ADOBE® DREAMWEAVER® CS5

CLASSROOM IN A BOOK®

Instructor Notes

www.adobepress.com

Adobe

Adobe® Dreamweaver® CS5 Classroom in a Book

© 2010 Adobe Systems Incorporated and its licensors. All rights reserved.

If this guide is distributed with software that includes an end user 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, 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.adobepress.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: Lynn Grillo, Clint Funk
Copyeditors: Darren Meiss and Scout Festa
Compositor: WolfsonDesign
Indexer: Jack Lewis
Media Producer: Eric Geoffroy
Cover Design: Eddie Yuen
Interior Design: Mimi Heft

Book:
ISBN-13: 978-0-321-70177-0
ISBN-10: 0-321-70177-1

Instructor Notes:
ISBN-13: 978-0-132-48139-7
ISBN-10: 0-132-48139-1

INSTRUCTOR NOTES

Introduction

The *Adobe Dreamweaver CS5 Classroom in a Book* course provides students with techniques, tips, and solutions for using HTML, CSS, and Adobe Dreamweaver CS5 software. These instructor notes are intended to complement the information in the *Adobe Dreamweaver CS5 Classroom in a Book* and to provide insights and behind the scenes information specifically for classroom instructors. The information in these notes is organized to follow the sequence of instructions in each lesson.

About the workbooks

Each student in the class should have an individual copy of *Adobe Dreamweaver CS5 Classroom in a Book*. Students will use this book as you guide them through the lessons. You can buy copies of *Adobe Dreamweaver CS5 Classroom in a Book* for your students or you can refer then to a local bookseller to purchase the book.

For additional information, call Pearson Education at 800-922-0579.

Course strategy

This book consists of 17 lessons. If you complete one or two lessons per week, students will have time at the end of the course term to work on their own to construct a complete site using the Dreamweaver tools.

The lesson exercises are designed to introduce students to the Dreamweaver tools and functions, as well as important concepts in web design.

While performing the exercises, the students create a complete website for a fictitious community association and learn how to use and modify Dreamweaver's built-in CSS layouts.

You can evaluate each student based on his/her weekly completion of the lesson exercises. Requiring a student to design and build his/her own original website is a good culmination activity and could serve as the final exam.

Managing student projects

One way to simplify file storage and cleanup in the classroom is to instruct students to create a main folder on their hard disk and name it [*Student's Name*]-**webs** that will contain all their websites. Then, create a **DW-CIB** folder in this student folder to contain all the lesson content. Keeping all the files in this folder will make cleaning up the files easier when the course is complete.

Two methods for completing the course

The course can be completed in two different scenarios. The recommended scenario requires the student to build an entire website by completing the lessons in order. Lessons must be completed in sequence to build the necessary files for each subsequent lesson. This method simulates the scenario an actual web designer experiences when building a site from scratch. However, this method can be difficult and prone to error, because it relies on the successful completion of each page, component, and asset as the student builds his/her site. Incomplete or incorrectly built components from one lesson may adversely affect a subsequent lesson or actually cause it to fail.

In this event, a second method for completing the course is offered. It is referred to as the "jumpstart" method. The jumpstart method treats each lesson folder as if it is a complete website that includes all pages and assets needed to complete the specific requirements or goals of the lesson.

This method is also a good way to repeat a specific lesson or topic with which a student may be having difficulty. When following the jumpstart method, pages and assets from one lesson folder should not be used in another, unless so directed by the instructions within the lesson itself.

Using the jumpstart method, each lesson folder will be defined in Dreamweaver as a standalone website. Therefore, at the end of the course the student may have defined 17 websites, one for each lesson.

Additional resources

Adobe Dreamweaver CS5 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, refer to the following 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. Whether you're looking for a code sample or an answer to a problem, have a question about the software, or want to share a useful tip or recipe, you'll benefit from Community Help. Search results will show you not only content from Adobe, but also from the community.

With Adobe Community Help you can:

- Access up-to-date definitive reference content online and offline
- Find the most relevant content contributed by experts from the Adobe community, on and off Adobe.com
- Comment on, rate, and contribute to content in the Adobe community
- Download Help content directly to your desktop for offline use
- Find related content with dynamic search and navigation tools

If you have any Adobe CS5 product, then you already have the Community Help application. To invoke Help, press F1 or choose Help > Dreamweaver Help. This companion application lets you search and browse Adobe and community content, plus you can comment on and rate any article just like you would in the browser. However, you can also download Adobe Help and language reference content for use offline. You can also subscribe to new content updates (which can be automatically downloaded) so that you'll always have the most up-to-date content for your Adobe product at all times. You can download the application from www.adobe.com/support/chc/index.html.

Adobe content is updated based on community feedback and contributions. You can contribute in several ways: add comments to content or 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 http://community.adobe.com/help/profile/faq.html for answers to frequently asked questions about Community Help.

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

Adobe TV: http://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 includes three free curriculums 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 Forums: http://forums.adobe.com lets you tap into peer-to-peer discussions, questions, and answers on Adobe products.

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 CS5 product home page: www.adobe.com/products/dreamweaver.

Adobe Labs: http://labs.adobe.com gives you access to early builds of cutting-edge technology, as well as forums where you can interact with the Adobe development teams building that technology and 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. Check out http://partners.adobe.com/public/partnerfinder/tp/show_find.do for a directory of AATCs.

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

Lesson 1: Customizing Your Workspace

Lesson 1 exposes students to the Dreamweaver program interface and familiarizes them with program commands and conventions. Students experiment with different standard workspaces and learn how to create a custom keyboard shortcut.

If time permits, you can conduct this lesson as a hand-on exercise to familiarize students with the Dreamweaver interface.

Review questions

1. Why would you use Code view?
2. What is the difference between grouping and stacking panels?
3. What are the purposes of the HTML and CSS modes in the Property inspector?

Review answers

1. Code view gives you direct access to the HTML code for review and editing.
2. When panels are grouped, only one panel is visible at a time. When panels are stacked, all panels can be seen at the same time.
3. The HTML mode provides access to HTML-based tagging and attributes for page content. The CSS mode enables you to access, create, and edit CSS formatting for page content.

Lesson 2: HTML Basics

Students explore HTML theory and application. They learn how to create HTML code and a basic web page by hand in a text editor, exposing them to the language and basic coding techniques. They re-created the same web page in Dreamweaver. By learning how to write HTML by hand, the students can easily compare both methods of code creation and get a better understanding of the productively gains they will experience using Dreamweaver.

Students also review the history of HTML and its future. The lesson also includes a table of the most frequently used HTML tags and important background information. This lesson is an essential foundation for all web design and development.

Review questions

1 What are the advantages of using Dreamweaver over hand-coding?

2 What tag can be used to apply formatting inline, within another element?

3 How is an entity used in HTML?

Review answers

1 Dreamweaver provides numerous productivity enhancements over hand-coding, such as code hints, a WYSIWYG interface, and debugging features to mention a few.

