Julie C. Meloni Michael Morrison



EIGHTH EDITION

Includes HTML 5
Preliminary
Coverage

Sams Teach Yourself

HTML and CSS

n 24 Hours

SAMS

Sams Teach Yourself HTML and CSS in 24 Hours, Eighth Edition

Copyright © 2010 by Sams Publishing

All rights reserved. No part of this book shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission from the publisher. No patent liability is assumed with respect to the use of the information contained herein. Although every precaution has been taken in the preparation of this book, the publisher and author assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained herein.

International Standard Book Number: 0-672-33097-0

Library of Congress Cataloging-in-Publication Data

Meloni, Julie C.

Sams teach yourself HTML and CSS 24 hours / Julie Meloni, Michael Morrison. – 8th ed.

p. cm.

Rev. ed. of: Sams teach yourself HTML and CSS in 24 hours / Dick Oliver, 7th ed., 2006.

ISBN 978-0-672-33097-1 (pbk.)

1. HTML (Document markup language) 2. XHTML (Document markup

language) 3. Cascading style sheets. I. Morrison, Michael, 1970- II.

Oliver, Dick. Sams teach yourself HTML and CSS in 24 hours. III. Title. $\label{eq:local_equation}$

QA76.76.H9404526 2010

006.7'4-dc22

2009046100

Printed in the United States of America

First Printing: December 2009

Trademarks

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Sams Publishing cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Warning and Disclaimer

Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied. The information provided is on an "as is" basis. The author and the publisher shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the CD or programs accompanying it.

Bulk Sales

Sams Publishing offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales. For more information, please contact

U.S. Corporate and Government Sales 1-800-382-3419

corpsales@pearsontechgroup.com

For sales outside of the U.S., please contact

International Sales international@pearson.com

Acquisitions Editor
Mark Taber

Development Editor

Michael Thurston

Managing Editor
Patrick Kanouse

Project Editor

Jennifer Gallant

Indexer

Ken Johnson

Proofreader

Dan Knott

Technical Editor

William Wolff

Publishing Coordinator

Vanessa Evans

Composition

Mark Shirar

Book Designer

Gary Adair

Introduction

In 2009, it is estimated that more than 1.5 billion people have access to the Internet, including 220 million in the U.S. alone. Throw in 338 million Chinese users, 55 million German users, 48 million British users, 38 million Russian users, and 67 million Brazilians, and you can see the meaning of the word "world" in the term World Wide Web. Many of these Internet users are also creating content for the Web—you can be one of them! Although accurate measurements of the total number of web pages are difficult to come by, Google's most recent data indicates they hit the 1 trillion mark of indexed pages in the middle of 2008.

In the next 24 hours, hundreds of millions of new pages will appear in accessible areas of the Internet. At least as many pages will be placed on private intranets, where they will be viewed by businesspeople connected via their local networks. Every one of those pages—like the more than 1 trillion pages already online—will use Hypertext Markup Language (HTML).

As you complete the 24 one-hour lessons in this book, your web pages will be among those that appear on the Internet. These lessons will also help you develop one of the most valuable skills in the world today: mastery of HTML.

Can you really learn to create top-quality web pages yourself, without any specialized software, in less time than it takes to schedule and wait for an appointment with a highly paid HTML wizard? Can this relatively short, easy-to-read book really enable you to teach yourself state-of-the-art web page publishing?

Yes. In fact, within the first two lessons in this book, someone with no previous HTML experience at all can have a web page ready to place on the web.

How can you learn the language of the Web so fast? By example. This book organizes HTML into simple steps and then shows you exactly how to tackle each step. Every HTML code example is listed directly before a picture of the web page it produces. You see how it's done, you read a clear, concise explanation of how it works, and then you immediately do the same thing with your own page. Ten minutes later, you're on to the next step.

After 24 hours of work, you're marveling at your own impressive pages on the Internet.

Beyond HTML

This book covers more than just HTML because HTML isn't the only thing you need to know to create web content today. The goal of this book is to give you all the skills you need to create a modern, standards-compliant web site in just 24 short, easy lessons. This book covers the following key skills and technologies:

▶ XHTML (eXstensible Hypertext Markup Language) is the current standard for web page creation. Every example in this book is fully XHTML-compatible. Where applicable, HTML 5 is also covered.

- ▶ All the examples in the book have been tested for compatibility with the latest version of every major web browser. That includes Apple Safari, Google Chrome, Microsoft Internet Explorer, Mozilla Firefox, and Opera. You'll learn from the start to be compatible with the past, yet ready for the future.
- ▶ There is extensive coverage of Cascading Style Sheets (CSS), which allows you to carefully control the layout, fonts, colors, and formatting of every aspect of your web pages, including both text and images. When it comes to creating eye-popping web pages, CSS goes far beyond what traditional HTML pages could do by themselves. For example, did you know that CSS allows you to specifically tailor the information on a page just for printing, in addition to normal web viewing?
- ▶ Hours 10 through 12 introduce you to multimedia applications and their use, including where to find industry-standard software you can download and try free.
- ▶ The technical stuff is not enough, so this book also includes the advice you need when setting up a web site to achieve your goals. Key details—designing an effective page layout, posting your page to the Internet with FTP software, organizing and managing multiple pages, and getting your pages to appear high on the query lists at all the major Internet search sites—are all covered in enough depth to get you beyond the snags that often frustrate beginners.

Attention to many of these essentials are what made the first seven editions of this book best-sellers, and this updated edition—the first for this title since 2005—is no different. All of the examples have been updated and a significant portion of the content has been revised to match new examples and new technologies.

Visual Examples

Every example in this book is illustrated in two parts:

- ▶ The text you type to make an HTML page is shown first, with all HTML and CSS code high-lighted.
- ▶ The resulting web page is shown as it will appear to users who view it with the world's most popular web browsers.

You'll often be able to adapt the example to your own pages without reading any of the accompanying text at all.

All the examples in this book are standards-compliant and work with Apple Safari, Google Chrome, Microsoft Internet Explorer, Mozilla Firefox, and Opera. While all of the screenshots are taken in Firefox, rest assured that all of the code has been tested in all other browsers.

You will also find the use of colors within code examples and when elements of code appear in the explanatory text. These colors highlight the different bits and pieces of code both to enhance your familiarity with them and to call attention to their use.

- ▶ HTML tags are displayed in dark blue.
- ► HTML comments are displayed in brown.
- CSS elements are displayed in green.
- ▶ HTML attribute names are displayed in light blue.
- ▶ HTML attribute values are displayed in magenta.

Be aware that the colors of certain terms change depending on their context. For instance, when CSS elements are used within the style attribute of an HTML tag, they will be color-coded as HTML attribute values (magenta) rather than CSS elements (green).

