



Adobe®
Dreamweaver® CC

CLASSROOM IN A BOOK®

The official training workbook from Adobe Systems

Instructor Notes

Adobe® Dreamweaver CC Classroom in a Book®

© 2014 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, Acrobat, Adobe Reader, the Adobe PDF logo, Bridge, Classroom in a Book, Creative Suite, EchoSign, InDesign, and Photoshop are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Apple and Mac OS are trademarks of Apple, registered in the U.S. and other countries. Microsoft, Excel, Internet Explorer, Outlook, PowerPoint, Windows, Windows 7, and Windows XP are either registered trademarks or trademarks of Microsoft Corporation in the U.S. and/or other countries. Lotus Notes is a trademark or registered trademark of IBM. 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.

Book:

ISBN-13: 978-0-321-91941-0

ISBN-10: 0-321-91941-6

Instructor Notes:

ISBN-13: 978-0-133-41596-4

ISBN-10: 0-133-41596-1

INSTRUCTOR NOTES

Getting Started

The *Adobe Dreamweaver CC Classroom in a Book* course provides students with techniques, tips, and solutions for using HTML, CSS, and Adobe Dreamweaver CC software. These instructor notes are intended to complement the information in the *Adobe Dreamweaver CC 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

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

Adobe Press books are published by Peachpit, a division of Pearson Education located in San Francisco, California. For the latest on Adobe Press books, go to www.adobepress.com. You can find ordering info at www.peachpit.com/academics/academics_detail.aspx?st=67279.

Course strategy

This book consists of 14 print lessons and one bonus lesson available online. If you complete one or two lessons per week, students will have time at the end of the semester 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, students create a complete website for a fictitious community association and learn how to use and modify one of Dreamweaver's built-in CSS layouts.

You can evaluate each student based on his weekly completion of the lesson exercises. Requiring a student to design and build his 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 named [*Student's Name*]-**web** that will contain all their websites. Then, create a **DW-CC** 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 intended scenario requires the student to build an entire website by completing the lessons in order. In this method, 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 her site. Incomplete or incorrectly built components from one lesson may adversely affect a subsequent lesson or actually cause it to fail. On the other hand, making errors is a natural occurrence and learning how to troubleshoot and track down such errors has valuable implications in a “teaching” environment.

If you'd rather not deal with the wide assortment of errors that may occur, a “jumpstart” method for completing the course is offered. This method treats each lesson folder as a complete website that includes all pages and assets needed to complete the specific requirements or goals of the lesson.

The jumpstart 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 lesson, unless the instructions within the lesson itself specify to do so.

Using the jumpstart method, each lesson folder will be defined in Dreamweaver as a stand-alone website. Therefore, at the end of the course the student may have defined 15 websites, one for each lesson.

Additional resources

Instructors and users of Dreamweaver rely on a variety of resources to supplement their knowledge of the program. You may find the following resources useful to exploit.

Dreamweaver Help

The Help feature of Dreamweaver CC uses web-based resources that will provide a richer experience for most users and includes complete descriptions of all the program's features. Since the contents are hosted online, they are frequently updated live and will feature the latest documentation. If you think you may need to access the Help contents offline, you can print or save the contents to a PDF document to access the information at any time.

Adobe CC video tutorials

The Adobe Help site also contains links to movies and step-by-step tutorials that showcase new CC features as well as basic program functions and web design techniques.

Adobe website

The Adobe website (www.adobe.com) contains a variety of Dreamweaver training and support resources. Many step-by-step tutorials, galleries, and other resources are available here, and the user-to-user forums are great places for students to learn from more experienced users. The Adobe Developer Center (www.adobe.com/devnet/dreamweaver.html) provides dozens of tutorials and other articles from experts and authors in the Adobe community, as well as thoughtful articles about design and technology.

● **Note:** Many hyperlinks are case sensitive; be sure to type them as shown.

Adobe Certified Expert (ACE) program

This program is designed to validate an expert skill level for Adobe Dreamweaver CC. Testing procedures ensure that each ACE-certified student demonstrates expert product knowledge of the current release of Adobe Dreamweaver, resulting in enhanced marketability and credentials for ACE graduates. ACE program training is available through Adobe Authorized Learning Providers (AALP) and self-study. For more information about this program, send an e-mail to certification@adobe.com or visit the Adobe website (www.adobe.com).