2 The `<span>` tag is intended for applying formatting inline.

3 An entity allows you to enter characters that are not mapped directly to the keyboard, such as ©, ® and ™.

Lesson 3: CSS Basics

Lesson 3 provides an overview of Cascading Style Sheets (CSS). Students should be aware that a movement began over 10 years ago to separate the structure and content of web pages from their formatting. CSS was the technology that sprang from that endeavor. This lesson gives students an essential understanding of this important language.

HTML vs. CSS formatting

Students compare and contrast side-by-side formatting applied via HTML and CSS. This exercise demonstrates the advantages of CSS-based formatting over HTML-based formatting.

CSS box model

An essential concept to CSS is the box model, for it is the basis for all web page formatting and construction today. To achieve a successful outcome for page design and development using CSS, students must understand how this model works and how to apply it. This concept is a difficult one for most people. You may want to create additional exercises to help develop this understanding.

HTML defaults

It is vital that students appreciate the concept that HTML exists complete and apart from CSS. HTML elements all possess built-in formatting that frequently must be overridden by CSS. This exercise describes how HTML default formatting can impact web design.

Formatting text

This exercise provides an in-depth exploration of CSS-based text formatting. It defines and describes the proper construction of CSS rules and how CSS formatting is applied and affected by cascade, inheritance, descendant, and specificity theories.

Formatting objects

In the past, most web page layouts were built using tables. These techniques have been replaced with layouts based on `<div>` elements formatted by CSS. This exercise takes the students through this concept step by step, illustrating this important concept.

Multiples, classes, and IDs, oh my!

CSS rules can be built by combining multiple tags (compound), and custom classes and ids to target formatting to a specific instance of an HTML tag or specific element in context. This concept is essential to a complete understanding of CSS and its pivotal role in web design and development today. Before proceeding to subsequent lessons, you may need to repeat these examples and test the students' understanding. Be prepared with your own examples of CSS-based formatting using compound rules, multiple classes, and ids.

Note: This concept is continuously reinforced throughout the subsequent lessons.

Review questions

1 What is a disadvantage of using HTML-based formatting?

2 What theory decides which formatting wins when two CSS rules collide?

3 What's the difference between margins and padding?

Review answers

1 HTML-based formatting must be applied to each element and page manually, which decreases productivity.

2 Specificity is the theory that decides which rule wins during a conflict.

3 Margins add space outside of the element border, padding adds space within the element border.

Lesson 4: Getting a Quick Start

In Lesson 4, students get up and running by being exposed to multiple web design concepts while creating a complete web page using a built-in CSS layout. As they complete this first page, they experience many aspects of the Dreamweaver interface.

Defining a Dreamweaver site

Key to all website development is setting up the basic site structure. Students define the main project site structure using folders they create and by copying content to the computer hard drive. This is an important step to perform for each site, now and in the future, so it should be emphasized here.

We recommend that a folder be created at or close to the root level of the student's hard drive, where they have full read-write permissions. This main folder will contain all student files and folders. Within this folder, students should create a site folder named **DW-CIB**. Regardless of which course method you select (sequential lesson order or jumpstart), all the lesson folders should then be copied into this location.

If you decide to use the recommended course method, the **DW-CIB** folder will become the site root folder. If you decide to use the jumpstart method, each individual lesson folder will be selected as a standalone root folder. At the end of the course the student will have approximately 17 websites, one for each lesson.

Dreamweaver maintains links and relationships between all the files in the site by identifying the site root folder. Although the students won't set up the connection to the remote site or testing server in this lesson, make sure they realize that the site setup dialog box is where they will do this task.

Using the Welcome screen

Discuss using the Welcome screen to access recently used documents and to create new documents. Remind students of how they can disable the screen. Be sure to remind them not to disable the screen during this class, so it will available later in the training.

Selecting a CSS layout

Students create their first new page using the New Document dialog box. Spend time exploring all the options for creating new documents before settling on the one-column, fixed-width layout that will be used for the new page. You may want to demonstrate some of the other layout options to the students to make sure that they understand the difference between fixed-width and liquid layouts.

You may also want to demonstrate the difference between the layouts by opening an example of each and enlarging the window in a browser to show students how a layout changes when based on percent compared with pixels. All the exercises in the book involve fixed-width designs, so students will benefit from time spent explaining the features of other Dreamweaver CSS layouts.

Saving a page

When saving a page, make sure to explain the process of identifying and locating the site root folder. Dreamweaver normally opens to the site root folder by default, but not always. With the exception of images, all HTML and CSS files should be saved to the site root folder.

Inserting and editing titles, headings, text, and images

Students will edit placeholders and create page content by hand and by using external text and image files. Text will be entered by typing and by copying and pasting from provided RTF and TXT files. Students will learn how HTML tags can apply basic formatting.

Modifying CSS styles

Once content is added to the page, students modify the page appearance: changing colors, fonts, size, and background colors. They have their first experience with the CSS Styles panel and other CSS features that they will use throughout the book.

Using the Property inspector

One of the most indispensible tools in Dreamweaver is the Property inspector, which provides instant access to HTML and CSS formatting for most types of page elements. In this lesson, the students are given their first hands-on experience

with the Property inspector. Lesson 1 explored the various features of the Property inspector, but you should re-emphasize these capabilities.

Explain how the Property inspector is used to apply heading or paragraph tags as well as custom classes and ids. Provide some examples of situations where students would need to select between the HTML and CSS modes of the panel.

Using Design and Code views

In Lesson 4, the students have their first experience switching between Design and Code views. Have the students experiment with the individual views and show them how Split view gives them the best of both worlds. Demonstrate how the text or content selected in one view is simultaneously selected in the other. Explain that when they add new text, images, or structural elements, they may need to update other views manually before the changes display. Updating the view can be accomplished by clicking the mouse in the specific view window.

Previewing pages in Live view and the browser

The lesson concludes by previewing the page in Live view within Dreamweaver and then within a browser. While Live view can save time, it doesn't substitute completely for seeing the page in a browser. Reinforce the concept that HTML pages don't exist until they are rendered in a web browser and that all web browsers might display pages differently from one another. The inconsistencies even extend to different versions of the same browser.

Review questions

1. Why should you select a default images folder when setting up a site?
2. Do you have to keep the Welcome screen visible?
3. What's the advantage of specifying more than one typeface in a CSS format?

Review answers

1. Using separate folders to organize file types makes managing the overall site easier. Dreamweaver will automatically move images to the default images folder for you.
2. No. The Welcome screen can be shut off permanently by selecting the Don't Show Again option.
3. A selected typeface may not be installed on the user's computer. By designating a list of fonts, you can control how the site is formatted.

Lesson 5: Creating a Page Layout

Lesson 5 walks students through the various steps needed to develop their own website design from scratch.