Special Elements

As you complete each hour, margin notes help you immediately apply what you just learned to your own web pages.

TIP	NOTE	WARNING
Tips and tricks to save you precious time are set aside in "Tip" boxes so that you can spot them quickly.	"Note" boxes provide additional information about the topics being discussed.	When there's something you need to watch out for, you'll be warned about it in "Warning" boxes.

Q&A, Quiz, and Exercises

Every hour ends with a short question-and-answer session that addresses the kind of "dumb questions" everyone wishes they dared to ask. A brief but complete quiz lets you test yourself to be sure you understand everything presented in the hour. Finally, one or two optional exercises give you a chance to practice your new skills before you move on.

HOUR 3

Understanding HTML and XHTML Connections

The first two hours gave you a basic idea of the process behind creating web content and viewing it online or locally, if you do not yet have a web hosting provider. In this hour, we'll get down to the business of explaining the various elements that must appear in an HTML file.

At the end of the hour, you'll learn how HTML differs from XHTML and why there are two different languages designed to do the same thing—create web content. In general, this hour provides a quick summary of HTML and XHTML basics and gives some practical tips to make the most of your time as a web page author and publisher. It's not all theory, however; you do get to see a real web page and the HTML code behind it.

Here's a review of what you need to do before you're ready to use the rest of this book:

- **1.** Get a computer. I used a computer with Windows Vista to test the sample web content and capture the figures in this book, but you can use any Windows, Macintosh, or Linux/UNIX machine to create and view your web content.
- 2. Get a connection to the Internet. Whether you have a dial-up, wireless, or broadband connection doesn't matter for the creation and viewing of your web content, but the faster the connection, the better for the overall experience. The Internet Service Provider (ISP), school, or business that provides your Internet connection can help you with the details of setting it up properly. Additionally, many public spaces such as coffee shops, bookstores, and libraries offer free wireless Internet service that you can use if you have a laptop computer with Wi-Fi network support.

WHAT YOU'LL LEARN IN THIS HOUR:

- ▶ How to create a simple web page in HTML
- ► How to include all the HTML Tags that every web page must have
- How to organize a page with paragraphs and line breaks
- ▶ How to organize your content with headings
- How to validate your web content
- How to differentiate between HTML, XML, XHTML, and HTML 5

NOTE

Not sure how to find an ISP? The best way is to comparisonshop online (using a friend's computer or a public computer that's already connected to the Internet). You'll find a comprehensive list of national and regional ISPs at http://www.thelist.com/.

NOTE

Although all web browsers process and handle information in the same general way, there are some specific differences among them that result in things not always looking the same in different browsers. Be sure to check your web pages in multiple browsers to make sure that they look reasonably consistent.

NOTE

As discussed in Hour 1, if you plan to put your web content on the Internet (as opposed to publishing it on CD-ROM or a local intranet), you'll need to transfer it to a computer that is connected to the Internet 24 hours a day. The same company or school that provides you with Internet access might also provide web space; if not, you might need to pay a hosting provider for the service.

WARNING

Do not create your first HTML file with Microsoft Word or any other HTML-compatible word processor; most of these programs attempt to rewrite your HTML for you in strange ways, potentially leaving you totally confused. Additionally, I recommend that you do not use a graphical, what-you-see-is-whatyou-get (WYSIWYG) editor, such as Microsoft FrontPage or Adobe Dreamweaver. You'll likely find it easier and more educational to start out with a simple text editor while you're just learning HTML. You can progress to visual tools (such as FrontPage and Dreamweaver) after you have a better understanding of what's going on "under the hood."

- **3.** Get web browser software. This is the software your computer needs in order to retrieve and display web content. As you learned in the first hour, the most popular browser software (in alphabetical order) is Apple Safari, Google Chrome, Mozilla Firefox, Microsoft Internet Explorer, and Opera. It's a good idea to install several of these browsers so that you can experiment and make sure that your content looks consistent across them all; you can't make assumptions about the browsers other people are using.
- **4.** Explore! Use a web browser to look around the Internet for web sites that are similar in content or appearance to those you'd like to create. Note what frustrates you about some pages, what attracts you and keeps you reading others, and what makes you come back to some pages over and over again. If there is a particular topic that interests you, consider searching for it using a popular search engine such as Google (http://www.google.com/) or Bing (http://www.bing.com/).

Getting Started with a Simple Web Page

In the first hour, you learned that a "web page" is just a text file that is "marked-up" by (or surrounded by) HTML codes that tell the browser how to display the text. To create these text files, use a *text editor* such as Notepad (on Windows) or TextEdit (on a Mac)—do not use WordPad, Microsoft Word, or other full-featured word-processing software, as those create different sorts of files than the plain-text files we use for web content.

Before you begin working, you should start with some text that you want to put on a web page:

- **1.** Find (or write) a few paragraphs of text about yourself, your family, your company, your softball team, or some other subject in which you're interested.
- 2. Save this text as plain, standard ASCII text. Notepad and most simple text editors always save files as plain text, but if you're using another program, you might need to choose this file type as an option (after selecting File, Save As).

As you go through this hour, you will add HTML markup (called *tags*) to the text file, thus making it into web content.

When you save files containing HTML tags, always give them a name ending in .html. This is important: if you forget to type the .html at the end of the filename when you save the file, most text editors will give it some other extension (such as .txt). If that happens, you might not be able to find the file when you try to look at it with a web browser; if you find it, it certainly won't display properly. In other words, web browsers expect a web page file to have a file extension of .html.

You might also encounter a web page with a file extension of .htm, which is also acceptable. You might find other file extensions used on the Web, such as .jsp (Java Server Pages), .asp (Microsoft Active Server Pages), or .php (PHP: Hypertext Preprocessor), but these file types use server-side technologies that are beyond the scope of HTML.

Listing 3.1 shows an example of text you can type and save to create a simple HTML page. If you opened this file with Firefox, you would see the page shown in Figure 3.1. Every web page you create must include the <html></html>, <head></head>, <title></title>, and <body></body> tag pairs.

NOTE

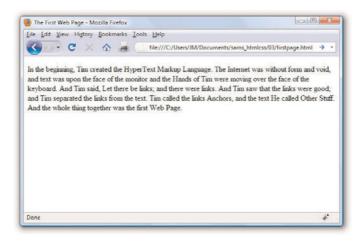
If you're using TextEdit on a Macintosh computer, the steps for creating an HTML file are a little different than for using Notepad on a Windows computer-both are popular text editors, but with the latter you must first click on the Format menu and select Make Plain Text and then change the preferences under the Saving header by unchecking the box for Append '.txt' Extension to Plain Text Files. Also, the default preferences are set to show .html documents as they would appear in a browser, which won't allow you to edit them. To fix this, check Ignore Rich Text Commands in HTML Files under the Rich Text Processing header.

