–114
 - raster graphics, 206–207
 - true color, 207
 - web-safe colors, 207
- columns
 - faux columns, 128–130
 - multicolumn elements, 66–67
 - width in tables, 189–191
- comma-separated values (CSV) files, 191
- comments
 - CSS (`/*` and `*/`), 52, 109
 - HTML (`<!-- . . . -->`), 30, 107
- Compare command, 405
- compression of images, 208–209
- connection methods to remote sites
 - FTP (File Transfer Protocol), 390–394
 - FTP (File Transfer Protocol) over SSL/TLS (explicit encryption), 390
 - FTP (File Transfer Protocol) over SSL/TLS (implicit encryption), 390
 - local/network, 390
 - RDS (Remote Development Services), 390
 - SFTP (Secure File Transfer Protocol), 390
 - WebDav (Web Distributed Authoring and Versioning), 390
- Conseil Européen pour la Recherche Nucléaire (CERN), 20
- contrast/brightness of images, 230
- `<© >` (copyright entity), 32
- copyright entity (`<© >`), 32
- CorelDRAW, vector graphics, 204
- cropping images, 230
- CSS (cascading style sheets), 24. *See also* CSS
 - Styles panel; CSS3
 - box model, 45–46
 - code hinting, 27
 - Code Navigator, 53–55
 - content *versus* presentation, 46
 - definition, 40–41
 - disabling/enabling, 83–84
 - editing with Code Navigator, 263–265
 - formatting objects
 - backgrounds, 59–60
 - borders, 59–60
 - height, 57–59
 - margins, 60–62
 - padding, 60–62
 - positioning, 62–64
 - width, 56–57
 - formatting text, 46–47
 - cascade theory, 47–48
 - descendant theory, 50–51
 - versus* HTML-based formatting, 41–42
 - inheritance theory, 48–50
 - multiple elements, 64
 - specificity theory, 51–53
 - `<style>` tags, 31
 - formatting text or objects
 - class attributes, 65
 - versus* HTML-based formatting, 41–42
 - ID attributes, 65, 119–122
 - multiple elements, 64
 - modifying existing layouts, 81–84
 - normalization, 62
 - pseudoclasses, 123
 - web animation and video, 288
- CSS Styles panel, 4, 106–110
 - alignment, 92
 - backgrounds
 - color, 95
 - images, 84, 86–87
 - images, 3D effect, 113–114
 - faux columns, 128–130
 - interactive menus, 122–125
 - visual effects, 127–128
 - page width, 93–94
 - Properties section, 94
 - rollover effects, 122

- rules
 - basics, 106–110
 - creating, 89–90, 115–122
 - custom classes, 117–119
 - custom IDs, 119–122
 - descendant selectors, 115–116
 - moving in lists, 108–109
 - moving to external style sheets, 130–132
 - reordering in lists, 110
 - Spry tables, 342–344
 - style sheets
 - external, 130–132
 - forms, 329–331
 - hiding content, 134–137
 - for other media, 132
 - for print, 133–137
 - removing unwanted styles, 137–138
 - Style Rendering toolbar, 132–133
 - tables, 185–187
 - captions, 194–195
 - cells, 187–189
 - column width, 189–191
 - vertical alignment, 193
 - text/type
 - font groups, 110–112
 - font sizes, 112–113
 - Trash Can icon, 126
 - CSS: The Missing Manual*, 68
 - CSS3 (cascading style sheets). *See also* CSS; CSS
 - Styles panel
 - features and effects, 66–67
 - overview, 66
 - support, 68
 - CSV (comma-separated values) files, 191
- ## D
- databases, 352
 - connections with MySQL, 354–358
 - data display, 367–368, 371–372
 - repeat regions, 369–370
 - testing servers, 369
 - design basics, 352–354
 - master/detail page sets, 372–374
 - behaviors, displaying multiple items, 376
 - behaviors, displaying record counts, 379–380
 - behaviors, go-to-detail-page, 383
 - behaviors, hiding paging controls, 378–379
 - behaviors, record paging, 377–378
 - detail pages, 384–386
 - displaying images, 374–376
 - master pages, 372–374
 - styling dynamic data, 380–382
 - recordsets, 362–366
 - `<datalist>` tags, 307
 - `<dataroot>` tags, 346
 - date full date data, 340
 - declarations, CSS rules, 46–47
 - Delete command, 405
 - descendant theory, 50–51, 115–116
 - Design view, 4–5, 41–42
 - Designer workspace layout, 12
 - detail pages, 384–386
 - dithering, 206
 - `<div>` (division) tags, 30, 83, 85–95, 98
 - versus* AP divs, 120
 - `<div#apDiv1>` tags, 89, 108, 113
 - `<div.container>` tags, 82, 93–95, 108
 - `<div.content>` tags, 95
 - alternate workflow, 171
 - `<div#logo>` tags, 120
 - `<div.sidebar1>` tags, 101, 112
 - templete editable regions, 145
 - Document toolbar, 4, 13
 - dragging/dropping images, 227
 - dynamic content
 - Ajax, 336
 - basics, 336
 - database applications, 352
 - design basics, 352–354
 - database connections, with MySQL, 354–358
 - HTML data, 338–342
 - updating, 344–345
 - Spry data, 338–342
 - styling tables, 342–344
 - Spry data types, 340

web server models

- ASP (Active Server Pages), 338, 348, 357
- ColdFusion, 338, 348–349, 357
- PHP (PHP Hypertext Processor), 338, 349–352, 354–358, 357