Basics, thumbnails and wireframes

Students are given a broad overview of all the concepts that must be considered in a comprehensive site design. Reinforce the notion that a website design must consider multiple aspects, including site purpose, customer demographics, and accessibility.

Have the students choose a basic site theme, such as entertainment, information, sales, or other and then conduct a search for two or more sites serving this need, using Google or Yahoo. Have them analyze the design and content of each site and compare and contrast them. Finally, have the students, individually or as a whole, rate the site's success or failure at achieving their goal.

Give the students sheets of blank paper. First, have them make an inventory of pages and page components they will need for their site. (Note: The list may vary from page to page.) Then, have them sketch out their own design concepts, incorporating the previously listed components. Have each student produce at least two variations on their basic designs. Discuss the advantages and disadvantages of each concept.

If there is time, you may want to demonstrate how to use Adobe Fireworks or Adobe Photoshop to create a mockup or proof-of-concept of a student's design. Since the students may not have the skills or understanding of these products, you may want to perform this demonstration yourself.

Previewing your completed file

The student will open a file saved within the lesson folder that has been completed according to goals within the lesson or exercise.

In many exercises, these completed files are provided, usually saved within the specific lesson folder, to provide an example of the goal of the lesson or exercise. Briefly describe the concepts and goals that will be taught in the lesson while the file is open. You may want to leave this file open during the entire lesson so the student can refer back to it. In most cases, the completed file should not be used in place of the final page.

Note: To prevent the completed pages from being reformatted by changes to the CSS or inserted images, these pages may not be linked to the style sheets or other desired assets.

Modifying an existing CSS layout

To save time, students will select and modify a built-in CSS layout. Explain that this is a recommended starting point for all beginning web designers, since the basic structure and CSS is already built and tested. If there is time, explore other CSS layouts and provide suggested uses for some.

Working with content and formatting

The students work step by step through the process of building a complete web page design. They learn how to insert a background graphic and a new `<div>` element. They modify the basic page structure by changing the width of an existing CSS rule. Refer back to what they learned in Lesson 3 regarding formatting objects with CSS and how it relates to the current layout.

When the students modify the vertical menu, have them switch to Code or Split view to examine the structure of the code. If necessary, refer them back to Lesson 2, regarding the basic structure of an HTML unordered list and how it relates to the menu structure.

Inserting placeholder content

Students start to build their site template by inserting placeholder elements for pictures and text. Explain how these elements allow them to structure and style the page even before the content is ready for the site.

Modifying the footer

Students add copyright information in the footer `<div>`. This is a good moment to discuss copyright, fair-use, and other content restrictions.

Checking browser compatibility

Reinforce the previous concept of browser display differences and incompatibility issues. Refer back to Lesson 2 and Lesson 3.

Review questions

1. How does Dreamweaver treat AP divs differently than ordinary divs?
2. How can you create a hyperlink placeholder?
3. What is an image placeholder?

Review answers

1. When an AP div is inserted, Dreamweaver assigns it a custom id and creates a corresponding CSS rule.
2. Insert the hash (#) mark in the hyperlink URL field to create a hyperlink placeholder.
3. An image placeholder is an element that Dreamweaver can insert into the layout to hold the spot for an image that may not be available at the moment.

Lesson 6: Working with Cascading Style Sheets

In Lesson 6, students delve into Dreamweaver's style tools and explore CSS more deeply. Students learn how to create and apply CSS rules, create descendant selectors and custom classes, attach an external style sheet, and create a separate style sheet for print applications.

Working with the CSS Styles panel

Students use the CSS Styles panel to create and edit CSS rules. Students will also learn how to re-order CSS rules within the panel. Reinforce the concepts of cascade, inheritance, descendant, and specificity theories from Lesson 3 and how the order of rules within the panel can affect how they format the page content.

Creating new CSS rules

Students will add new CSS rules throughout the book. Most of the new rules will use descendant selectors. Students will enter specifications for the rules using a variety of methods. Reinforce good technique and careful attention to detail. Be sure the students identify the proper fields they need to access and understand the various units of measurement offered by Dreamweaver and supported in HTML and CSS. At the appropriate time, discuss the advantages and disadvantages of using fixed and/or relative measurements.

Building and formatting an interactive menu

This exercise is a good opportunity to examine how to use modern CSS techniques to create an interactive navigation bar. Take this moment to examine the interactive nature of the vertical menu and compare it to the horizontal menu as you build and format it. Differentiate the default nature of HTML hyperlinks and the ability to control and modify these behaviors using CSS.

Modifying hyperlink behavior

Discuss the default hyperlink behaviors with the students and emphasize how these behaviors, and others, can be modified by CSS. Remind the students of the four hyperlink pseudoclasses they just learned.

Creating faux columns

This exercise is a good opportunity to discuss the limitations of HTML and CSS for creating custom and/or pixel-perfect layouts.

Creating external style sheets for multiple media

The power of CSS is in the ability to format an entire website from one or more external CSS files. Students learn how to move internal CSS rules to a separate external style sheet and then how to convert a style sheet for screen media to one for print or other applications.

Review questions

1. What three ways can CSS styling be used to format content?
2. What happens if you change the id assigned to an AP div?
3. How is an AP div different from ordinary divs?
4. What is a pseudoclass?
5. Describe the default behaviors of a hyperlink.

Review answers

1. CSS styling can be applied as an external linked file, as an embedded style sheet, or inline in the element itself.
2. Dreamweaver will automatically rename the corresponding CSS rule.
3. AP divs are positioned at specific coordinates on a web page and are assigned a CSS rule as soon as they are created. If you change the ID on the AP div, Dreamweaver will change the name of the CSS rule to match it.
4. A pseudoclass is a predefined state of a specific HTML element that can be formatted using CSS rules, such as `a:link`, `a:visited`, `a:hover`, and `a:active`.
5. Hyperlinks are formatted in blue and appear with an underscore by default. If you move the mouse cursor over the hyperlink, the browser displays a hand icon. If the page contains a link that has already been visited, it will be displayed in a different color than links that haven't been visited.

Lesson 7: Working with Templates

In Lesson 7, students learn how to increase productivity by creating and using templates, library items, and server-side includes.

Creating a template from an existing layout

In this exercise, students save their existing layout as a site template. The template file is automatically moved into a **templates** folder and given the extension .dwt. Point out how Dreamweaver rewrites any of the existing hyperlinks since the file is now stored in a subfolder. Reinforce the concept that pages built from this template can be updated automatically by Dreamweaver whenever any changes are made to the file and saved.

Inserting editable regions

Review the basic design of the site pages based on the completed file provided in the lesson. Identify the editable regions and the areas that are locked and the reasons why. Note that the entire sidebar is editable and what limitations may be imposed by that design. List some alternative ways of designing the page that would accommodate the vertical menu within the template and allow it to be updated on each child page.