Listing 3.1 The html, html, httpl://html, httpl://html,

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
  "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
 <head>
   <title>The First Web Page</title>
 </head>
 <body>
   >
      In the beginning, Tim created the HyperText Markup Language. The Internet
     was without form and void, and text was upon the face of the monitor and
     the Hands of Tim were moving over the face of the keyboard. And Tim said,
     Let there be links; and there were links. And Tim saw that the links were
      good; and Tim separated the links from the text. Tim called the links
     Anchors, and the text He called Other Stuff. And the whole thing together
      was the first Web Page.
   </body>
</html>
```

FIGURE 3.1

When you save the text in Listing 3.1 as an HTML file and view it with a web browser, only the actual title and body text are displayed.



NOTE

Technically speaking, HTML 5 will be the next web standard but it's not quite at the point of full adoption. Current estimate put the full adoption of HTML somewhere in the year 2011. However, as you learn about important features of HTML and XHTML in this book, I will include notes about how HTML 5 features might differ.

In Listing 3.1, as in every HTML page, the words starting with < and ending with > are actually coded commands. These coded commands are called HTML tags because they "tag" pieces of text and tell the web browser what kind of text it is. This allows the web browser to display the text appropriately.

The first few lines of code in the web page serve as standard boilerplate code that you will include in all of your pages. This code actually identifies the page as an XHTML 1.1 document, which means that, technically, the web page is an XHTML page. All the pages developed throughout the book are XHTML 1.1 pages. Because XHTML is a more structured version of HTML, it's still okay to generally refer to all the pages in the book as HTML pages. By targeting XHTML 1.1 with your code, you are developing web pages that adhere to the very latest web standards. This is a good thing!

Before you learn the meaning of the HTML tags used in Listing 3.1, you might want to see exactly how I went about creating and viewing the document itself. Follow these steps:

- Type all the text in Listing 3.1, including the HTML tags, in Windows Notepad (or use Macintosh TextEdit or another text editor of your choice).
- 2. Select File, Save As. Be sure to select plain text (or ASCII text) as the file type.
- 3. Name the file firstpage.html.
- 4. Choose the folder on your hard drive where you would like to keep your web pages—and remember which folder you choose! Click the Save or OK button to save the file.
- **5.** Now start your favorite web browser. (Leave Notepad running, too, so you can easily switch between viewing and editing your page.)

In Internet Explorer, select File, Open and click Browse. If you're using Firefox, select File, Open File. Navigate to the appropriate folder and select the firstpage.html file. Some browsers and operating systems will also allow you to drag and drop the firstpage.html file onto the browser window in order to view it.

Voilà! You should see the page shown in Figure 3.1.

If you have obtained a web hosting account, you could use FTP at this point to transfer the firstpage.html file to the web server. In fact, from this hour forward, the instructions will assume you have a hosting provider and are comfortable sending files back and forth via FTP; if that is not the case, please review the first two hours before moving on. Or, if you are consciously choosing to work with files locally (without a web host), be prepared to adjust the instructions to suit your particular needs (such as ignoring "transfer the files" and "type in the URL").

TRY IT YOURSELF ▼

Creating and Viewing a Basic Web Page

NOTE

You don't need to be connected to the Internet to view a web page stored on your own computer. By default, your web browser tries to connect to the Internet every time you start it, which makes sense most of the time. However, this can be a hassle if you're developing pages locally on your hard drive (offline) and you keep getting errors about a page not being found. If you have a full-time web connection via a LAN, cable modem, or DSL, this is a moot point because the browser will never complain about being offline. Otherwise, the appropriate disciplinary action will depend on your breed of browser: check the options under your browser's "Tools" menu.

NOTE

It isn't terribly important that you understand concepts such as character encoding at this point. What is important is that you include the appropriate boilerplate code in your pages so that they adhere to the latest web standards. As of this writing, XHTML 1.1 is a web standard. HTML 5 is not yet a web standard, but if you were creating an HTML 5 document, these lines at the beginning of your HTML file would not be necessary.

NOTE

The XML/XHTML boilerplate code isn't strictly required in order for you to create web pages. You can delete the opening lines of code in the example so that the page starts with the <html> tag and it will still open fine in a web browser. The extra code is included to ensure your pages are up to date with the current web standards. Additionally, the extra code allows you to validate your web pages for accuracy, which you'll learn how to do a bit later in this lesson.

HTML Tags Every XHTML Web Page Must Have

The time has come for the secret language of HTML tags to be revealed to you. When you understand this language, you will have creative powers far beyond those of other humans. Don't tell the other humans, but it's really pretty easy.

Before you get into the HTML tags, let's first address the messy-looking code at the top of Listing 3.1. The first line indicates that the HTML document is, in fact, an XML document:

```
<?xml version="1.0" encoding="UTF-8"?>
```

The version of XML is set to 1.0, which is fairly standard, as is the type of character encoding (UTF-8).

The second and third lines of code in Listing 3.1 are even more complicated looking:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```

Again, the specifics of this code aren't terribly important as long as you remember to include the code at the start of your pages. This code identifies the document as being XHTML 1.1, which then allows web browsers to make sure the code meets all the requirements of XHTML 1.1.

Most HTML tags have two parts: an *opening tag*, which indicates where a piece of text begins, and a *closing tag*, which indicates where the piece of text ends. Closing tags start with a / (forward slash) just after the < symbol. Another type of tag is the *empty tag*, which is unique in that it doesn't include a pair of matching opening and closing tags. Instead, an empty tag consists of a single tag that starts with a < and ends with a / just before the > symbol. Following is a quick summary of these three tags just to make sure you understand the role each of them plays:

- ► An *opening tag* is an HTML tag that indicates the start of an HTML command; the text affected by the command appears after the opening tag. Opening tags always begin with < and end with >, as in <html>.
- ▶ A closing tag is an HTML tag that indicates the end of an HTML command; the text affected by the command appears before the closing tag. Closing tags always begin with </ and end with >, as in </html>.

▶ An *empty tag* is an HTML tag that issues an HTML command without enclosing any text in the page. Empty tags always begin with < and end with />, as in

or /> and .

For example, the <body> tag in Listing 3.1 tells the web browser where the actual body text of the page begins, and </body> indicates where it ends. Everything between the <body> and </body> tags will appear in the main display area of the web browser window, as shown in Figure 3.1.

The very top of the browser window (refer to Figure 3.1) shows title text, which is any text that is located between <title> and </title>. The title text is also used to identify the page on the browser's Bookmarks or Favorites menu, depending on which browser you use. It's important to provide titles for your pages so that visitors to the page can properly bookmark them for future reference.

You will use the <body> and <title> tag pairs in every HTML page you create because every web page needs a title and body text. You will also use <html> and <head>, which are he other two tags shown in Listing 3.1. Putting <html> at the very beginning of a document simply indicates that the document is a web page. The </html> at the end indicates that the web page is over.

Within a page, there is a head section and a body section. Each section is identified by <head> and <body> tags. The idea is that information in the head of the page somehow describes the page but isn't actually displayed by a web browser. Information placed in the body, however, is displayed by a web browser. The <head> tag always appears near the beginning of the HTML code for a page, just after the opening <html> tag.

The <title> tag pair used to identify the title of a page appears within the head of the page, which means it is placed after the opening <head> tag and before the closing </head> tag. (Upcoming hours reveal some other advanced header information that can go between <head> and </head>, such as style sheet rules that are used to format the page.)

The tag used in Listing 3.1 encloses a paragraph of text. You should enclose your chunks of text in the appropriate container tags whenever possible.

NOTE

You no doubt noticed in Listing 3.1 that there is some extra code associated with the <html> tag. This code consists of two attributes (xmlns and xml:lang), which are used to specify additional information related to the tag. These two attributes are standard requirements of all XHTML web pages; the former defines the XML namespace, while the latter defines the language of the content. Throughout this book a standard namespace is defined, and the English language is used. If you are writing in a different language, replace the "en" (for English) with the language identifier relevant to vou.

TIP

You might find it convenient to create and save a *bare-bones page* (also known as a *skeleton* page) with just the opening and closing <html>, <head>, <title>, and <body> tags, similar to the document used in Listing 3.1. You can then open that document as a starting point whenever you want to make a new web page and save yourself the trouble of typing all those obligatory tags every time.

Organizing a Page with Paragraphs and Line Breaks

When a web browser displays HTML pages, it pays no attention to line endings or the number of spaces between words. For example, the top version of the poem shown in Figure 3.2 appears with a single space between all words, even though that's not how it's entered in Listing 3.2. This is because extra whitespace in HTML code is automatically reduced to a single space. Additionally, when the text reaches the edge of the browser window, it automatically wraps to the next line, no matter where the line breaks were in the original HTML file.

Listing 3.2 HTML Containing Paragraph and Line Breaks

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
  "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
   <title>The Advertising Agency Song</title>
 </head>
 <body>
     When your client's hopping mad,
     put his picture in the ad.
     If he still should prove refractory,
     add a picture of his factory.
   <hr />
     When your client's hopping mad, <br />
     put his picture in the ad.
    >
     If he still should prove refractory, <br />
     add a picture of his factory.
   </body>
</html>
```

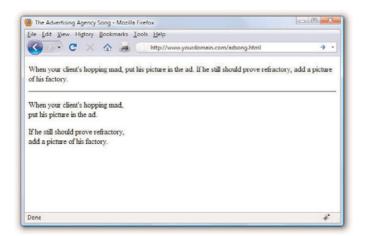


FIGURE 3.2

When the HTML in Listing 3.2 is viewed as a web page, line and paragraph breaks only appear where there are
br /> and tags.

You must use HTML tags if you want to control where line and paragraph breaks actually appear. When text is enclosed within the container tags, a line break will be assumed after the closing tag. In later hours, you will learn to control the height of the line break using CSS. The /> tag forces a line break within a paragraph. Unlike the other tags you've seen so far, /> doesn't require a closing tag—this is one of those empty tags discussed earlier. Although HTML 4 does not require the / in empty tags, XHTML does and future standards will, so it's important for you to stick to the latest standards and create web pages that are coded properly. Always code empty tags so that they end with />.

The poem in Listing 3.2 and Figure 3.2 shows the <code>
br</code> /> and tags being used to separate the lines and verses of an advertising agency song. You might have also noticed the <code><hr</code> /> tag in the listing, which causes a horizontal rule line to appear on the page (see Figure 3.2). Inserting a horizontal rule with the <code><hr</code> /> tag also causes a line break, even if you don't include a <code><br</code> /> tag along with it. Like <code><br</code> />, the <code><hr</code> /> horizontal rule tag is an empty tag and therefore never gets a closing <code></hr></code>

WARNING

You might find that a lot of web content that includes
 instead of
. Or you might find that it does not include the closing tag. Just remember there is a lot of antiquated web content floating around the Internet, and just because you see it in use doesn't mean it's correct. Save yourself a lot of future work and frustration by adhering to the standards you learn in this book. Developing clean HTML coding habits is a very important part of becoming a successful web designer.

▼ TRY IT YOURSELF

Formatting Text in HTML

WARNING

If you are using a word processor to create the web page, be sure to save the HTML file in plain-text or ASCII format.

Take a passage of text and try your hand at formatting it as proper HTML:

- **1.** Add <title>My Title</title></head><body> to the beginning of the text (using your own title for your page instead of My Title). Also include the boilerplate code at the top of the page that takes care of meeting the requirements of XHTML.
- **2.** Add </body></html> to the very end of the text.
- **3.** Add a tag at the beginning of each paragraph and a tag at the end of each paragraph.
- **4.** Use
 tags anywhere you want single-spaced line breaks.
- **5.** Use <hr /> to draw horizontal rules separating major sections of text, or wherever you'd like to see a line across the page.
- **6.** Save the file as *mypage*.html (using your own filename instead of mypage).
- **7.** Open the file in a web browser to see your web content. (Send the file via FTP to your web hosting account, if you have one.)
- **8.** If something doesn't look right, go back to the text editor to make corrections and save the file again (and send it to your web hosting account, if applicable). You then need to click Reload/Refresh in the browser to see the changes you made.

Organizing Your Content with Headings

When you browse through web pages on the Internet, you'll notice that many of them have a heading at the top that appears larger and bolder than the rest of the text. Listing 3.3 is sample code and text for a simple web page containing an example of a heading as compared to normal paragraph text. Any text between <h1> and </h1> tags will appear as a large heading. Additionally, <h2> and <h3> make progressively smaller headings, and so on as far down as <h6>.