Lesson 1: Customizing Your Workspace

In Lesson 1, students are exposed to the Dreamweaver program interface and are familiarized with program commands and conventions. The lesson explores the various tools, document windows, and program panels that allow students to create and edit webpages and content. Students experiment with customizing toolsets and their own workspaces and are shown how to create a custom keyboard shortcut.

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

Review questions

- 1 Why would you use Code view? Design view? Live 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. Design view allows you to see an approximate preview of the completed web content. Live view previews the final web content and supports interactive and dynamic content elements, such as hyperlinks, video, and audio.
- 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 are shown how to create HTML code and a basic webpage by hand in a text editor, exposing them to the language and basic coding techniques. The same webpage is then re-created in Dreamweaver. By learning how to write HTML by hand, the students will lose their fear of coding in general and can easily compare both methods of code creation, getting a better understanding of the productivity 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. Finally, students are introduced to the latest version of the language: HTML5. The concept of semantic web design is described and several of the new elements are listed. In fact, the subsequent exercises and examples will be based on these new elements and structures. 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?
- 4 What is the purpose of HTML5 semantic markup?

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 `` 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 ™.
- 4 Semantic tags, such as `<header>`, `<article>`, and `<section>`, convey the meaning or purpose of the webpage content directly. These tags will enable advanced functionality in the future by identifying specific content elements for search engines and other types of applications.

Lesson 3: CSS Basics

Lesson 3 provides an overview of cascading style sheets (CSS). Students should be aware that a movement was made over a decade ago to separate the structure and content of webpages from its 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. This helps to illustrate how web design differs from other types of design. Students may be familiar with using MS Word or Adobe InDesign, Illustrator or other similar design tools. In the beginning, web design started in the same way: Each line or paragraph was selected and formatted individually. CSS was created to break this model and revolutionize how documents are styled.

HTML defaults

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

HTML5 has broken from this concept of *default* formatting. At the time of this writing, there was no default style sheet for HTML5 elements. This means in the future HTML elements may not exhibit any intrinsic formatting at all. For the time being, browser vendors are continuing to support the default style sheet for HTML 4.01, even for pages using HTML5 code and structures.

CSS box model

An essential concept to CSS is the box model; it is the basis for all webpage formatting and construction today. To achieve a successful outcome for page design and development using CSS, students must understand how this model works and is applied. This concept is a difficult one for most people. You may want to create additional exercises to help develop this understanding. The lesson provides several hands-on files that will demonstrate this and other CSS concepts.

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 selectors, classes, and IDs.

● **Note:** This concept is continually reinforced throughout the subsequent lessons.

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. The student is also introduced to the Code Navigator and the CSS Designer to learn how to check and troubleshoot CSS formatting.

Formatting objects

In the past, most webpage layouts were built using tables. These techniques have been replaced with layouts based on `<div>` and other elements formatted by CSS. These hands-on exercises take the students through this process step by step, illustrating the important concepts of CSS-based layout. They will learn how CSS can control the width, height, borders, backgrounds, margins, padding, and positioning of HTML elements.

CSS3 overview and support

This section introduces the student to the advances that are being incorporated in the new version of CSS. Students are also apprised of the usability of CSS3 today. A sample file illustrates just a few of the amazing new features.

CSS rule syntax: write or wrong

This sidebar is included because of changes to Dreamweaver CC that require users to frequently create and edit CSS selectors and declarations by hand. Spelling, punctuation, and the construction of each CSS rule are critical. Even the slightest error can break the rule, or even the entire style sheet. Testing students on these aspects and stressing their importance would be valuable since they will be more directly responsible for the code construction in Dreamweaver CC.

At the end of this lesson, try creating several test webpages with intentional errors inserted in the cascading style sheet and have the students troubleshoot these pages to find the errors.

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?
- 4 What advantages does the CSS Designer have over the Code Navigator in troubleshooting CSS styling?

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.
- 4 The CSS Designer can display CSS styling applied inline. It can show you the COMPUTED or accumulated value of all the CSS rules applied to a selected element. And, it allows you to edit the specifications live.

Lesson 4: Creating a Page Layout

In Lesson 4, students are walked 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 a search engine, like Google or Bing. 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 the pages and page components they will need for their site. (Note that the list may vary from page to page.) Then, have students 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, Adobe Photoshop, or Adobe Illustrator 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 final 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.