Producing child pages

All pages for the website project will be developed from child pages of this template. Be sure to demonstrate that site templates can be accessed from the Assets panel as well as the New Page dialog box.

Updating a template

Changes made to both editable and locked regions of the template are compared in this exercise. It is important for students to understand that the template will update only areas within locked regions.

Using Library items

Library items have all the benefits of templates, allowing users to create website components that can be deployed to multiple pages but still updated automatically. Be sure to explain that Library items can be made from any type of content or code and that they do not need to be interactive or based on a Spry widget.

It is important to point out that when updating Library items site wide all affected pages must be saved and uploaded, too. Emphasize that the code for the Library items is wholly contained in the dependent page and that the Library items themselves and their folder do not need to be uploaded to the remote site.

Using server-side includes

Students will explore using server-side includes (SSI). SSIs are more powerful than Library items because they load dynamically when the containing page is called by the browser. That means, to update the content of all instances of the component, only the SSI itself has to be uploaded to the remote site. Point out that SSIs have a few disadvantages, too:

- They have to be loaded by a compatible server application that may not be available on your local computer, making testing locally difficult if not impossible.
- You have to use a file extension that supports SSIs, such as .asp, .cfm, .php, or .shtml.
- Loss or damage of the SSI on the server will eliminate the content from all dependent files.

Review questions

1. What is an editable region?
2. How many editable regions does a template have automatically?
3. How are child pages updated?
4. Do you have to update child pages each time you save a template?
5. What is the main advantage of a server-side include?

Review answers

1. An editable region is a user-designated area of the web page that is not locked and not updated by the template, where you can insert unique content on child pages.
2. A template starts off with no editable regions.
3. A child page can be updated automatically each time the template is changed and then saved. Or, you can manually update child pages.
4. No. You don't have to update each time you save changes to a template.
5. Server-side includes are loaded each time a page is called. So, to update the entire website only one file needs to be uploaded.

Lesson 8: Working with Text, Lists, and Tables

In Lesson 8, students learn how to add text content to their child pages, including HTML lists and tables.

Importing and formatting text

Students learn how to import text content from an RTF file and to apply HTML heading and paragraph tags using the Property inspector. Explain to the students the hierarchy of HTML headings (<h1>, <h2>, <h3>, and so on) and how the number and application of these headings can affect page ranking by search engines.

Demonstrate how Dreamweaver's ordered and unordered list buttons actually create and apply specific HTML code markup to the text. Have the students experiment applying and removing the list formatting and examining the markup in Code or Split view. You may want to assign an additional activity to make a nested list to see how the markup is modified.

A goal of this lesson is to indent some text content. There are several methods for doing this. Some web designers use the <blockquote> tag to indent the text for the **about_us.html** page. Discuss how the <blockquote> tag has been used incorrectly to indent text in the past, but how it has a specific semantic value for identifying quoted material. If you want to adhere to web standards, it shouldn't be used simply to indent text. Evaluate the merits of other methods for indenting text, such as using a custom class or a tag and work the discussion around to web standards and the benefits or detriments in supporting them.

Creating and styling tables

Students learn how to create and format the size and structure of a table. Explain how CSS styling should be used in most instances, but that HTML specifications are still appropriate in limited applications. This is a good time to point out that Dreamweaver doesn't provide all CSS properties within its dialog boxes, but that you can specify any valid property and value in the CSS Style panel or directly within the CSS file itself. You may want to mention that Dreamweaver code hints work for all valid properties and values.

Spell checking web pages

The content imported in this lesson contains several words misspelled purposely. Alert the students to not correct these errors manually in this exercise. Mention that misspelled words detract from the professionalism and credibility of a website and its content. Students will learn how to use Dreamweaver's spell check feature to locate and correct such errors.

Finding and replacing text

The Find and Replace dialog box is one of the most powerful features of Dreamweaver. It should be stressed that it can be used for more than finding and replacing simple phrases. It can perform complex alterations to code and content. Emphasize its ability to search and replace text, code, and even white space across a page, selected pages/folders, and/or the entire site.

Review questions

1. Can Dreamweaver open an RTF file?
2. Why do you need to put extra paragraph returns between paragraphs before copying and pasting text from other programs to Dreamweaver?
3. How many `<h1>` tags should be used on each page?
4. To create a text indent would you use the `<blockquote>` tag?
5. If you want the entire column of a table to display at a specific width, do you need to apply a custom class to the entire column?
6. Can you add custom terms to the Dreamweaver dictionary for spell-checking?

Review answers

1. No. Dreamweaver can't open RTF files. But it will launch a compatible program when you double-click on a file.
2. If you copy single paragraphs from most programs, they will be inserted into Dreamweaver with break characters `<br />` instead of paragraph `<p>` elements.
3. To following semantic best practices, only one `<h1>` tag should be used per page.
4. No. The `<blockquote>` tag has a specific semantic value so, to comply with web standards, it should only be used to format quoted material.
5. No. Since the entire table column must be the same width, a class only needs to be applied to one cell within the column.
6. Yes. Custom terms can be added while spell checking.

Lesson 9: Working with Images

In Lesson 9, students learn essential information about raster and vector image basics. Then they add images to their pages and learn how to position those using CSS classes. Students are also introduced to Adobe Bridge, Fireworks, and Photoshop.

Reviewing web image basics

This section is devoted to developing an essential understanding of web images. The most important concept to emphasize is the relationship between image size, resolution, and download speed. Explain how increasing image size creates a larger file and that large files take longer to download. You may want demonstrate how selecting different color palettes can decrease image size, but adversely affect quality.

Inserting images

Students explore multiple methods for inserting raster images into their layouts. Emphasize the use and principles behind writing good Alt text (alternative text). Two helpful resources are "Writing good Alt text" (**http://tinyurl.com/5nawce**) and "How to Write Good Alt Text for the Images on Your Web Page" (**http://tinyurl.com/29ghvdg**).

Adjusting image positions with CSS classes

This exercise compares and contrasts using HTML positioning attributes with CSS custom classes. Explain how CSS allows you to control the placement of images across the entire site at once, while HTML attributes must be changed one by one directly on the page itself and don't offer any fine control.

Working with the Insert panel

The Insert panel is a productivity tool that has been available in Dreamweaver for a long time. It gives users instant access to important commands for inserting a variety of code and components using a single click. It provides seven preset categories and one that's customizable. Take time to explore all the categories and demonstrate how the students can add specific selected tools to the Favorites category as desired.

Using Adobe Bridge and Photoshop Smart Objects

Adobe Bridge is a separate application that enables users to view the contents of various proprietary and open-source file types. It simplifies the workflow between all the Creative Suite applications. In this exercise, students use Bridge to access folders within the site and choose images to insert into the web page.