Listing 3.3 Heading Tags

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
  "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
   <title>My Widgets</title>
 </head>
 <body>
   <h1>My Widgets</h1>
   My widgets are the best in the land. Continue reading to
   learn more about my widgets.
   <h2>Widget Features</h2>
   If I had any features to discuss, you can bet I'd do
   it here.
   <h3>Pricing</h3>
   Here, I would talk about my widget pricing.
   <h3>Comparisons</h3>
   Here, I would talk about how my widgets compare to my
   competitor's widgets.
 </body>
</html>
```

As you can see in Figure 3.3, the HTML that creates headings couldn't be simpler. In this example, the phrase "My Widgets" is prominently displayed using the <h1> tag. To create the biggest (level-1) heading, just put an <h1> tag at the beginning and a </h1> tag at the end of the text you wish to use as a heading. For a slightly smaller (level-2) heading—for information that is of lesser importance than the title— use the <h2> and </h2> tags around your text. For content that should appear even less prominently than a level-2 heading, use the <h3> and </h3> tags around your text. Your headings should follow a content hierarchy; use only one level-1 heading, have one (or more) level-2 headings after the level-1 heading, use level-3 headings directly after level-2 headings, and so on.

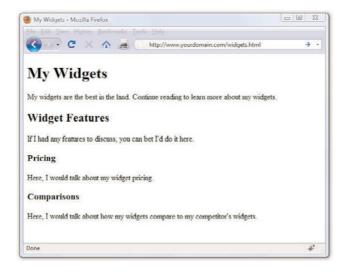
Theoretically, you can also use <h4>, <h5>, and <h6> tags to make progres-

NOTE

By now you've probably caught on to the fact that HTML code is often indented by its author to reveal the relationship between different parts of the HTML document. This indentation is entirely voluntary—you could just as easily run all the tags together with no spaces or line breaks and they would still look fine when viewed in a browser. The indentations are for you so that you can quickly look at a page full of code and understand how it fits together. Indenting your code is a very good web design habit and ultimately makes your pages easier to maintain.

FIGURE 3.3

The use of three levels of headings shows the hierarchy of content on this sample product page.



NOTE

On many web pages nowadays, graphical images of ornately rendered letters and logos are often used in place of the ordinary text headings discussed in this hour. However, using text headings is one of many search engine optimization (SEO) tips that you will learn about in Hour 24, "Helping People Find Your Web Pages." Search engines look at heading tags to see how you organize your content; they give higher preference to content that you have indicated is more important (for example, a level-1 heading) versus content that you indicate is of lesser importance (lower-level headings).

sively less important headings, but these aren't used very often. Web browsers seldom show a noticeable difference between these headings and the <h3> headings anyway, and content usually isn't displayed in such a manner as to need six levels of headings in order to show the content hierarchy.

It's important to remember the difference between a *title* and a *heading*. These two words are often interchangeable in day-to-day English, but when you're talking HTML, <title> gives the entire page an identifying name that isn't displayed on the page itself; it's displayed only on the browser window's title bar. The heading tags, on the other hand, cause some text on the page to be displayed with visual emphasis. There can be only one <title> per page and it must appear within the <head> and </head> tags, whereas you can have as many <h1>, <h2>, and <h3> headings as you want, in any order that suits your fancy. However, as I mentioned before, you should use the heading tags to keep tight control over content hierarchy; do not use headings as a way to achieve a particular "look," as that's what CSS is for.

You'll learn to take complete control over the appearance of text on your

web pages in Part II of this book. Short of taking exacting control of the size, family, and color of fonts, headings provide the easiest and most popular way to draw extra attention to important text.

Peeking at Other Designers' Pages

Given the visual and sometimes audio pizzazz present in many popular web pages, you probably realize that the simple pages described in this hour are only the tip of the HTML iceberg. Now that you know the basics, you might surprise yourself with how much of the rest you can pick up just by looking at other people's pages on the Internet. You can see the HTML for any page by right-clicking and selecting View Source in any web browser.

Don't worry if you aren't yet able to decipher what some HTML tags do or exactly how to use them yourself. You'll find out about all those things in the next few hours. However, sneaking a preview now will show you the tags that you do know in action and give you a taste of what you'll soon be able to do with your web pages.

Validating Your Web Content

In Hour 2, I discussed ways to test your pages; one very important way to test your pages is to *validate* them. Think of it this way: it's one thing to design and draw a beautiful set of house plans, but it's quite another for an architect to stamp it as a safe structure suitable for construction. Validating your web pages is a similar process; in this case, however, the architect is an application—not a person.

In brief, validation is the process of testing your pages with a special application that searches for errors and makes sure your pages follow the strict XHTML standard. Validation is simple. In fact, the standards body responsible for developing web standards—the World Wide Web Consortium (W3C)—offers an online validation tool you can use. To validate a page, follow this URL: http://validator.w3.org/. The W3C Markup Validation Service is shown in Figure 3.4.

WARNING

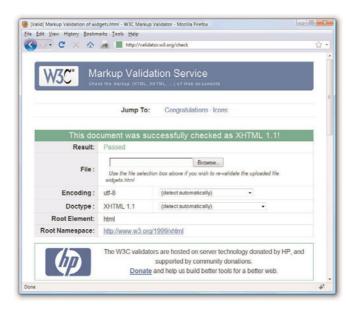
Don't forget that anything placed in the head of a web page is not intended to be viewed on the page, whereas everything in the body of the page is intended for viewing.

FIGURE 3.4
The W3C Markup Validation
Service allows you to validate an
HTML (XHTML) document to
ensure it has been coded
accurately.



If you've already published a page online, you can use the Validate by URI tab. Use the Validate by File Upload tab to validate files stored on your local computer file system. The Validate by Direct Input tab allows you to paste the contents of a file from your text editor. If all goes well, your page will get a passing report (see Figure 3.5).

FIGURE 3.5 If a page passes the W3C Markup Validation Service, you know it is ready for prime time.



If the W3C Markup Validation Service encounters an error in your web page, it will provide specific details (including the line numbers of the offending code). This is a great way to hunt down problems and rid your pages of buggy code. Validation not only informs you whether your pages are constructed properly, it also assists you in finding and fixing problems before you post pages for the world to see.

The Scoop on HTML, XML, XHTML, and HTML 5

In its early days, HTML was great because it allowed scientists to share information over the Internet in an efficient and relatively structured manner. It wasn't until later that graphical web browsers were created and HTML started being used to code more than scientific papers. HTML quickly went from a tidy little markup language for researchers to an online publishing language. After it was established that HTML could be jazzed up for graphical browsing, the creators of web browsers went crazy by adding lots of nifty features to the language. Although these new features were neat at first, they compromised the original design of HTML and introduced inconsistencies when it came to how browsers displayed web pages; new features worked on only one browser or another, and you were out of luck if you happened to be running the wrong browser. HTML started to resemble a bad remodeling job on a —a job done by too many contractors and without proper planning. As it turns out, some of the browser-specific features created during this time have now been adopted as standards while others have been dropped completely.