During this lesson, the student will learn more about the new movement toward semantic web design, or using new HTML5 elements built specifically for certain purposes like navigation menus and text content. An explanation is also provided on how to substitute HTML5 structures for equivalent HTML4 elements for users who desire or require backward compatibility.

Working with content and formatting

The student works step by step through the process of building a complete webpage design. They learn how to insert a background graphic and a new HTML5 `<nav>` element. They modify the basic page structure by adding new elements and 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 good moment to discuss copyright, fair-use, and other content restrictions.

Validating webpages

Reinforce the previous concept of proper HTML page structure and potential coding errors. The student will check the validity of their completed layout with an external validator. Note: You must have Internet access to complete this exercise. Once the page is checked, Dreamweaver will display a list of errors within the page. If the instructions were followed correctly, there should be only one error. In this case, the error that should be reported in the current lesson refers to the “empty” image source (`src`) attribute. Since the image placeholder will not get this `src` reference until they start building pages from the site template in subsequent lessons, they can ignore the error for now.

If students report other errors, take time to evaluate the error messages and walk them through possible ways to correct the errors. Refer back to Lessons 2 and 3 as necessary for the proper construction of HTML and CSS.

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 5: Working with Cascading Style Sheets

In Lesson 5, students delve into Dreamweaver's styling 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 Designer

Students use the CSS Designer 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.

Students will add new CSS rules as they proceed through the lessons in 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.

Working with type

Much of the content inserted into a webpage will be text. It is essential that students are familiarized with the way text is created and formatted for the web. Explain how type is represented by computer-based typefaces and font variants and how, until recently, web designers were dependent mostly on fonts installed on their visitor's computers. Describe how web-based fonts evolved and are now becoming more popular. Be sure to emphasize that they cannot use just any font and what the legal ramifications are for themselves and their companies for using fonts illegally.

Here's where you can provide a brief explanation of the advantages Adobe Creative Cloud subscribers have over other users. Show them all the programs, products, and features offered by Creative Cloud, including Typekit and Adobe Edge Web Fonts.

Working with classes, IDs, and descendant selectors

In the three exercises in this section, the student will learn how to create and apply CSS classes, IDs, and descendant selectors. A lot of space is devoted to these subjects to make sure the student understands how important these items are in the creation of a modern website. Be sure to emphasize the essential differences between classes and IDs. Reinforce the proper naming conventions and application of these attributes. You may want to create additional exercises to drill the points home.

Creating 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.

In these exercises, layout errors will occur caused by the new CSS rules created by the student. This is a great teaching moment to show how formatting can be inherited unintentionally by related elements. Show how the horizontal menu uses similar structures to the vertical menu and how the HTML element relationships or the CSS selectors will need to be modified to correct the situation. Explain how these “unintended consequences” can be a regular occurrence in web design.

Creating hyperlink rollover effect

Discuss the default hyperlink behavior with the students and emphasize how these behaviors, and others, can be modified by CSS. Remind the students of the four hyperlink pseudo-classes 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, and how to use different techniques to create advanced web layouts.

Moving rules to an external style sheet

This is a good time to reiterate how CSS can be implemented in a webpage. Explain how CSS can be created inline, embedded, and stored in external files. Compare and contrast the advantages of each method.

Hyperlink pseudo-classes

This sidebar introduces the topic of CSS pseudo-classes. Explain how pseudo-classes provide additional capabilities to CSS to format unique structures, relationships, or behaviors in an HTML layout. Warn students that many of the pseudo-classes are new and not supported in every browser or fully implemented in the browsers that do. Students should be careful to test any pseudo-classes they wish to use to make sure they are supported in their chosen application.

Creating external style sheets for other media

The power of CSS is in the ability to format an entire website from one or more external CSS files. Students are shown 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 pseudo-class?
- 5 Describe the default behavior 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 webpage 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 pseudo-class 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 6: Working with Templates

In Lesson 6, 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 .dwt extension. 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 only the aside section of the sidebar is editable and what limitations, if any, 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, if desired.

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. Content inserted into editable regions of the child pages will not be altered when templates are updated.

Using Library items

Library items are like miniature versions of templates designed to hold stand-alone components that can be deployed to multiple pages, such as the vertical menu. Library items have all the benefits of templates allowing users to modify the components and then update them automatically on whatever page they appear. 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 dynamic widget.