XML data, 345–347

dynamic webpages

behaviors

- displaying multiple items, 376
- displaying record counts, 379–380
- go-to-detail-page, 383
- hiding paging controls, 378–379
- record paging, 377–378
- repeat regions, 369–370

data display, 367–368, 371–372

- testing servers, 369

detail pages, 384–386

master/detail page sets

- displaying images, 374–376
- master pages, 372–374
- styling dynamic data, 380–382

recordsets, 362–366

E

Edge (Adobe), 290–292

em dash entity, 32

 (italic) tags, 30

- versus* <i> tags, 24–25

email

- data from forms, 324–329
- hyperlinks, 237, 243–244

embedded formatting, 46

<empty> tags, 375, 386

em-width *versus* fixed-width containers, 259

en dash entity, 32

entities, 23, 32

EPS (Encapsulated PostScript) file format, 204, 215

European Council for Nuclear Research

(CERN), 20

Extensible Markup Language (XML), 204

external hyperlinks, 234, 241–243

F

faux columns, 128–130

field sets, forms, 312–313

<fieldset> tags, 312–313, 315, 322

<figcaption> (caption) tags, 33

<figure> (images or video) tags, 33

File Transfer Protocol (FTP), 390–394

- over SSL/TLS (explicit encryption), 390

- over SSL/TLS (implicit encryption), 390

Files panel, 4

finding/replacing

- HTML table data, 344–345

- text, 197–200

Firefox (Mozilla), 73

Fireworks (Adobe)

- copying/pasting images from, 223–226

- website mockups, 77–78

fixed-width *versus* em-width containers, 259

Flash (Adobe)

- lack of support for, 74

- web animation, 288–289

- web video, 288–289, 294–297

Flash Player, 288, 297

FLV file format, 288, 294–295, 294–297

Flynn, Bob, 328, 349

 tags, 41

fonts, HTML, 44

<footer> tags, 83, 85, 101–102

- <h1> in each semantic structure, 173

- semantic design, 33–35

<form> tags, 30, 303–305

formatting objects

CSS

- backgrounds, 59–60

- borders, 59–60

- class attributes, 65

- height, 57–59

- versus* HTML-based formatting, 41–42

- ID attributes, 65

- margins, 60–62

- multiple elements, 64

- padding, 60–62

- positioning, 62–64

- width, 56–57

- CSS3
 - borders, 66–67
 - box shadows, 66–67
 - gradient fills, 66–67
 - rounded corners, 66–67
 - transparency and translucency, 66–67
 - HTML
 - backgrounds, 44
 - padding, 44
 - formatting text
 - CSS, 46–47
 - cascade theory, 47–48
 - class attributes, 65
 - descendant theory, 50–51
 - font groups, 110–112
 - font sizes, 112–113
 - versus* HTML-based formatting, 41–42
 - ID attributes, 65
 - inheritance theory, 48–50
 - multiple elements, 64
 - placeholders, 101
 - specificity theory, 51–53
 - CSS3
 - multicolumn elements, 66–67
 - shadows, 66–67
 - HTML
 - body, 44
 - fonts, 44
 - headings, 44
 - margins, 44
 - table cells, 44
 - table headers, 44
 - forms
 - accessibility preferences, 307–308
 - actions, specifying, 324–325
 - adding to webpages, 305–306
 - basics, 303–304
 - checkboxes, 304, 306, 315–317
 - emailing data, 325–329
 - <form> tags, 30, 303–305
 - lists/menus, 304, 306, 320–322
 - radio buttons, 304, 306, 317–318
 - Spry form widgets, 249
 - checkboxes, 315–317
 - lists/menus, 320–322
 - passwords, 313–315
 - radio buttons, 317–318
 - text areas, 319–320
 - text fields, field sets, 312–313
 - text fields, passwords, 313–315
 - text fields, validation, 309–312
 - styling, 329–331
 - Submit buttons, 322–324
 - text areas, 304, 306, 319–320
 - text fields, 304, 306, 308–309
 - field sets, 312–313
 - passwords, 313–315
 - validation, 309–312
 - Freeman, Elizabeth and Eric, 29
 - FTP (File Transfer Protocol), 390–394
 - over SSL/TLS (explicit encryption), 390
 - over SSL/TLS (implicit encryption), 390
- ## G
- Get command, 403–405
 - GET method, 324–325
 - GIF (graphic interchange format) file format
 - Adobe Bridge, 215–216
 - Adobe Photoshop and Fireworks, 223
 - basics, 208
 - Google Analytics, 73
 - Google Chrome, 73
 - go-to-detail-page behavior, 383
 - gradient fills, 66–67
 - graphic interchange format (GIF) file format
 - Adobe Bridge, 215–216
 - Adobe Photoshop and Fireworks, 223
 - basics, 208
 - graphics
 - <canvas> tags, 33
 - growing/shrinking with behaviors, 270
- ## H
- <h1> to <h6> (headings) tags, 21, 30, 173–175
 - custom classes, 117–119
 - descendant selectors, 115–116
 - <h1> in each semantic structure, 173
 - Head First HTML with CSS & XHTML*, 29