Students are exposed to using Photoshop Smart Objects. Students learn that Smart Objects maintain a connection to a copy of the original Photoshop image. They allow you to use the same image multiple times at different sizes and configurations and then resize the image live within the layout as desired.

Copying and pasting and using drag and drop with images

Students explore using multiple techniques for inserting images including copy and paste and drag and drop. Copy and paste allows you to move individual layers from a Photoshop file to a Dreamweaver layout. It's probably a good time to discuss the advantages and disadvantages of these various workflows. For example, the speed and convenience of these methods may not outweigh the availability of the full toolset of Photoshop or Fireworks. Caution students that drag-and-drop placement may insert images in the wrong position in the code.

Optimizing images with the Property inspector

Dreamweaver has built-in graphics tools that can be accessed via the Property inspector. These tools give the user the ability to perform specific image editing tasks such as cropping, sharpening, and resampling directly within Dreamweaver. You may want to review the basic concepts of raster images described at the beginning of the lesson. Remind the students of the connection between image size, resolution color space, and display quality.

Review questions

1. What vector-based image format is supported on the web?
2. What effect does increasing compression on a JPEG image have?
3. Can you insert a 300 ppi image on your web page?
4. Can you copy and paste a multi-layer Photoshop file?
5. What does it mean if there is a circular arrow in the Property inspector by the image Width and Height values?
6. What is a *hotspot* used for in Dreamweaver?
7. If you reduce the size of an image using the Resample tool, can you restore the image to its original size later?

Review answers

1. Scalable Vector Graphics (SVG) is the only vector graphic format supported on the web.
2. In most cases, increasing image compression decreases file size. But, it may also decrease image quality.
3. Yes. But the image would be much larger than necessary. The maximum resolution of the web is 72 ppi.
4. You can only copy one layer at a time from Photoshop.
5. A circular arrow indicates that the displayed dimensions of the selected image are not same as the actual dimensions.
6. A hotspot is used to designate an area on an image as hyperlink.
7. The resample feature permanently changes the quality of an image. The image can be resized, but it will probably display at unacceptable quality.

Lesson 10: Working with Navigation

In Lesson 10, students work with relative and absolute hyperlinks to create both internal and external site navigation. Students learn how to insert and format a Spry Menu Bar.

Hyperlink basics

Students review the basics of hypertext navigation to learn the differences between internal and external links as well as absolute and relative links. Students learn how to build links as well as what attributes and values can be applied to them.

Creating text and image-based hyperlinks

Students learn how to apply hyperlinks to both text and images. Emphasize to the students that text- and image-based hyperlinks use the same basic code structure.

Setting up e-mail links

Students learn how to create a link to generate a custom e-mail. Be sure to discuss with students the limitations of this method and how it is not recommended because it relies on client-side software that may not be installed as well as user-based actions to send the e-mail.

Targeting page elements

The default behavior of a hyperlink is to replace the existing browser content with the targeted selection. Students learn how to change this default behavior so that a new window opens to load a specific web page. Take time to remind students why they should use this method to open a separate window for external links. Suggest other times when this method would be appropriate.

Inserting Spry menu bars

Navigation menus are universally created now using CSS and various HTML components, typically unordered lists. In this exercise, students are led through the process of inserting and formatting a Spry Menu Bar that will be used as a main navigation component. The menu inserted in this method replaces the existing menu created by Dreamweaver's CSS layout that is not based on a Spry widget.

Take time to review the other available Spry widgets and how they can be used. You may want to demonstrate some of the other components that will not be used within the site. A good suggestion is to show the students the Spry widget gallery at **http://tinyurl.com/2553nke** to demonstrate some of the possibilities.

Inserting Spry menus as Library items

In the current layout, the Spry menu is part of the editable region of the template and will not update automatically when the menu or the rest of the template is changed. Discuss the limitations of keeping the Spry menu in this position. Suggest alternative designs that could address this problem. The fix selected in the exercise is to convert the Spry menu to a Library item and re-insert it. This is not an ideal solution, but it allows the student to see first hand the limitations of editable regions and contemplate their own fixes.

You may want to examine the advantages of re-creating the sidebar to move the menu into a locked area of the template. Discuss the pros and cons of this solution.

You may want to contrast and compare the previous method to using a server-side include and the advantages of that method over using a Library item or even the locked region of a template. Remind students that their local browser may not support server-side includes and that they will be difficult to test if that is the case.

Checking your page

This exercise allows you to demonstrate some of the other important features in Dreamweaver that help you maintain a website. For example, Dreamweaver has a full set of features that can check links, browser compatibility, and accessibility among other attributes that web designers and developers should master. Be sure that the students know how to use Adobe Online Help for any errors found in the browser compatibility check.

Review questions

1. What does URL stand for?
2. What's the difference between an absolute and relative link?
3. Why should e-mail links be discouraged?
4. Can you apply hyperlinks to images?
5. How can you create navigation to a specific part of a page?
6. What role does an ID attribute play in hyperlink navigation?
7. How are server-side applications better than client-side applications?

Review answers

1. URL stands for Uniform Resource Locator.
2. Absolute links spell out the entire URL address of a link. A relative link only contains the name of the file name and the path information needed to load the image in relationship to its location to the web page itself.
3. E-mail links depend on software that is installed on the user's computer.
4. Yes. Just select the image and enter the URL info into the Link field in the Property inspector.
5. Insert a named anchor at the position you want to jump to. Then, create the hyperlink using the page name (if necessary) and the name of the anchor with a hash (#) in front of it.
6. Since an ID attribute should be unique, you can use it in the same way as a named anchor.
7. Server-side applications rely on software and scripting that is provided by your own server. Since you can't control what software and services are installed on the users' computers, you can be more certain that a specific function will work as desired when using your own server-side application.

Lesson 11: Adding Interactivity

In Lesson 11, students learn about Dreamweaver behaviors and how to use them within a layout along with Spry widgets.

Learning about and working with Dreamweaver behaviors

This section gives the student a general background in what behaviors can do and how they work. Make sure that students know how to pick an appropriate trigger, such as `onClick` or `onMouseOver` and how to edit the parameters of any behavior they use.

Working with Spry Accordion widgets

Students will insert a Spry Accordion widget to add an interactive element to the site. Discuss the advantages of using Spry Accordions to insert large amounts of contents to a page without requiring users to scroll up and down long pages.

Review questions

1. Name two available behaviors?
2. Name two possible triggers for a behavior?
3. When you apply a Swap Image behavior to an item, how do you get the original image to return?
4. What does the Preload Images options do?

Review answers

1. Some of the available behaviors are: opening a browser window, swap image, fading images, growing or shrinking graphics, and displaying pop-up messages, among others.
2. Behaviors can be triggered by: `onMouseOver`, `onMouseOut`, `onFocus`, and `onBlur`, among others.
3. You need to apply a Swap Image Restore behavior.
4. The Preload Images option downloads the images involved in a behavior when the pages is first loaded so that there is no delay in the performance of the behavior.