Make sure you point out that when updating library items site-wide all affected pages must be 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 the use of server-side includes (SSI). SSIs are more powerful than Library items because they are loaded dynamically when the page containing the item 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. It should be pointed 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 webpage 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 Child pages 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 7: Working with Text, Lists, and Tables

In Lesson 7, students learn how to add text content to their child pages, including HTML lists and tables.

Creating and styling text

Students learn how to import text content from an RTF file and to apply HTML heading and paragraph tags using the Property inspector. The sample text will be inserted by creating semantic structures. Refer back to Lesson 2, if necessary to emphasize the importance of these structures.

Text will be formatted using HTML elements such as `<h1>`, `<h2>`, and `<p>`. 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 would 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 Styles 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 webpages

The content imported in this lesson contains several words purposely misspelled. 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. Stress 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 differences across a page, selected pages and 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 a file.
- 2 If you copy single paragraphs from most programs, they will be inserted into Dreamweaver with break `
` characters 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 be used only to format quoted material.
- 5 No. Since the entire table column must be the same width, a class needs to be applied to only one cell within the column.
- 6 Yes. Custom terms can be added while spell checking.

Lesson 8: Working with Images

In Lesson 8, 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 and resolution to 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 during optimization 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 (alternative) text. Two helpful resources are “Writing great Alt text” (<http://tinyurl.com/alternative-text-tips>) and “How to Write Good Alt Text for the Images on Your Webpage” (<http://tinyurl.com/alternative-image-tips>).

Adjusting image positions with CSS classes

This exercise describes the creation and application of CSS positioning with custom classes. Explain how CSS allows you to control the placement of images across the entire site at once. Explain that HTML positioning attributes have been deprecated in HTML5 and must be changed one by one directly on the page itself and doesn'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 eight 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.

● **Note:** To complete this exercise, you will have to download and install Bridge if it is not already present.

Using Adobe Bridge and Photoshop Smart Objects

Adobe Bridge is a stand-alone application installed automatically with other Creative Cloud apps, but no longer with Dreamweaver. It enables users to view the contents of various proprietary and open-source file types. It simplifies the workflow between all the Creative Cloud applications. In this exercise, students use Bridge to access folders within the site and choose images to insert into the webpage.

Students are walked through the process of using Photoshop Smart Objects. Students learn that Smart Objects maintain a connection to a copy of the original Photoshop image. Smart Objects 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. One limitation of Smart Objects is that when the original image is changed it will update all versions of that image wherever it appears in the site.

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. Students should be cautioned that drag and drop placement may insert images in the wrong position in the code.

Optimizing images with the Property inspector

Dreamweaver has a set of 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 among others, 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 JPEG image have?
- 3 Can you insert a 300 ppi image on your webpage?
- 4 Can you copy and paste a multi-layer Photoshop file?
- 5 In the Property inspector, what does a circular arrow by the image Width and Height values mean?
- 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 copy only 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 9: Working with Navigation

In Lesson 9, students work with relative and absolute hyperlinks to create both internal and external site navigation. Students will learn how to create various types of hyperlinks, including text, image, and e-mail.

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 are shown how to change this default behavior so that a new window can be opened to load a specific webpage. Take time to remind students why they should use this method to open a separate window for external links. Suggest times when this method would be appropriate.

Students will also be shown how to create hyperlink destinations using unique ID attributes, and how to set up links to target these internal destinations.

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?

Review answers

- 1 URL stands for uniform resource locator.
- 2 Absolute links spell out the entire URL address of a link. A relative link contains only the filename and the path information needed to load the image in relationship to its location to the webpage 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 information 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 (#) mark in front of it.
- 6 Since an ID attribute should be unique, you can use it in the same way as a named anchor.

Lesson 10: Adding Interactivity

In Lesson 10, students learn about Dreamweaver behaviors and how to use them within a layout along with jQuery 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. Behaviors are usually dynamic functions created by predefined JavaScript inserted automatically by Dreamweaver. These functions allow users to add sophisticated actions to their websites without any prior programming or scripting knowledge. 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 jQuery Accordion widgets

Students will insert a jQuery Accordion widget to add an interactive element to the site. Discuss the advantages of using jQuery 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” option 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 page is first loaded so that there is no delay in the performance of the behavior.

Lesson 11: Working with Web Animation and Video

In Lesson 11, students learn how to incorporate HTML-compatible animation and video in their site.