- <head> tags, 21, 25, 30
 - template editable regions, 144
- <header> tags, 83, 85–87, 108
 - <h1> in each semantic structure, 173
 - semantic design, 34–35
- headers, backgrounds, adding images, 86–87
- height of objects, 57–59
- help
 - books
 - Adobe Dreamweaver CS3 with ASP, ColdFusion, and PHP: Training from the Source*, 328, 349
 - Adobe Dreamweaver CSS with PHP: Training from the Source*, 328
 - The Art & Science of CSS*, 68
 - Build Your Own Database Driven Web Site, 4th Edition*, 328
 - Bulletproof Web Design: Improving Flexibility and Protecting Against Worst-Case Scenarios with HTML5 and CSS3 (3rd edition)*, 68
 - CSS: The Missing Manual*, 68
 - Head First HTML with CSS & XHTML*, 29
 - HTML, XHTML, and CSS3: Visual QuickStart Guide, Seventh Edition*, 29
 - HTML and XHTML Pocket Reference*, 29
 - Introduction to Adobe Edge Preview 6* e-book, 290
 - SQL (Structured Query Language), 362
 - Stylin' with CSS: A Designer's Guide (2nd edition)*, 68
 - W3Schools/other resources
 - browser statistics, 44, 73
 - Dreamweaver Marketplace & Exchange, behaviors, 270
 - HTML, 29
 - elements list, 35
 - local web server configuration, 349
 - operating systems, 73
 - screen resolutions and sizes, 73
 - SQL (Structured Query Language), 362
 - SVG graphics, 204
 - web form mail scripts, 328
- <hgroup> (multiple-level heading) tags, 34
- hidden data fields, forms, 304
- high color, 207
- home pages, 398–400
- horizontal rule (<hr />) tags, 31
- <hr /> (horizontal rule) tags, 21, 31, 59
- HTML, XHTML, and CSS3: Visual QuickStart Guide, Seventh Edition*, 29
- HTML 4 (HyperText Markup Language)
 - alternate workflow, 85, 171
 - angle brackets, 21
 - code, basic structure, 21
 - elements, 21
 - elements, block, 24
 - entities, 23
 - nonbreaking spaces, 23
 - root, 21
 - slashes, forward, 21
 - syntax, 22–23
 - tags, 21
 - tags, frequently used, 30–31
 - tags, heading, 24
 - content *versus* presentation, 46
 - defaults, 42–45
 - definition, 20
 - layouts available, 81
 - origin, 20–21
 - text
 - versus* CSS formatting, 41–42
 - displaying, 23
 - editing, 22
 - heading tags, 24
 - inline, 24
 - HTML and XHTML Pocket Reference*, 29
- html marked-up text data, 340
- <html> tags, 21–22, 25
- HTML5 (HyperText Markup Language)
 - alternate workflow, 85, 171
 - backward compatibility, 32
 - layouts available, 81
 - new features, 33
 - semantic web design, 34–35, 82
 - tags, new, 33–34
 - web animation, 288, 290–293
 - web video, 288, 293–294

- hyperlinks
 - <a> elements, 234
 - behaviors, 109
 - adding, 277–289
 - modifying, 123–127
 - checking, 265–266
 - email, 243–244
 - external, 234, 241–243
 - with ID attributes, 246–248
 - image-based, 240–241
 - interactive menus, 122–125
 - visual effects, 127–128
 - internal, 234, 237–240
 - internal targeted, 245–246
 - pseudoclasses, 123–127
 - relative *versus* absolute, 234–235
 - Spry menu bars, 248, 250–254
 - customizing, 257–263
 - modifying directly, 254–257
 - updating remote sites, 400–401
- HyperText Markup Language. *See* HTML
- Hyslop, Bruce, 29

I

- <i d> attributes, 65, 82
 - custom IDs, 119–122
 - hyperlinks to page sections, 246–248
 - multiple elements, 64
- iDisk, 390
- <i frame> (inline frame) tags, 31
- Ignore command, 405
- IIS (Internet Information Services), 348
- Illustrator (Adobe)
 - vector graphics, 204
 - website mockups, 77–78
- images
 - adjusting brightness and contrast, 230
 - backgrounds
 - 3D effect, 113–114
 - headers, 84, 86–87
 - copying/pasting from Adobe Photoshop or Fireworks, 223–226
 - cropping, 230
 - editing image settings, 230
 - fading in/out with behaviors, 270
 - <figure> tags, 33
 - image-based hyperlinks, 240–241
 - inserting
 - with Adobe Bridge, 215–217
 - with Adobe Photoshop Smart Objects, 221–223
 - with built-in tools, 210–212
 - with dragging and dropping, 227
 - with Insert panel, 213–214
 - non-web file types, 218–221
 - positioning with CSS classes, 212–213
 - master/detail page sets, 374–376
 - optimizing with Property inspector, 227–229
 - placeholders, 100–101
 - raster graphics
 - color, 206–207
 - file formats, 207
 - file formats, GIF, 208
 - file formats, JPEG, 208
 - file formats, PNG, 209
 - pixels, 204–205
 - resolution, 205–206
 - size, 206
 - resampling, 230
 - sharpening, 230
 - updating, 230
 - vector graphics, 204
- (image) tags, 31, 212
- indents, 178–181
- inheritance theory, 48–50
- inline formatting, 46
- inline frame (<i frame>) tags, 31
- <input /> tags, 31, 306
- Insert panel, 213–214
- internal hyperlinks, 234, 237–240
 - to page sections, 245–246
- Internet application-based email, hyperlinks, 244
- Internet Explorer, 73
- intranets, 390
- Introduction to Adobe Edge Preview 6* e-book, 290
- Invert panel, 4
- italic typeface () tags, 30
 - versus* <i> (italic) tags, 24–25