Lesson 12: Working with Flash

In Lesson 12, students learn how to incorporate Flash animation and Flash video in their site.

Adding Flash animation to a page

Dreamweaver offers a high degree of integration with Flash components. Flash animation is saved in a SWF file, which provides the means of inserting animation and powerful interactive applications and web interfaces into your site.

Adding Flash video to a page

When Flash video (FLV) is added to a web page, make sure students are aware that specific JavaScript files as well as the SWF installer are required and must be uploaded to the remote site along with the web page and the FLV files themselves. Don't forget to remind the students that they can test the video right inside Dreamweaver.

Review questions

1. What extension is used for Flash animation?
2. What's the advantage of using vector-based animation?
3. Can you edit the alternative content or add your own?

Review answers

1. Flash animation files are saved with the .swf extension.
2. Vector-based animation typically requires only a fraction of the disk space used by raster animation.
3. Yes, you can add your own content and even create custom formatting, if desired.

Lesson 13: Working with Forms

In Lesson 13, the students learn how to design an accessible form that will allow them to collect useful information from site visitors.

Learning about forms

This section describes general information that explains the purpose of a web-based form, its constituent elements, and the logistics involved in how data can be collected.

Inserting a form and form elements

In these series of exercises, the students insert a form and populate it with text fields, check boxes, radio buttons, and other form elements. Students also insert and configure Spry validation elements and learn how to make forms more accessible.

Students insert various types of text fields. Be sure to discuss the differences between text, password, e-mail, and validation fields. Point out that some settings are applied using HTML attributes while others are handled by Spry programming using JavaScript. Remind them that all the supporting files must be uploaded to the remote server. Guide students through writing sensible error messages for validation elements so that users clearly understand what data is being requested.

Long forms should be divided into logical structures contained in fieldsets. Encourage students to use Code and Split views when adding fieldsets to the form to ensure the correct elements are being selected.

Submitting forms

Student learn how to create a submit button and choose a form action. Discuss the disadvantages of sending the form data via e-mail. Explain how many people don't use a locally installed e-mail application, like Microsoft Outlook or Apple Mail. Instead, they use Internet-based applications, like Gmail or AOL.

A better plan is to use a server-based application. But, these are more difficult to implement and require access to the server and installed scripts and services running there. Remind the students that such applications will probably require administrative rights on the remote server and some form of e-mail or webmail service.

The students are led through the process of creating a server-side PHP-based e-mail application in a new page that will transmit the form data to a specified address.

Styling forms

Students learn that form elements can be modified using CSS as well as any other HTML elements. Custom styling is created for fieldsets and other form elements.

Review questions

1 What do text fields, check boxes, and radio buttons have in common?

2 Give a reason for using a hidden field in a form.

3 How do password fields differ from normal text fields?

4 Why should you enable the accessibility feature in Dreamweaver for forms?

5 Is the form action applied to the submit button?

6 What happens in some browsers when you try to load a page with active content?

Review answers

1 They are all created by the `<input>` tag.

2 You can use a hidden field to pass a predetermined piece of data to a dynamic application.

3 When a user types in a password field the input is masked by bullets or asterisks so it can't be read.

4 By enabling the accessibility features of Dreamweaver, you will be prompted to add labels and alternative text among other attributes to form elements to assist people with disabilities to use your website more easily.

5 No. The action is applied directly to the `<form>` element itself.

6 Some browsers block active content and require users to approve the operation of the element.

Lesson 14: Working with Online Data

In Lesson 14, students learn how to work with online data. This lesson guides students through using HTML and XML data sets to build dynamic Spry tables. Then, it walks the students through the workflow for developing applications using hosted databases.

Lessons 14 and 15 are certainly the most difficult within the book. Be aware that the instructor or the training facility must accomplish a great deal of prep work prior to attempting these lessons to achieve a successful outcome. A local web server must be installed properly and tested. Once the server is installed and working

correctly, back up the configuration so that you can restore it to this initial state for each subsequent class.

The book describes using a PHP server using a MySQL database, but the exercises can be directly adapted to both ASP and ColdFusion server models, too. This book does not supply direct support for any of the server models. To learn more about installing and configuring different server models check out the following URLs:

- ASP and IIS—**http://tinyurl.com/2yx46y** *or* **http://tinyurl.com/ybhfaef**
- ColdFusion—**http://tinyurl.com/2h8hey**
- PHP—**http://tinyurl.com/39p3gh**

Using XML and HTML data

The easiest way to develop dynamic content is to use Adobe's Spry framework for Ajax. It is a comprehensive solution to producing data-driven page content. The students walk through building dynamic tables based on HTML and XML data.

Choosing a server model

Take this opportunity to discuss the decision factors that go into choosing a server model. The factors involved in this decision include, but are not limited to:

- Host server type
- Budget/cost
- Types of applications desired
- Availability of support/training
- Pre-built components available
- Personal preference

Configuring a local web server

We recommend using the MAMP/WAMP server configuration for the training described in Lessons 14 and 15. This server is a complete turnkey solution for users who want to develop PHP/MySQL applications that works on both Windows and Mac computers and best of all, it's free. Be sure to discuss the limitations of the free server with students and help them assess their future requirements and determine at what time they would need to select a more robust enterprise solution.

Training centers that want to consider teaching these exercises using a ColdFusion server model, should download and install the latest version of ColdFusion from the Adobe website at **http://tinyurl.com/27clst5** and select the Developer Edition during the installation to enable a free unlimited version of the server intended for developers only.

Windows-based training centers may want to use the ASP/IIS server model since it is supported directly by Windows and easily installed and configured. Mac-based centers may think they have to avoid ASP/IIS, since it is incompatible with OSX. However, students should be informed that Mac users can still develop for ASP/IIS by using a Windows-based remote server as the testing server. Using this method, when Mac users try to test ASP pages Dreamweaver uploads the page and dependent components to the remote testing server first before displaying the data.

But no matter what server model you select, the server should be tested thoroughly before the class begins. Remember that all servers need to be started before they can be used. In some cases, the server will start automatically when the computer reboots. Otherwise, you will need to launch it manually. Make careful note of any needed port addresses or other settings for your specific server and walk students through this procedure to prepare them for their own server implementation.

Setting up a testing server

For the purposes of the exercises in Lessons 14 and 15, the local web server—installed earlier—will be used as the testing server. You can also use a remote server as the testing server, if you have access to one. Remind the students that they should check to see if the local or remote server actually supports the server model. The folder used for the testing site will also be used for the remote site in Lesson 15. But the students should be told that they don't have to use the same folders for both testing and remote servers. Also remind students that multiple servers can be set up and tested in the Site Setup dialog box.