Understanding web animation and video

Major changes have occurred on the web concerning web animation and video. Apple dropped support for Flash on their iOS devices, including the iPhone and iPad. This decision sent shockwaves through the web, which had settled on the use of both Flash animation and video for the last 10 years. The introduction of Flash video a few years ago revolutionized the way video was delivered via the web and made services like YouTube a reality.

Unfortunately, the burden of supporting Flash on a mobile device was too great, and Apple, as well as other manufacturers, decided on a different course. Work has progressed with various web standards organizations to create non-proprietary methods for delivering animation and video directly compatible with HTML5.

In this lesson, students will be exposed to the current techniques for incorporating HTML5-compatible animation and video within their websites. Make the students aware that these methods have not been formally adopted by the industry and may change over time.

Adding web animation to a page

With the loss of Flash as the de facto standard, Adobe has been working on other methods for delivering high-quality web animation. Adobe Edge Animate is a new application, which promises to provide much of the capabilities of Flash animation using HTML5-compatible files and functionality. Students will learn how to incorporate an Edge Animate deployment file into their site.

Adding web video to a page

Flash video literally revolutionized the delivery of video content on the web. Before Flash, web designers had to pick among a list of proprietary video formats to use and insert it within their page. Unfortunately, each video format required a proprietary video player, too. Files that worked on Macs may not have worked in Windows. Frequently users had to download a new player before they could view a video. Flash video changed this situation overnight.

Almost any type of video could be inserted into Flash, saved in the FLV or F4V format, and inserted within a webpage. The Flash player that supported SWF animation would automatically handle the video, too. Services like YouTube sprung up and provided easy ways to distribute and view high-quality video.

This ease of use and simplicity vanished with Apple's decision to drop support for Flash on iOS devices. Instead, the industry is almost back to square one, with multiple video formats fighting for supremacy in the market.

In this lesson, students will be introduced to a method for incorporating several HTML5-compatible video formats into their websites. They should be made aware that this method has not been formally adopted by the standards organizations and may change in the future.

Dreamweaver continues to offer a high degree of integration with Flash components, which still commands over 90 percent installation on desktop systems. Flash video can be used as the alternate content for old browsers that do not recognize the new HTML5 `<video>` element and most desktop-based systems that still incorporate the Flash player browser plug-in. Make sure students are aware that when Flash video (FLV) is added to a webpage, specific JavaScript files as well as the SWF installer are required and must be uploaded to the remote site along with the webpage and the FLV files themselves.

Note that the version of Dreamweaver CC available as of this writing does not incorporate the necessary files for fully supporting Flash video. So, the lesson provides the steps to replace the default Flash fallback video format by using the legacy insertion method. Remind the students that Creative Cloud apps are updated on a regular basis and any errors in the program will eventually be corrected.

Don't forget to remind the students that they can test the video right inside Dreamweaver.

Review questions

- 1 Why shouldn't users rely on Flash animation for their sites?
- 2 What type of web animation can be used instead?
- 3 Can you use Flash video still?

Review answers

- 1 Flash animation files are proprietary file formats that are not supported by iOS and other mobile devices.
- 2 Animation created using native HTML5 elements, JavaScript, and CSS, like that created by Adobe Edge.
- 3 Yes, as alternative content for older browsers and desktop systems that still support the Flash player. Otherwise, you can insert video using HTML5-compatible structures and resources.

Lesson 12: Working with Forms

In Lesson 12, 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. This would be a good time to engage the students to discuss what types of forms they have used, both on paper and on the web.

Ask the students some of the following questions with regard to their experience with forms and the information collected:

- Do they think the information was going to be used on the computer?
- If so, what limitations do they think the use of paper forms placed on the process?
- What advantages do they think an electronic form provides in this process?

Inserting a form and form elements

In these series of exercises, the students insert an HTML `<form>` element and populate it with text fields, check boxes, radio buttons, and other form elements. Students also insert and configure HTML5 form elements and learn how to use advanced form features.

Students insert various types of text fields. Be sure to discuss the differences between text, password, and e-mail fields. Point out that some settings are applied using new HTML5 attributes that may not be compatible with older browsers. Remind them that all supporting files must be uploaded to the remote server along with the page that contains the form.

Advise students that long forms should be divided into logical structures using the `<fieldset>` element. Encourage students to use Code and Split views when adding `<fieldset>` elements to the form to ensure the correct elements are being selected and grouped.