J

- JavaScript. *See also* Ajax
 - calling functions with behaviors, 271
 - code hinting, 27
 - Dreamweaver support, 4
- Johnson, David, 68
- JPEG (Joint Photographic Experts Group) file
 - format
 - Adobe Bridge, 215–216, 219
 - Adobe Photoshop and Fireworks, 223
 - basics, 208
- jQuery Mobile, 270

K

- keyboard shortcuts, 14–15
- <keygen> tags, 307

L

- <label> tags, 308
- (list) tags, 31, 96, 178
- library items. *See also* SSIs; templates
 - basics, 152
 - creating, 152–156
 - versus* server-side includes, 164
 - updating, 156–157
- <link /> (document-external resource) tags, 31
- linked formatting, 46
- lists/menus
 - formatting, 175–178
 - forms, 304, 306
 - Spry widgets, 320–322
 - interactive menus, 122–125
 - visual effects, 127–128
- tags
 - (list), 31, 96, 178
 - (ordered list), 31, 178
 - (unordered list), 31, 98
- local servers, 349, 394
- local sites, 390
 - connecting to remote sites, 390
 - publishing (putting) online, 402–404
 - synchronizing with remote sites, 393, 404–405
- local/network connections to remote sites, 390
- lossy/lossless compression, 208–209

M

- margins
 - CSS, 60–62
 - HTML, 44
- master/detail page sets, 372–374
 - behaviors
 - displaying multiple items, 376
 - displaying record counts, 379–380
 - go-to-detail-page, 383
 - hiding paging controls, 378–379
 - record paging, 377–378
 - detail pages, 384–386
 - displaying images, 374–376
 - master pages, 372–374
 - styling dynamic data, 380–382
- Max Chars attribute, 308
- McFarland, David Sawyer, 68
- media/multimedia
 - animation
 - with Adobe Flash, 288
 - basics, 288
 - with HTML5, 288, 290–293
 - tags
 - <audio>, 34
 - <figcaption>, 33
 - <figure>, 33
 - (image), 31
 - <source>, 34
 - <track>, 34
 - <video>, 34
 - video
 - with Adobe Flash, 288–289, 294–297
 - basics, 288
 - with HTML5, 288, 293–294
- menu bar, 4
- menus/lists
 - formatting, 175–178
 - forms, 304, 306
 - Spry widgets, 320–322
 - interactive menus, 122–125
 - visual effects, 127–128
- tags
 - (list), 31, 96, 178
 - (ordered list), 31, 178
 - (unordered list), 31, 98

- <meta /> (metadata) tags, 31
- metadata (<meta />) tags, 31
- Microsoft
 - Access databases, 348
 - IIS (Internet Information Services), 348
- Mozilla Firefox, 73
- MP4 file format, 289, 294
- multicolumn elements, 66–67
- MySQL databases, 348, 354–358

N

- <nav> (navigation section) tags, 34, 82–83, 90–91
- navigation and hyperlinks
 - <a> elements, 234
 - email, 243–244
 - external, 234, 241–243
 - image-based, 240–241
 - internal, 234, 237–240
 - relative *versus* absolute, 234–235
 - Spry menu bars, 248, 250–254
 - customizing, 257–263
 - modifying directly, 254–257
 - targeting page elements
 - with ID attributes, 246–248
 - internal, 245–246
 - named anchors, 244
- < > (nonbreaking spaces), 23, 32
- network connections to remote sites, 390
- network web servers, 394
- normalization, 62
- number numeric data, 340

O

- object formatting
 - CSS
 - backgrounds, 59–60
 - borders, 59–60
 - class attributes, 65
 - height, 57–59
 - versus* HTML-based formatting, 41–42
 - ID attributes, 65
 - margins, 60–62
 - multiple elements, 64

- padding, 60–62
- positioning, 62–64
- width, 56–57

CSS3

- borders, 66–67
- box shadows, 66–67
- gradient fills, 66–67
- rounded corners, 66–67
- transparency and translucency, 66–67