Pay close attention to the path names for the testing servers. In the case of PHP and ColdFusion servers, a port address may need to be appended to the URL/server path for it to function properly. It is very important that you test your local and testing server configurations thoroughly before the class begins so you can derive the proper settings. It is a good idea to review these steps with the student, especially if you have any difficulties.

Building database applications

All your hard work pays off in this exercise as students access the server and create connections to the hosted database. A MySQL database is included in the lesson assets. It must be copied to the server and installed properly before the students can connect to it. A MySQL database appears nearly instantaneous in the database list once it is installed, so you may want to walk students through the installation process as a hand-on activity. An alternative would be to demonstrate the process only on the instructor's computer and install the database ahead of time on the students' computers.

On a WAMP configuration, the database folder **greendata** should be copied to the **WAMP\bin\mysql\mysql(version)\data** folder. On a MAMP server, it should be copied to **MAMP > db > mysql**. Other server models may require that the database be installed in different locations. Check with your IS/IT manager or web hosting service for specific details. In any case, the database connection should be thoroughly tested before attempting to use it in the classroom. Remind students that they may also need to obtain/define read/write permissions for the configuration if they're going to insert new data into the database via web-based applications.

Review questions

1. What's the advantage of using HTML tables for Spry data sets?
2. What attribute must be added to an HTML table before it can be used for a Spry data set?
3. Name at least two Spry data types?
4. What data type would you use to pass code elements directly to the web page?
5. Describe the format Spry does not recognize as a legitimate date?
6. How do you format Spry tables so that the rows alternate their formatting?
7. What are the advantages of using the MAMP/WAMP servers?

Review answers

1. HTML tables are static, so that multiple instances of such a table must be updated manually. Multiple instances of a Spry table will update automatically whenever the parent HTML table is changed.
2. An HTML table must be assigned a unique ID before it can be used to create a Spry data set.
3. Spry data types can be any of the following: String, Number, Date, or HTML.
4. To pass code from the data table directly to the web page, you should use an HTML data type.
5. Spry doesn't recognize dates written as: 1-1-2011.
6. Create two different classes and apply them to the Even and Odd class options in the Spry Data Set—Insert Table dialog box.
7. The MAMP/WAMP servers can be downloaded, installed, and operated for free for both Mac and Window-based systems.

Lesson 15: Building Dynamic Pages with Data

In Lesson 15, the students build pages that will be populated with dynamic content. It is recommended that you test all exercises several times before attempting to teach this lesson. There are many situations where these applications can fail. In troubleshooting an error consider at the following:

- Is the local web server installed and functioning properly?
- Is the testing server configured properly?
- Are the testing server path statements correct?
- Is the database installed correctly?
- Does the database appear in the phpMyAdmin start page?
- Did you save the file with the proper extension?
- Are there any outstanding steps specified in the Databases panel?
- Does the greendata connection appear in the Databases panel?
- When you reveal the contents of the greendata connection, does it display the expected list of tables?

Be aware that, in certain circumstances, simply adding the PHP extension to a Dreamweaver document won't necessarily display the database connection, established in Lesson 14, in the Databases panel. That's because the following line of code that establishes the database connection may be missing from the beginning of the file:

```
<?php require_once('Connections/Localhost.php'); ?>
```

Occasionally, this line can be inserted two or more times causing a different type of error. If you don't see this line, first try clicking the File Type link in the Databases panel, select the desired server model, and save the page. This should insert the necessary code. If the connection code is still not visible, try closing the page and reopening it. If the database connection still doesn't appear, you may need to re-create the database connection as described in Lesson 14.

Building pages with ASP, ColdFusion, or PHP

Be sure to emphasize the care students need to maintain throughout this entire lesson. Even the simplest typos can cause these applications to fail. For example, when selecting table cells or rows, it is vital that only the specified elements should be selected.

Remind the students that if anything unexpected happens, they should select Edit > Undo immediately and recheck the instructions carefully.

The most crucial step is the creation of the recordset. If this is successfully accomplished, the remainder of the lesson should proceed successfully. Do not gloss over errors or problems connecting to the database or making recordsets. These are difficult procedures and some mistakes should be expected. Use any errors as valuable teaching moments. Dreamweaver makes this process mostly pain free and this point should be emphasized.

It may be helpful to know that recordsets, once created, can be copied from one page to another. Simply select the existing recordset and press Ctrl-C/Cmd-C. Switch to another dynamic page in the same site, open the Bindings panel, if necessary, and press Ctrl-V/Cmd-V.

If a specific behavior doesn't seem to be working properly, the easiest solution may be to select it in the Server Behaviors panel and delete it. If an entire page seems to be responding badly, it may be easier to delete the entire page and start from scratch.

Tell students that sometimes when you edit or delete dynamic applications from a page, errant bits of code may remain from the deleted process and cause trouble. Dynamic code is often complex and hard for novices to read. It may be helpful to examine the code elements as they are added to give the students a better understanding of how much power is brought to the process by Dreamweaver.

Creating a master/detail page set

This exercise is the culmination of the dynamic application. It is also the most demanding of the students, requiring them to enter specific code elements by hand. As mentioned earlier, these steps are very prone to simple errors. It's recommended that you write out the desired path names on a white board or display them prominently in the classroom to ensure the students know what they should be typing. The dynamic links in this exercise can be edited in Code view, if necessary, or you may want to delete them and re-create them from scratch.

Be sure that the product images are installed on all students computers properly. If the images do not appear in the dynamic display, check the path statements created by the students. The name of the folder containing the images (**products** in this case) must be inserted before the PHP code that generates the image display. Remind the students that only the image name is stored in the database, not the path information. Explain that storing only the image name is an advantage allowing you to move or rename the image folder as needed. This may be a good time to walk students through the process of viewing the data that is stored in the database. You can use the Admin page of MySQL or simply review the test display in the Recordset dialog box.

If the application doesn't display text or image data, check the following:

- Is the database connection visible in the Databases panel?
- Does the database connection display the expected tables when you open it in the Databases panel?
- Is the recordset visible in the Bindings panel?
- When you open the recordset and test it, does it display data?
- Does the recordset test display the file names for the images?
- Is the page populated with the correct placeholders for the data?
- Was the proper pathname inserted for the dynamic image code?

In most cases, if the recordset test displays the correct data, the error can be found either in the placeholders or in the path statement. As mentioned, if you can't solve the problem by re-creating the text or image placeholders, it may be easier to delete the entire page and start from scratch.

Remind students that most of the exercises and examples are directly transferable to the other server models, but that specific implementations and code construction will probably be different. You should also tell the students that each server model may offer different or additional behaviors available only to that server.

You may also have the students perform a search in Google or Yahoo for specific types of dynamic behaviors, such as shopping cart or blogcast solutions that can be implemented in their own sites. One recommended site is the Dreamweaver Exchange at **http://tinyurl.com/3yrby7** that offers a clearinghouse for a whole host of plugins and widgets that provide helpful applications and services, many of them free.