Submitting forms

Students learn how to create buttons to submit and reset the form. Discuss the disadvantages of sending the form data via e-mail. Explain how many people don't use a locally installed e-mail application, such as Microsoft Outlook or Apple Mail. Instead, they use Internet-based applications, such as 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 How do password fields differ from normal text fields?
- 3 Why should you enable the accessibility feature in Dreamweaver for forms?
- 4 Is the form action applied to the submit button?
- 5 What happens in some browsers when you try to load a page with active content?
- 6 What is a variable used for in the PHP form e-mail page?

Review answers

- 1 They are all created by the `<input>` tag.
- 2 When a user types into a password field, the input is masked by bullets or asterisks so it can't be read.
- 3 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.
- 4 No. The action is applied directly to the `<form>` element itself.
- 5 Some browsers block active content and require users to approve the operation of the element.
- 6 A variable is a piece of data that will be created within the code or retrieved from the form and then used within the script.

Lesson 13: Publishing to the Web

In Lesson 13, 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 web 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. Recommend that local files be backed up separately to prevent data loss.

Cloaking folders and files

Dreamweaver CC allows you to cloak individual folders and files. Explain to the students that some folders and files, such as template and Library items, may not need to be uploaded to the server. 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, and 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 webpages 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.”

Advise students to minimize the number of subfolders employed and to keep the names of the folders short, if not abbreviated, to help pages load faster.

Review questions

- 1 What remote connection option would you select to bypass 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 .xls in the extension 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.

Lesson 14: Designing for Mobile Devices

In Lesson 14, students will be guided through the process of adapting the website for smart phones and mobile devices.

Designing for mobile devices

In this section, the student is introduced to the concept of mobile design, media types, and media queries. This technique is a new concept that was developed with the rise in popularity of smart phones and tablet computers.

Students should be made aware that the older technique of media types was intended more for fixed format devices that conform to a recognizable standard, such as print and projection devices. The introduction of a raft of phones and mobile devices that could access the Internet upset this established standard.

Originally, the idea to adapt existing fixed-width websites to fixed-width devices was a simple and straightforward accommodation. But the new mobile devices come in a variety of widths and heights and can often view webpages in both portrait and landscape modes. The media type property was never designed to deal with multiple devices or orientations at once.

The development of media queries was an answer to the limitations of the media type property. Media queries work in tandem with the browser. The browser interrogates the display device to determine the actual width and height of the viewport. Once the dimensions are established, the browser checks to see if there is a media query designed to meet those specifications. If there is one, the browser loads any CSS styling contained within the media query.

Media queries can be written to target specific media types, dimensions, and even orientations. Queries can be used to load separate style sheets, or simply to reset specific rules in the default style sheet.

Discuss with the students their own experiences using the Internet with mobile devices. If possible, have students use their own phones and devices to access the Internet in class. Have them browse to their favorite sites using their devices and their desktop computers, if possible. How do these websites respond to each environment? How do different devices display the targeted websites? Do specific devices display the websites differently? Have the student compare, contrast, and describe the different methods.

Make sure you have a few websites in mind to demonstrate how different companies deal with mobile users. Identify sites that provide a custom mobile experience and ones that do not. If possible, find sites that actually default to a separate mobile app. Compare and contrast the difference between mobile-ready sites and sites that provide a completely separate experience. For example, membership or subscription services like Facebook and Netflix offer separate mobile apps that can be used on tablets and smart phones.

Creating a media query for smart phones

In this exercise, students get hands-on experience building their own media query for smart phones. Explain the various limitations of smart phones compared to desktop systems. These include but are not limited to size, processing power, memory, browser compatibility, and connection speed. Make sure students are aware that smart phone users may also be concerned with using up their plan minutes or other factors that affect their Internet access.

These limitations can dramatically affect the experience of a user on your website. One aspect of the media query is the ability to resize and turn off various components to make them fit the screen of the mobile device. However, emphasize to the student that making a component smaller or turning it off does not prevent it from downloading to the phone. Even invisible elements are using up the bandwidth of the device and can potentially slow down or otherwise affect the webpage. Remind students that certain types of content, such as Flash animation and movies, are not compatible at all with various devices.

Adding rules to a media query

Students will start to add new rules to their first media query. Be sure to remind them to select the media query in the @Media pane of the CSS Designer before creating any rules. Otherwise the rule may be added to the wrong place of the style sheet. You may want to demonstrate what happens if they don't select the media query first and how they could fix the error without having to delete and re-create the rule.