HTML

- backgrounds, 44
- padding, 44

- <object> tags, 296

- Ogg file format, 289, 294

- (ordered list) tags, 31, 178

- Omniure (Adobe), 73

P

- <p> (paragraph) tags, 23, 31

- padding

- CSS, 60–62

- HTML, 44

- panels

- Spry accordion widgets, 281–282

- in workspaces

- available panels, 7

- docking, 10

- floating, 8

- grouping, 9

- minimizing/expanding, 7–8

- reordering panel tabs, 9

- repositioning panels, groups, and stacks, 9

- stacking, 10

- standard grouping, 7

- paragraph (<p>) tags, 23, 31

- passwords, forms, 313–315

- PDF (Portable Document Format) file format, 215

- PhoneGap, 270

- Photoshop (Adobe)

- copying/pasting images from, 223–226

- inserting Smart Objects, 221–223

- PSD (Photoshop Document) file format, 218,

- 221

- website mockups, 77–78

- PHP (PHP Hypertext Processor)
 - data display, 367–368
 - database connections with MySQL, 354–358
 - Dreamweaver support, 4
 - .html extension, 357
 - recordsets, 362–367
 - testing servers, 350–352
 - web form actions, 324–325
 - web form mail scripts, 325–328
 - web server models, 349
- PICT file format, 204
- pixels, 204–205
- placeholders, images and text, 100–101
- PNG (Portable Network Graphic) file format, 209
 - Adobe Bridge, 215–216
 - Adobe Photoshop and Fireworks, 223
 - alpha transparencies, 89
 - raster graphics, 209
- pop-up messages, 271
- positioning, 62–64
- POST method, 324–325
- Powers, David, 328
- prebuilt workspace layouts
 - accessing, 11
 - Classic, 11
 - Coder, 12
 - Designer, 12
- preferences
 - accessible forms, 307–308
 - custom workspaces, 13
 - enabling/disabling Welcome screen, 80
 - graphics-editing programs, 230
 - hiding/showing invisible elements, 161
 - secondary proxy host connections, 392
 - uploading dependent files to remote sites, 369, 402
- printing
 - CSS style sheets, 133–137
 - dpi (dots per inch), 206
- Properties section, CSS Styles panel, 94, 98–99
- Property inspector, 4, 28
 - CSS tab, 16
 - HTML tab, 15
 - images, 16
 - optimizing images, 227–229
 - tables, 16

- PSD (Photoshop Document) file format, 218, 221
- pseudoclasses, 123–127
- psuedo-selectors, a:link, 99–100
- publishing (putting) to remote sites, 402–404
 - dependent files, 402
 - prelaunch checklist, 401
- Put command, 403–405

Q

- quotation (<blockquote>) tags, 30

R

- radio buttons, forms, 304, 306, 317–318
- raster graphics
 - color, 206–207
 - file formats, 207
 - GIF, 208
 - JPEG, 208
 - PNG, 209
 - pixels, 204–205
 - resolution, 205–206
 - size, 206
- RDS (Remote Development Services), 390
- record paging behavior, 377–378
- recordsets, 362–366
- <®> (registered trademark entity), 32
- relative hyperlinks, 234–235
- Remote Development Services (RDS), 390
- remote sites
 - cloaking folders and files, 396–398
 - connection methods supported
 - FTP (File Transfer Protocol), 390–394
 - FTP (File Transfer Protocol) over SSL/TLS (explicit encryption), 390
 - FTP (File Transfer Protocol) over SSL/TLS (implicit encryption), 390
 - local/network, 390
 - RDS (Remote Development Services), 390
 - SFTP (Secure File Transfer Protocol), 390
 - WebDav (Web Distributed Authoring and Versioning), 390
- definition, 390
- home pages, 398–400

- on local or network web servers, 394–396
- publishing (putting) online, 402–404
 - dependent files, 402
 - prelaunch checklist, 401
- synchronizing with local sites, 404–405
- updating links, 400–401
- resolution, 205–206
- resources
 - books
 - Adobe Dreamweaver CS3 with ASP, ColdFusion, and PHP: Training from the Source*, 328, 349
 - Adobe Dreamweaver CS5 with PHP: Training from the Source*, 328
 - The Art & Science of CSS*, 68
 - Build Your Own Database Driven Web Site, 4th Edition*, 328
 - Bulletproof Web Design: Improving Flexibility and Protecting Against Worst-Case Scenarios with HTML5 and CSS3 (3rd edition)*, 68
 - CSS: The Missing Manual*, 68
 - Head First HTML with CSS & XHTML*, 29
 - HTML, XHTML, and CSS3: Visual QuickStart Guide, Seventh Edition*, 29
 - HTML and XHTML Pocket Reference*, 29
 - Introduction to Adobe Edge Preview 6*
 - e-book, 290
 - SQL (Structured Query Language), 362
 - Stylin' with CSS: A Designer's Guide (2nd edition)*, 68
 - W3Schools/other resources
 - browser statistics, 44, 73
 - Dreamweaver Marketplace & Exchange,
 - behaviors, 270
 - HTML, 29
 - elements list, 35
 - local web server configuration, 349
 - operating systems, 73
 - screen resolutions and sizes, 73
 - SQL (Structured Query Language), 362
 - SVG graphics, 204
 - web form mail scripts, 328
- Robbins, Jennifer Niederst, 29