As with most revolutions, the birth of the Web was very chaotic, and the modifications to HTML reflected that chaos. Over the years, a significant effort has been made to reel in the inconsistencies of HTML and restore some order to the language. The problem with disorder in HTML is that it results in web browsers having to guess at how a page is to be displayed, which is not a good thing. Ideally, a web page designer should be able to define exactly how a page is to look and have it look the same regardless of what kind of browser or operating system someone is using. Better still, a designer should be able to define exactly what a page *means* and have that page look consistent across different browsers and platforms. This utopia is still off in the future somewhere, but a markup language called XML (Extensible Markup Language) began to play a significant role in leading us toward it.

XML is a general language used to create specific languages, such as HTML. It might sound a little strange, but it really just means that XML

TIP

tures you can use in lieu of the W3C Markup Validation Service. Some examples include browser extensions such as Firebug (http://getfirebug.com/) and HTML Validator (http://users.skynet.be/mgueury/mozilla/), but many other programs offer similar functionality—check your user documentation.

Some web development tools

include built-in validation fea-

provides a basic structure and set of rules to which any markup language must adhere. Using XML, you can create a unique markup language to describe just about any kind of information, including web pages. Knowing that XML is a language for creating other markup languages, you could create your own version of HTML using XML. You could even create a markup language called BCCML (Bottle Cap Collection Markup Language), for example, which you could use to create and manage your extensive collection of rare bottle caps. The point is that XML lays the ground rules for organizing information in a consistent manner, and that information can be anything from web pages to bottle caps.

You might be thinking that bottle caps don't have anything to do with the Web, so why mention them? The reason is that XML is not entirely about web pages. XML is actually broader than the Web in that it can be used to represent any kind of information on any kind of computer. If you can visualize all the information whizzing around the globe among computers, mobile phones, handheld computers, televisions, and radios, you can start to understand why XML has much broader applications than just cleaning up web pages. However, one of the first applications of XML is to restore some order to the Web, which is why XML is relevant to learning HTML.

If XML describes data better than HTML, does it mean that XML is set to upstage HTML as the markup language of choice for the Web? No. XML is not a replacement for HTML; it's not even a competitor of HTML. XML's impact on HTML has to do with cleaning up HTML. HTML is a relatively unstructured language that benefits from the rules of XML. The natural merger of the two technologies resulted in HTML's adherence to the rules and structure of XML. To accomplish this merger, a new version of HTML was formulated that follows the stricter rules of XML. The new XML-compliant version of HTML is known as XHTML. Fortunately for you, you'll actually be learning XHTML throughout this book since it is really just a cleaner version of HTML.

You might have heard about HTML 5, which is touted as the next web standard. It will be, but not for several years. When it does become a web standard, it will not render XHTML useless—HTML 5 is not a replacement for XHTML, but is a major revision of HTML 4. In other words, XHTML and HTML 5 can coexist on the web, and web browsers that currently support XHTML will also (one day) support HTML 5 as well.

The goal of this book is to guide you through the basics of web publishing, using XHTML and CSS as the core languages of those pages. However, whenever possible I will note elements of the languages that are not present in HTML 5, should you want to design your content for even further

sustainability. If you gain a solid understanding of web publishing and the ways in which CSS works with the overall markup language of the page (be it XHTML or HTML 5), you will be in a good position if, in a few years, you decide you want to move from XHTML to HTML 5.

Summary

This hour introduced the basics of what web pages are and how they work. You learned that coded HTML commands are included in a text file, and that typing HTML text yourself is better than using a graphical editor to create HTML commands for you—especially when you're learning HTML. You were introduced to the most basic and important HTML tags. By adding these coded commands to any plain-text document, you can quickly transform it into a bona fide web page. You learned that the first step in creating a web page is to put a few obligatory HTML tags at the beginning and end, including a title for the page. You then mark where paragraphs and lines end and add horizontal rules and headings if you want them. Table 3.1 summarizes all the tags introduced in this hour.

Table 3.1 HTML Tags Covered in Hour 3

Tag	Function
<html></html>	Encloses the entire HTML document.
<head></head>	Encloses the head of the HTML document. Used within the html tag pair.
<title></title>	Indicates the title of the document. Used within the <head> tag pair.</head>
<body></body>	Encloses the body of the HTML document. Used within the html tag pair.
	A paragraph; skips a line between paragraphs.
 	A line break.
<hr/>	A horizontal rule line.
<h1></h1>	A first-level heading.
<h2></h2>	A second-level heading.
<h3></h3>	A third-level heading.
<h4></h4>	A fourth-level heading (seldom used).
<h5></h5>	A fifth-level heading (seldom used).
<h6></h6>	A sixth-level heading (seldom used).

Finally, you learned about XML and XHTML, and how they relate to HTML, as well as what "HTML 5" means in relation to what it is you're learning here.

Q&A

- Q I've created a web page, but when I open the file in my web browser, I see all the text including the HTML tags. Sometimes I even see weird gobbledygook characters at the top of the page! What did I do wrong?
- A You didn't save the file as plain-text. Try saving the file again, being careful to save it as Text Only or ASCII Text. If you can't quite figure out how to get your word processor to do that, don't stress. Just type your HTML files in Notepad or TextEdit instead and everything should work just fine. (Also, always make sure that the filename of your web page ends in .html or .htm.)
- Q I've seen web pages on the Internet that don't have <html> tags at the beginning. You said pages always have to start with <html>. What's the deal?
- A Many web browsers will forgive you if you forget to include the <html>
 tag and will display the page correctly anyway. However, it's a very good
 idea to include it because some software does need it to identify the
 page as valid HTML. Besides, you want your pages to be bona fide
 XHTML pages so that they conform to the latest web standards.

Workshop

The workshop contains quiz questions and activities to help you solidify your understanding of the material covered. Try to answer all questions before looking at the "Answers" section that follows.

Quiz

- **1.** What four tags are required in every HTML page?
- **2.** What HTML tags and text would you use to produce the following web content:
 - ▶ A small heading with the words **We are Proud to Present**
 - ▶ A horizontal rule across the page
 - ► A large heading with the one word **Orbit**

- ▶ A medium-sized heading with the words **The Geometric Juggler**
- ▶ Another horizontal rule
- **3.** What code would you use to create a complete HTML web page with the title **Foo Bar**, a heading at the top that reads **Happy Hour at the Foo Bar**, followed by the words **Come on down!** in regular type?

Answers

- https://www.neads.com/, <a href="https://www
- 2. Your code would look like this:

```
<h3>We are Proud to Present</h3>
<hr />
<h1>0rbit</h1>
<h2>The Geometric Juggler</h2>
<hr />
```

3. Your code would look like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
  "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
        <title>Foo Bar</title>
        </head>
        <body>
            <h1>Happy Hour at the Foo Bar</h1>
            Come on Down!
            </body>
        </html>
```

Exercises

- ▶ Even if your main goal in reading this book is to create web content for your business, you might want to make a personal web page just for practice. Type a few paragraphs to introduce yourself to the world and use the HTML tags you've learned in this hour to make them into a web page.
- ▶ Throughout the book you'll be following along with the code examples and making pages of your own. Take a moment now to set up a basic document template containing the XML declaration, doctype declaration, and tags for the core HTML document structure. That way, you can be ready to copy and paste that information whenever you need it.

INDEX

text-align property, 57-58, 223

attributes

align, 69

CSS, 224

Symbols

@import command, 316 !— — tags (HTML), 364-365, 392 !doctype tags, 392	text-align property, 57-58, 223 vertical-align property, 223 graphics, 165 HTML tables, 102-104 images horiztonal alignment, 165-167	align, 69 default values, 69 HTML tags, 111 XHTML common atttributes, quick reference, 414-415 audio. See multimedia
A	vertical alignment, 167-169	
a href tags, 169	text, 68	В
a tags (anchors), 114, 184-185, 400-	page sections, 69-71	В
401	paragraphs, 69	b tags, 81-83, 398
psuedoclasses, 123	AllTheWeb, 374	backgrounds
web pages	alt attribute	banner creation, 152
identifying page locations, 114	area tags, 176	color, 133-138
linking to page locations, 114-116	img tags, 162-163	CSS
absolute addresses, 112-113	AltaVista, 374	background property, 419
absolute positioning,	alternative text, images, 163	background style properties,
54, 235, 238-239	Amazon.com, website organization, 361	quick reference, 419-420
absolute value (position property),	analogous color schemes, 132	background-attachment property, 419
235	anchors (a tags), 184-185	background-color attribute
accesskey attribute, 414	psuedoclasses, 123	(tables), 104
action attribute	web pages	background-color property, 56,
form tags, 340-341	identifying page locations, 114	419
input tags, 349 Adobe Photoshop, 143	linking to page locations, 114-116	background-color style (body tags), 133, 135
advanced developer resources, 388 aligning	animated web graphics, 156	background-image attribute (tables), 104
align attribute, 69, 393	animations. See multimedia	background-image property, 419
img tags, 168	area / tags, 407	background-position property,
tables, 102	area tags, 176-177	419
	areas (imagemaps), 174-175	background-repeat property, 420
	ASP (Active Server Pages), 326	
	.asp file extension, 29	

border-style property, 55-56, 422	C
border-top property, 55-56, 422	cameras (digital), exporting images to computer, 146
border-top-color property, 422	caption tags, 402
border-top-style property, 422	caption-side property (CSS), 428
border-top-width property, 422	cellpadding attribute (tables), 104
border-width property,	cells (tables), defining, 96
55-56, 423	cellspacing attribute (tables), 104
changing color, 136-138	character entities (HTML), 90
CSS box model, 232	check boxes, HTML forms, 346-348
HTML lists, 248	cite tags, 396
bottom property (CSS), 235, 425	class attribute, 414
bottom value (vertical-align property), 224	Classic FTP (File Transfer Protocol), operating, 16-18
br / tags, 396	classid attribute (object tags), 187
br /> tags (HTML), 35	clear property (CSS), 228, 242, 425
br tags, 84	client-side scripting, 326
browsers	clip art, 144, 161-163
client-side scripting, 326	clip property (CSS), 425
color, 139	closing tags (HTML), 32
comparing, 28	code listings, providing hints for
CSS, viewing style rules, 53	search sites, 377
distributing, 23	code tags, 396
helper applications, 185	col / tags, 402
HTML lists, formatting, 72	colgroup tags, 403
links, opening in new windows,	color
	analogous color schemes, 132
	backgrounds
5	color, 133-134
G .	CSS, 136-138
	borders (CSS), 136-138
o .	browser-safe colors, 139
o .	Color Scheme Generator, 133
	color wheel, 132
'	complementary color schemes, 132
	CSS, 136-138
	dithering, 153
	hexadecimal values, 135-136
	HTML
= '	adding color to tables, 104
, or odding, 202 202	lists, 248
	images
	correcting in, 149-150
	border-top property, 55-56, 422 border-top-color property, 422 border-top-style property, 422 border-top-width property, 422 border-width property, 55-56, 423 changing color, 136-138 CSS box model, 232 HTML lists, 248 bottom property (CSS), 235, 425 bottom value (vertical-align property), 224 br / tags, 396 br /> tags (HTML), 35 br tags, 84 browsers client-side scripting, 326 color, 139 comparing, 28 CSS, viewing style rules, 53 distributing, 23 helper applications, 185 HTML lists, formatting, 72

reducing color in, 153

text, 56, 85-88	retrieving, 5	image maps, creating, 251-254
changing color, 134	searching, 372-373	internal style sheets, 48, 62-63
CSS, 136-138	testing, 24	layout style properties, quick
triadic color schemes, 132	coords attribute (area tags), 176	reference, 424-427
web resources, 387	copyright symbol (©), 90, 144	links
websites	Corel DRAW, 144	displaying pseudoclasses, 125
background color, 133-134	Creative Commons licenses, images,	styling, 123-126
hexadecimal values, 135-136	144	list style properties, quick
selecting for, 131-132	cropping images, 146-147	reference, 427-428
text color, 134	CSS (Cascading Style Sheets), 48	margins
color attribute	alignment, 223-224	box model, 232
font tags, 85	backgrounds	margin property, 212-214
html tags, 96	changing color, 136-138	margin style properties, quick
color property (CSS), 56-58, 416	style properties, quick	reference, 423
color style (font tags), 85	reference, 419-420	margin left property, 212
cols attribute	borders	margin-left property, 212
frameset tags, 200	box model, 232	margin-right property, 212
textarea tags, 349	changing color, 136-138	margin-top property, 212
colspan attribute (th tags), 104	style properties, quick	properties of, 212
columns (tables)	reference, 420-423	styling, 217-219
color, adding to columns, 104	box model, 231-233	writing in shorthand, 213-214
fixed/liquid hybrid layouts, 300- 309	borders, 232 content, 232	marker style properties, quick reference, 427-428
spanning, 104	DOCTYPE declaration, 234	navigation
comments	HTML lists, 246-249	horizontal navigation, 270-273
documenting HTML code via,	margins, 232	navigation lists, 259
364-365	padding, 232	primary navigation, 260
hiding web page sections via, 365	clear property, 228, 242	secondary navigation, 260
common atttributes, XHTML quick	color, 136-138	vertical navigation, 260-269
reference, 414-415	.css file extension, 48	overflow property, 242
complementary color schemes, 132	current line, 242	overlapping elements, 235
complex (large) websites, organizing,	development of, 49	padding, 219-222
360-362	dimension style properties, quick	box model, 232
compression, .JPEG files, 150	reference, 415-416	padding property, 220
content (web)	display style properties, quick	padding style properties, quick
creating, 2-3	reference, 424-427	reference, 424
delivering, process of, 4	event handling, 283-284	padding-bottom property, 220
publishing	external style sheets, 48-51	padding-left property, 220
blogs, 23	float property, 224-227, 242	padding-right property, 220
locally, 23	font style properties, quick reference, 416-418	padding-top property, 220
operating a FTP client, 16-18	formatting text, 88	positioning
organizing web server directo- ries, 18-20	history of, 49	absolute positioning, 235, 238-239
selecting a FTP client, 14-15	HTML	bottom property, 235
web server index files, 20-22	linking to documents, 52	left property, 235
without web servers, 22	tags, 51	