Emphasize that in this exercise they are adding rules to the existing style sheet. This technique is an efficient way of building mobile designs because they have to reset only a handful of rules to adapt the design to the mobile environment. The majority of the default style sheet is still inherited.

Another workflow is to create separate style sheets for each type of device or orientation that can be loaded by a custom media query inserted in the <head> section of the page. In this workflow no styles are inherited, and you must build an entire set of styles to format all aspects of the site.

Adapting CSS navigation to mobile environments

This exercise reinforces the students' skills and techniques for identifying the CSS rules that affect major components within the site. They will have to identify the rule that controls the vertical menu and then create new rules to change the width of menus and position of the text. Review the techniques learned earlier with the CSS Designer and Code Navigator. Remind them of how multiple rules may gang up to create an overall effect.

Mobile-ready vs. mobile-optimized

This sidebar discusses the difference between two basic techniques for building websites compatible to mobile devices. Explain how making a web design mobile-ready is not the same as making it mobile-optimized. Mobile-ready sites use media queries and responsive images and video. Mobile-optimized sites use more sophisticated methods for adapting a site for mobile users. This can include using JavaScript to load alternative content, all the way to building a second site designed specifically for mobile use.

Some of the factors that determine whether you want to pursue one course or the other include: customer expectations, amount of traffic, type of content, and so on. As your traffic grows and demands increase on the site, the more sense a mobile-optimized makes.

Styling the main content

This exercise moves focus to the main content section by formatting headings and paragraph text. Discuss with students the needs of visitors using phones and tablets. Do they need larger text? Smaller text? Should they have smaller headings and larger body text? Remind them that they will have to review every page once the styling is created to make sure all the text is affected properly and as expected.

Testing a media query

Dreamweaver allows you to test most of the formatting of a media query directly in Design view. For dynamic behaviors you can switch to Live view or load the page into the browser. Discuss other ways to test mobile designs.

Responsive design

Working with basic HTML components is easy compared to images and video elements. There's no easy way to adapt these components to mobile designs because normally the widths and heights are applied directly to each image and video. A new technique was developed that allows you to make images respond automatically to the size of the screen.

Basically, you take the width of the image and divide it by the width of the original parent element. Then, you remove the width and height attributes from the image and insert the result (%) only in the width. When the browser loads the image, it calculates its size based on the dimensions of the screen itself and adjusts the height automatically.

Explain to the students that responsive design is not the perfect solution in every case. Some of the disadvantages include that the user is downloading images designed for desktop layouts and that some images might become too small to be usable. Alternatives to responsive design typically rely on scripting that can swap out for more appropriate images or even stop images and other inappropriate content from downloading altogether.

Creating style sheets for tablets

In the next section, you'll create a new media query to adapt the basic design for tablet devices. Since tablets are larger than phones, the requirements are different and not as demanding. Frequently, you can simply make the layout and text elements slightly smaller to support them.

Edge Inspect

Adobe offers a program called Edge Inspect that enables you to test mobile designs using a plug-in for the Chrome browser and an app you can download to your smart phone or mobile device. Edge Inspect allows you to test your pages on multiple devices at once without having to upload the code to the Internet. Discuss the merits of testing pages locally inside Dreamweaver and using Edge Inspect over uploading pages to a live web server. What are the benefits and detriments of each method?

Note that Edge Inspect currently works only with the Chrome browser. You must download and install Chrome on all student computers and then obtain the Edge Inspect app for all devices you wish to use in testing. Edge Inspect is only available as a subscriber to Adobe Creative Cloud. To learn more check out html.adobe.com/edge/inspect/.

Review questions

- 1 For what purpose is the media type property intended?
- 2 What does a media query do?
- 3 What special considerations does a designer have to take into account for smart phones that are different in a desktop computer?
- 4 What special accommodation do some designers make for mobile devices?
- 5 How does responsive design help images and video adapt to mobile environments?
- 6 How can you test media queries and responsive designs?