- rollover effects
 - behaviors, 270
 - CSS Styles panel, 122
- root, HTML, 21
- rounded corners, 66–67
- rulers, 92
- rules, CSS, 46–47
 - basics, 106–110
 - creating, 89–90, 115–122
 - custom classes, 117–119
 - custom IDs, 119–122
 - descendant selectors, 115–116
 - HTML 4 elements, 85
 - moving in lists, 108–109
 - moving to external style sheets, 130–132
 - proximity and order, 48
 - reordering in lists, 110
 - Spry tables, 342–344
 - targeted elements, 97–98

S

- sampling/resampling images, 230
- Scalable Vector Graphic (SVG) file format, 204
- `<script>` tags, 31
- search engines, `<title>` tags, 26
- `<section>` tags
 - `<h1>` in each semantic structure, 173
 - semantic design, 34–35, 82–83, 85, 101
 - semantic design, 172
- Secure File Transfer Protocol (SFTP), 390
- selectors, CSS rules, 46–47
- semantic web design
 - basics, 34–35
 - new elements, 82
 - text, 172–173
- server-side includes (SSIs). *See also* library items;
 - templates
 - basics, 157–158
 - creating, 158–159
 - inserting, 159–162
 - versus* templates and library items, 164
 - updating, 162–164
 - uploading to remote servers, 398, 402
 - viewing by enabling invisible elements, 161

- shadows, body or text, 66–67
- sharpening images, 230
- shortcut keys. *See* keyboard shortcuts
- sidebar (<aside>) tags, 33
- sizing images
 - raster graphics, 206
 - Resample tool, 230
- slashes, forward, 21
- Smart Objects, 221–223
- Smith, Steve, 68
- Snook, Jonathan, 68
- <source> (media source) tags, 34, 293–294
- (element section) tags, 31
- specificity theory, CSS text formatting, 51–53
- spell checking, 195–197
- Split view, 6, 41
- Spry accordion widgets, 279–280
 - customizing, 282–284
 - inserting, 280–281
 - panels, adding, 281–282
- Spry data, 249
 - date, 340
 - html, 340
 - HTML table-based data, 338–342
 - styling, 342–344
 - number, 340
 - string, 340
- Spry effects, 249
- Spry form widgets, 249
 - checkboxes, 315–317
 - lists/menus, 320–322
 - passwords, 313–315
 - radio buttons, 317–318
 - text areas, 319–320
 - text fields
 - field sets, 312–313
 - passwords, 313–315
 - validation, 309–312
- Spry layout widgets, 249
- Spry menu bars, 248–254
 - customizing, 257–263
 - modifying, 254–257
- SQL (Structured Query Language), 362–368
- src attribute, 293–294
- SSIs (server-side includes). *See also* library items; templates
 - basics, 157–158
 - creating, 158–159
 - inserting, 159–162
 - versus* templates and library items, 164
 - updating, 162–164
 - uploading to remote servers, 398, 402
 - viewing by enabling invisible elements, 161
- staging servers, 390
- Standard toolbar, 13
- SFTP (Secure File Transfer Protocol), 390
- string alphanumeric data, 340
- (boldface) tags, 31
 - versus* tags, 24–25
- Style Rendering toolbar, 13, 132–133
- style sheets, CSS
 - external, 130–132
 - forms, 329–331
 - hiding content, 134–137
 - for other media, 132
 - for print, 133–137
 - removing unwanted styles, 137–138
 - Style Rendering toolbar, 132–133
- <style> (CSS style rules) tags, 31, 107, 132
- Stylin' with CSS: A Designer's Guide* (2nd edition), 68
- Submit buttons, forms, 322–324
- SVG (Scalable Vector Graphic) file format, 204
 - Adobe Bridge, 215
- Swap Image behavior, 271
- Swap Image Restore behavior, 276–277
- SWF file format, 288
- Synchronized command, 405
- synchronizing remote with local sites, 393, 404–405
- syntax, 22–23

T

- tabindex attribute, 323
- tables
 - captions, 194–195
 - cells, 187–189
 - column width, 189–191
 - creating, 182–184
 - hyperlinks to, 247–248
 - inserting from other sources, 191–193
 - styling, 185–187

- tags
 - cell text, 44
 - headers, 44
 - <table>, 31, 183–187, 194
 - <td> (table data), 31, 187–188, 193
 - <th> (table header), 31, 187–190, 193
 - <tr> (table row), 31, 198–199
 - vertical alignment, 193
- <td> (table data) tags, 31, 187–188, 193
- templates. *See also* library items; SSIs
 - child pages, 146–149
 - editable regions, 142, 144–146
 - previewing completed files, 142–143
 - versus* server-side includes, 164
 - updating, 149–152
- testing servers, 350–352
- text areas, forms, 304, 306, 319–320
- text fields, forms, 304, 306, 308–309
 - field sets, 312–313
 - password, 313–315
 - Spry validation, 309–312
- <textarea> (multiline text input) tags, 31
- text/type
 - alt (text) attribute, 210
 - changing with behaviors, 271
 - CSS, 46–47
 - cascade theory, 47–48
 - class attributes, 65
 - descendant theory, 50–51
 - font groups, 110–112
 - font sizes, 112–113
 - HTML
 - body, 44
 - fonts, 44
 - headings, 44
 - margins, 44
 - table cells, 44
 - table headers, 44
 - versus* HTML-based formatting, 41–42
 - ID attributes, 65
 - inheritance theory, 48–50
 - multicolumn elements, 66–67
 - multiple elements, 64
 - placeholders, 101
 - shadows, 66–67
 - specificity theory, 51–53