Review questions

1. How do you create a recordset that will show you data for events occurring after today?
2. In what part of the Recordset dialog box can you enter your own SQL statements?
3. How can you see if the SQL statement will return the desired recordet?
4. How do you display a specific field from the recordset?
5. Do you need to upload any files to the testing server before you can test dynamic pages?
6. How can you get Dreamweaver to dynamically display an image from the images subfolder if the stored data doesn't contain the path name?

Review answers

1. Filter the recordset by date by using the operator `>now()`.
2. Select the Advanced mode of the Recordset dialog box to write your own SQL statements.
3. Click the Test button in the Recordset dialog box.
4. Open the Bindings panel and insert one of the field placeholders from the recordset into the webpage.
5. Yes. Dreamweaver will prompt you to update the specific file(s) on the server. Click Yes in the dialog box to upload dependent files if requested.
6. You need to add the folder, or path, to the dynamic link created to generate the image.

Lesson 16: Working with Code

In Lesson 16, students are guided through the various productivity tools for coders and developers offered by Dreamweaver. Although Dreamweaver is famous for being a WYSIWYG HTML editor, it still offers serious power and productivity enhancements to diehard coders.

Selecting, Expanding, and Collapsing code

Page code can become long and difficult to work with. These exercises show how students can make large amounts of code more manageable. Remind students that these operations do not damage the code in any way.

Adding and editing code

Dreamweaver offers code hints for HTML, CSS, and most of the supported programming languages, including JavaScript. Point out how Dreamweaver will only provide code hints for elements and processes that are valid at a particular moment or position in the code.

Using Code Navigator

Discuss with students how to use Code Navigator to troubleshoot errors in CSS formatting. Demonstrate how Code Navigator works in all views of Dreamweaver, even in Live view and Live code.

Live code and inspect mode

Emphasize that Dreamweaver is one of the only HTML editors that actually shows users interactively how the code runs within the browser. This feature allows users to debug and test most aspects of a dynamic application before uploading it to the server, saving time and money.

Commenting your code

Describe how comments within the code can leave important notes for you or other developers to explain why or how certain features were developed. Explain how vital it is to leave these notes when the work is fresh in your mind. Reiterate that comments are not rendered in the browser.

Review questions

1. In Code view, your cursor is inserted in a `<p>` element. How can you get Dreamweaver to quickly select the parent `<div>` to move it?
2. What option would you select to make Dreamweaver highlight broken or improperly structured code?
3. How do you get Dreamweaver to close a code element?
4. In Code view, what happens if you select a CSS rule from a linked file that applies formatting to an element in the source code?
5. You want to format a dynamic state of an Accordion panel, how do you find out what CSS rule is applied to the hover state?

Review answers

1. Click the Select Parent Tag icon until the desired parent element is selected.
2. Select Highlight Invalid Code from the Code View Options menu to have Dreamweaver identify broken or improperly structured code.
3. Type `</` and the program will close the first available open element.
4. The external style sheet will be loaded in the Code view window.
5. Select Live view and place the cursor over the element you wish to format until the hover state appears. Right-click and select Freeze JavaScript. Use Code Navigator on the frozen display to determine what CSS Rules need to be modified or created.

Lesson 17: Publishing to the Web

In Lesson 17, students will be guided through publishing their site online. Most websites use FTP, but RDS and local network connections are also common.

Defining a remote site

Although it is not required for the training, it is highly recommended that some method for establishing a remote site connection be provided. If cost is an issue, you can use the local testing server for the remote site.

For the FTP connection, students should know how to fill in the remote site specifications. Tell them that most web hosting companies will provide all this information when an account is opened. If the student is using a company-hosted site, their IS/IT manager will be able to provide the necessary information. In some cases, companies use a *staging* server to prevent direct access to online content. The students should be aware of all these types of server environments.

Stress that the Put and Get operations overwrite the files at the destination. Students need to be especially careful not to get a file from the server that will overwrite a newer version on the local site. Inform them that the server may be located in a remote location in a different time zone. And, that when files are posted to this server, the creation time of the file may be localized. The students should be aware that an older file on the server may actually appear to be newer than the one on the local computer. It should be recommended that local files be backed up separately to prevent data loss.

Cloaking folders and files

Dreamweaver CS5 allows you to cloak individual folders and files. Explain to the students that some folders and files may not need to be uploaded to the server, such as template and Library items. Students should also be aware that some items may actually be different on the remote server and should not be updated. This is especially true with data connection files when the connection scripts on the local computer are not the same as the scripts for the server.

Other file types that should be omitted are ones not typically supported by web browsers, such as TIFFs, PSDs, Word documents, spreadsheets as well as materials that may contain private or sensitive data.

Wrapping things up

Most lessons and exercises focus on accomplishing specific training goals. In many instances, the web pages remain incomplete. Students should be aware that in a normal workflow they should complete each page before proceeding to the next. If you have time now, have the students finish any incomplete pages by adding the content suggested in the book. Feel free to allow the students to use their imagination as permissible. Additional images are included in the **images** folder for this purpose.

Putting your site online

By this time, the site is complete and ready to be published. Stress the point that most if not all of the structure of the local and remote sites will be identical once the site is published (with the exception of cloaked files and folders). Make the students aware that certain types of servers use a specific folder to host the live site files. This folder is often named **public**, **public-html**, or **www**. Tell the students to check with the hosting company or their IS/IT manager for the proper name on their own site. This folder name will have to be added to the definition of the remote site.

Have students examine the site structure that Dreamweaver creates on the remote site. They should notice folder names in particular. Explain how these folder names become part of the page URL. Look at some actual websites and discuss what a URL, such as **http://www.adobe.com/products/dreamweaver,** means in terms of site structure. In this example, you can see that the server **www.adobe.com** contains folders for "products" that in turn contain a folder for "dreamweaver".

Students should be advised to minimize the number of subfolders employed and to keep the names of the folders short, if not abbreviated.

Review questions

1. What remote connection option would you select to by-pass firewall restraints?
2. How would you cloak Excel files (xls) saved inside the website folder?
3. You have cloaked the Library folder. Can you upload it without uncloaking it?
4. True or False. The MySQL database used in the sample website is uploaded automatically when you upload your site.
5. What happens if you select a file in the Local Site window and click the Get button?

Review answers

1. Choose Use Passive FTP to overcome firewall restraints.
2. Open the Site Setup dialog box and select the Cloaking category. Select the options Enable Cloaking and Cloak Files Ending With, if necessary. Enter the extension .xls in the field. Separate the entry from any others with a space.
3. Yes, it can be uploaded manually. Just right-click the folder and choose Put from the Context menu.
4. False. In most cases, the MySQL database must be uploaded and configured separately to the web server.
5. The Get button always downloads the selected file from the Remote server regardless of the window selected.