Review answers

- 1 The media type property is intended to address the use of noncomputer devices that can access the Internet.
- 2 Media queries are used to target specialized CSS styling to specific sizes and orientations of display devices.
- 3 The screens of smart phones are obviously smaller than those on a desktop computer, but you also have to consider connection speeds, memory restrictions, processor speed, and compatibility issues.
- 4 Some designers may build a completely separate mobile site hosted on a subdomain that provides content optimized for smart phones and tablets.
- 5 “Responsive design” uses the relationship of the width of the element as it compares to the width of its parent container to create a percentage. The percentage is used in place of the actual width of the element in the HTML markup. When the image loads in the browser, it automatically scales to assume a size that’s appropriate to the screen dimension and its parent element.
- 6 Media queries can be tested in Design, Split, and Live view modes of Dreamweaver. Adjust the width of the document window and the media queries will be applied as appropriate to the width and orientation of the screen. Images and video inserted using the responsive design technique will only be displayed accurately in Live view. Media queries and responsive design should also work in any modern browser.

Lesson 15: Working with Code (Online Bonus Lesson)

In Lesson 15, 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. While most users spend most of their time in Design view, emphasize to students that they should become familiar with HTML code and try to work with it as often as possible.

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. Note the slight graphical differences between the way Dreamweaver operates in Windows and on Macs. In Windows, Dreamweaver displays plus and minus signs for collapsed code, on the Macintosh it displays triangles that twirl open and closed.

Adding and editing code

Dreamweaver offers code hints for HTML, CSS, and several supported programming languages, such as JavaScript. Point out how Dreamweaver will provide code hints only for elements and processes that are valid at a particular moment or position in the code. Explain to students that if no code hinting appears that it does not mean they are doing something wrong.

Identifying CSS styling

Discuss with students how to use Code Navigator and the CSS Designer to identify and troubleshoot CSS formatting. Demonstrate how both features work equally well in Code, Design, and Live views, and even in Live Code and Inspect modes.

Be sure to highlight how to use both tools. Demonstrate that they can obtain different results by inserting the cursor in the content or by using the tag selectors. Point out the advantages the CSS Designer has over Code Navigator in that it shows the COMPUTED styling from all rules and allows you to edit the settings all in the same window. Also, note that Code Navigator does not identify inline CSS styling but CSS Designer does.

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. These features allow users to debug and test most aspects of a dynamic application before it is uploaded to the server, saving time and money.

● **Note:** This feature functions only with dynamic components that manipulate the coding or styling with a scripting or programming language.

Working in related files

Webpages frequently use one or more external files to provide styling or dynamic functions and behaviors. Tracking down and managing these files manually was always a time-consuming and difficult task. Dreamweaver does most of this work for you automatically. All web-compatible files called by or linked to within the code are listed in the Related Files interface that appears at the top of the document window, including CSS, external JavaScript, XML data, and so on.

At any time you can access those resources by clicking the name of the file displayed. The Code view window will open, if necessary, and focus on the content of the file allowing you to view and edit the code. Remind the students that when they make changes to related files that they must save the files separately. This can be done by using the Save All command, or by inserting the cursor in the Code view window when the related file is displayed and pressing Ctrl-S/Command-S. If the changes are not saved before the main HTML file is closed, Dreamweaver should prompt you to save each file. A dialog box will appear for each file in turn. Click OK in each dialog box to save any changes. Warn the students that if they click Cancel, their changes will be lost.

Accessing Split Code view

Although most Dreamweaver users know how to work in Split view, they may not be aware that they can also display two Code view windows simultaneously.

Show how you can display two Code view windows at the same time. The windows can display two separate files or two different views of the same file. You can display the source code by using the Related Files interface at the top of the document window. Show the students how they need to first insert the cursor in the Code view window and then select the file they wish to access. Explain how Split Code view can be used to compare or copy blocks of code from one section to another.

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. Demonstrate how comments are created differently in HTML and CSS markup. Reiterate that comments in either case are not rendered in the browser.

Although this is the last lesson, put this technique into practice earlier in the course, if possible. By Lesson 5 or Lesson 6 the students should be familiar enough with HTML code to introduce this concept to them. In fact, in Lesson 5, the students are introduced to commenting by reviewing the markup within the predefined CSS layout they use to create their site template. At any time when students adjust various page component or style sheets, prompt them to add appropriate comments.

Review questions

- 1 In Code view, your cursor is inserted in a `<p>` element contained within other markup. 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 automatically?
- 4 In Code Navigator, what happens if you select a CSS rule from a linked file that applies formatting to an element in the code?
- 5 A jQuery function changes the class of an element when the user moves the mouse over it. In Dreamweaver, how can you determine whether this function is working properly?

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 Switch to Split view and select Live view. Turn on Live Code and interact with the target element. Code view displays the changes to any attributes or elements.