- CSS3
 - multicolumn elements, 66–67
 - shadows, 66–67
- finding/replacing, 197–200
- fixed-width *versus* em-width containers, 259
- headings, 173–175
- importing, 170–171
- indents, 178–181
- lists, 175–178
- semantic structures, 172–173
- spell checking, 195–197
- tables
 - captions, 194–195
 - cells, 187–189
 - column width, 189–191
 - creating, 182–184
 - inserting from other sources, 191–193
 - styling, 185–187
 - vertical alignment, 193
- <textarea> (multiline text input) tags, 31
- <th> (table header) tags, 31, 187–190, 193
- 3D effect, backgrounds, 113–114
- TIFF file format, 215
- <title> tags, 21, 25, 31
 - search engines, 26
 - template editable regions, 144
- toolbars, adjusting, 13
- <tr> (table row) tags, 31, 198–199
- <track> (text tracks in media players) tags, 34
- trademark entity, 32
- transparency and translucency, 66–67
- true color, 207
- type. *See* text/type

U

- (unordered list) tags, 31, 98
- updating images, 230
- Use Alternative FTP Move Method option, 392
- Use FTP Performance Optimization option, 392–393
- Use IPV6 Transfer Mode option, 392
- Use Passive FTP option, 392–393
- Use Proxy (FTP) option, 392

V

validation, forms, 307, 309–312

vector graphics, 204

vertical alignment, tables, 193

video

with Adobe Flash, 288–289, 294–297

basics, 288

with HTML5, 288, 293–294

<video> tags, 34, 294, 296

View menu, toolbar selection, 13

W

W3Schools/other resources

browser statistics, 44, 73

Dreamweaver Marketplace & Exchange,

behaviors, 270

HTML, 29

elements list, 35

local web server configuration, 349

operating systems, 73

screen resolutions and sizes, 73

SQL (Structured Query Language), 362

SVG graphics, 204

web form mail scripts, 328

WAMP/MAMP web servers, 349

web animation

with Adobe Flash, 288

basics, 288

with HTML5, 288, 290–293

Web Distributed Authoring and Versioning

(WebDav), 390

Web Folders, 390

web servers

ASP (Active Server Pages)

Dreamweaver support, 4

.html extension, 357

web form mail scripts, 328

web server models, 348–349

ColdFusion

Dreamweaver support, 4

.html extension, 357

RDS (Remote Development Services), 390

web form mail scripts, 328

web server models, 348–349

local servers, 349

PHP (PHP Hypertext Processor), 349

data display, 367–368

database connections with MySQL, 354–358

Dreamweaver support, 4

.html extension, 357

recordsets, 362–367

testing servers, 350–352

web form actions, 324–325

web form mail scripts, 325–328

testing servers, 350–352, 369

web video

with Adobe Flash, 288–289, 294–297

basics, 288

with HTML5, 288, 293–294

WebDav (Web Distributed Authoring and

Versioning), 390

WebM file format, 289, 294

webpages

behaviors

displaying multiple items, 376

displaying record counts, 379–380

go-to-detail-page, 383

hiding paging controls, 378–379

record paging, 377–378

repeat regions, 369–370

data display, 367–368, 371–372

testing servers, 369

design considerations, 75–76

detail pages, 384–386

layout sketches, 75–77

master/detail page sets

displaying images, 374–376

master pages, 372–374

styling dynamic data, 380–382

recordsets, 362–366

thumbnails, 75

wireframes, 77–78

web-safe colors, 207

websites

design concerns

Adobe Flash, lack of support for, 74

browsers commonly used, 73

customers' access methods and usage, 73–74

desktop, laptop, and mobile device

challenges, 73–74

- purpose of websites, 72
 - W3Schools statistics, 73
- dynamic content
 - Ajax, 336
 - basics, 336
 - database applications, 352–354
 - database connections, 354–358
 - HTML data, 338–342, 344–345
 - Spry data, 338–344
 - Spry data types, 340
 - XML data, 345–347
- local *versus* remote sites, 78
- previewing files, 80–81
- site setup, 78–79
- Welcome screen, 79–80
 - creating/opening documents, 79–80
 - disabling, 80
- width
 - of objects, 56–57
 - of pages, 93–94
- WMF file format, 204
- Workspace menu, 4
 - prebuilt layouts, 11
 - saving custom workspaces, 13
- workspaces
 - customizing, 13
 - panels
 - available, 7
 - docking, 10
 - floating, 8
 - grouping, 9
 - minimizing/expanding, 7–8
 - reordering panel tabs, 9
 - repositioning panels, groups, and stacks, 9
 - stacking, 10
 - standard grouping, 7
 - prebuilt layouts
 - accessing, 11
 - Classic, 11
 - Coder, 12
 - Designer, 12

- World Wide Web Consortium (W3C)
 - HTML 4 default style sheet, 43
 - HTML5, 32, 35
- Wyke-Smith, Charles, 68
- WYSIWYG (What You See Is What You Get), 4–5

X

- XAMPP web servers, 349
- XML (Extensible Markup Language), 204
 - Ajax, 336
 - databases, 338
 - Spry data, 345–347

Y

- Yank, Kevin, 328

This page intentionally left blank

LEARN BY VIDEO

The **Learn by Video** series from video2brain and Adobe Press is the only Adobe-approved video courseware for the Adobe Certified Associate Level certification, and has quickly established itself as one of the most critically acclaimed training products available on the fundamentals of Adobe software.

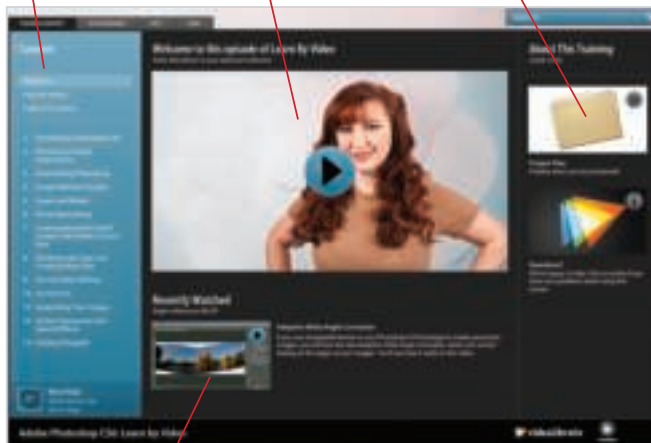
Learn by Video offers up to 15 hours of high-quality HD video training presented by experienced trainers, as well as lesson files, assessment quizzes, and review materials. The DVD is bundled with a full-color printed booklet that provides supplemental information as well as a guide to the video topics.

For more information go to www.adobe.com/learnbyvideo

Table of Contents never more than a click away

Up to 15 hours of high-quality video training

Lesson files are included on the DVD



Video player remembers which movie you watched last

Watch-and-Work mode shrinks the video into a small window while you work in the software



Titles

Adobe Photoshop CS6: Learn by Video: Core Training in Visual Communication
ISBN: 9780321840714

Adobe Illustrator CS6: Learn by Video
ISBN: 9780321840684

Adobe InDesign CS6: Learn by Video
ISBN: 9780321840691

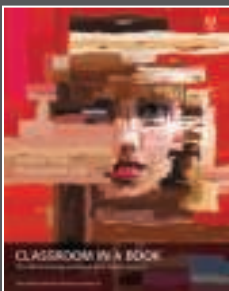
Adobe Flash Professional CS6: Learn by Video: Core Training in Rich Media Communication
ISBN: 9780321840707

Adobe Dreamweaver CS6: Learn by Video: Core Training in Web Communication
ISBN: 9780321840370

Adobe Premiere Pro CS6: Learn by Video: Core Training in Video Communication
ISBN: 9780321840721

Adobe After Effects CS6: Learn by Video
ISBN: 9780321840387

This page intentionally left blank



The fastest, easiest, most comprehensive way to learn **Adobe® Creative Suite® 6**

Classroom in a Book®, the best-selling series of hands-on software training books, helps you learn the features of Adobe software quickly and easily.

The **Classroom in a Book** series offers what no other book or training program does—an official training series from Adobe Systems, developed with the support of Adobe product experts.

To see a complete list of our Adobe® Creative Suite® 6 titles go to www.peachpit.com/adobecs6

Adobe Photoshop CS6 Classroom in a Book
ISBN: 9780321827333

Adobe Illustrator CS6 Classroom in a Book
ISBN: 9780321822482

Adobe InDesign CS6 Classroom in a Book
ISBN: 9780321822499

Adobe Flash Professional CS6 Classroom in a Book
ISBN: 9780321822512

Adobe Dreamweaver CS6 Classroom in a Book
ISBN: 9780321822451

Adobe Muse Classroom in a Book
ISBN: 9780321821362

Adobe Fireworks CS6 Classroom in a Book
ISBN: 9780321822444

Adobe Premiere Pro CS6 Classroom in a Book
ISBN: 9780321822475

Adobe After Effects CS6 Classroom in a Book
ISBN: 9780321822437

Adobe Audition CS6 Classroom in a Book
ISBN: 9780321832832

Adobe Creative Suite 6 Design & Web Premium Classroom in a Book
ISBN: 9780321822604

Adobe Creative Suite 6 Production Premium Classroom in a Book
ISBN: 9780321832689

AdobePress



WATCH READ CREATE

Unlimited online access to all Peachpit, Adobe Press, Apple Training and New Riders videos and books, as well as content from other leading publishers including: O'Reilly Media, Focal Press, Sams, Que, Total Training, John Wiley & Sons, Course Technology PTR, Class on Demand, VTC and more.

No time commitment or contract required! Sign up for one month or a year. All for \$19.99 a month

SIGN UP TODAY
peachpit.com/creativeedge

creative
edge