without web servers, 22

position property, 235 tooltips, creating, 277-280 dt tags, 71, 246, 399 relative positioning, 235-237 validating, 63 dummy tags (HTML), 63 right property, 235 web resources, 385 dynamic websites top property, 235 z-index property, 235, 239-241 defining, 325 print-friendly web pages CSS Zen Garden, liquid layouts, 293 DOM, 332 media-specific style sheets, current line (CSS), 242 scripting 315-317 cursor property (CSS), 425 changing images based on style sheet design, 317-320 user interaction via custom HTML tags, creating, 60 JavaScript, 333 quick reference client-side scripting, 326 background style properties, 419-420 displaying random content via D JavaScript, 328-332 border style properties, dashed values (CSS border-style 420-423 JavaScript in HTML, 326-327 property), 56 dimension style properties, server-side scripting, 326 dd tags, 71-73, 246, 399 415-416 default values (attributes), 69 display style properties, 424-427 definition lists (HTML), 73, 246 E font style properties, 416-418 deprecated attributes, align, 393 em tags, 82-83 layout style properties, deprecated HTML tags, 83 email addresses, linking to, 120-121 424-427 developers, advanced developer embedded content list style properties, 427-428 resources, 388 embed tags (HTML), 187, 190 margin style properties, 423 digital cameras, exporting images to computer, 146 multimedia, 187-190 marker style properties, 427-428 dire attribute, 415 XHTML, quick reference, 407-409 padding style properties, 424 direction property (CSS), 417 empty tags (HTML), 32 table style properties, 428-429 directories empty-cells property (CSS), 429 text style properties, 416-418 ESPN.com, website organization, 359 backslashes (\), 112 rollover text, displaying, 281-282 double dot command (..), 113 stacking elements, 239-241 forward slashes (/), 112 handling, mouse actions and, 283-284 style classes, 58-61 display property (CSS), 54, 425 onclick, 415 style ID, 61 dithering, 153 ondblclick, 415 style properties div tags, 165, 232-233, 284-289, 393 onkeydown, 415 defining, 48 dl tags, 71-73, 246, 399 formatting properties, 55-58 onkeypress, 415 DOCTYPE declaration, CSS box model, 234 onkeyup, 415 layout properties, 54-55 !doctype... tags, 392 onmousedown, 415 style rules, 59 document roots, 17-19, 112 onmousemove, 415 defining, 48 DOM (Document Object Model), 332 onmouseout, 415 viewing in web browsers, 53 onmouseover, 415 style sheets, 48-53 dotted value (CSS border-style onmouseup, 415 support for, 53 property), 56 external style sheets (CSS), 48-51 table style properties, quick double dot command (..), directories, reference, 428-429 113 double value (CSS border-style changing color, 136, 138 property), 56 managing flow, 242-243

DRAW (Corel), 144

text style properties, quick reference, 416-418

color-correcting images in, 149

cropping images in, 146-147

installing, 144

F	border-color property, 55-56	selection lists, 348
•	border-left property, 55-56	text areas, 349
fieldset tags, 411	border-right property, 55-56	naming data, 345
file extensions	border-style property, 55-56	submitting data, 340, 349
.asp, 29	border-top property, 55-56	forms (XHTML), quick reference,
.css, 48	border-width property, 55-56	410-413
.html, 29	color property, 56, 58	forward slashes (/), directories, 112
files	font property, 58	frames, 156
saving, 29	font-family property, 57	advantages/disadvantages of,
sound, streaming, 194	font-size property, 57	198
transfers. See FTP	font-style property, 57	compatiblity, 201
(File Transfer Protocol)	font-weight property, 57	defining, 197
fixed layouts, 294	line-height property, 58	frame tags (HTML), 202
fixed/liquid hybrid layouts	padding property, 58	framesets, 199
basic structure of, 298-299	text-align property, 57-58	adding individual frames, 202
defining columns in, 300-302	text-decoration property, 58	creating frameset documents,
handling column height in,	text-indent property, 57	199-201
304-309	HTML lists, 246-249	frameset documents, 203
setting minimum width of a layout, 302	text	frameset tags (HTML), 200- 202
Flickr website, 161-163	boldface text, 82-85	inline frames, 204-206
float property (CSS),	color, 85-88	linking with windows, 202-203
224-227, 242, 426	CSS, 88	naming, 202
float style property (img tags), 166	font size, 85-88	splitting, 200
flow (text), managing in CSS, 242-243	HTML web pages, 36	windows, sizing frames for, 200
fluid layouts. See liquid layouts	italic text, 82-85	FTP (File Transfer Protocol)
fonts (text)	line breaks, 84	Classic FTP, operating, 16-18
CSS font style properties, quick	monospaced text, 83-84	clients
reference, 416-418	spacing, 84	operating, 16-18
font property (CSS), 58, 417	special characters, 89-91	selecting, 14-15
font tags, 81, 85	special formatting, 82-85	Selecting, 14-15
font-family property (CSS), 57, 417	XHTML, quick reference,	
font-family style (font tags), 85	398-399	G
font-size property (CSS), 57, 417	websites, color, 131-136	
font-size style (font tags), 85	forms (HTML)	.GIF (Graphics Interchange Format)
font-style property (CSS), 57, 417	accepting text input, 344-345	files, 153
font-variant property (CSS), 417	creating, 340-344	animated web images, 156
font-weight property (CSS), 57, 417	defining, 339	tiled backgrounds, 154
size, 85-88	form-processing scripts, 340, 345	transparent images, 154
form tags, 340-341, 344, 411	hidden data, 345	GIMP (GNU Image Manipulation
formatting	input controls	Program)
formatting properties (CSS)	check boxes, 346-348	banner creation, 151-152
background-color property, 56	pull-down pick lists, 348	button creation, 151-152

radio buttons, 348

scrolling lists, 348

border property, 55

border-bottom property, 55-56

.JPEG file compression, 150	resolution, 145	hidden attribute (input tags), 345
scaling images in, 148	scaling, 148	hidden data in HTML forms, 345
Google, 374	scanners, getting graphics from,	hiding web pages via comments, 365
Google Images website, 161	146	home (main) pages (websites), index
graphics	sizing, 165	files, 22
aligning	software	horizontal aligning images, 165
horizontal alignment, 165-167	Adobe Photoshop, 143	horizontal navigation (CSS), 270-273
vertical alignment, 167-169	Corel DRAW, 144	horizontally aligning images, 165-167
alternative text descriptions, 163	GIMP, 144-152	hosting (web)
animated web graphics, 156	Picasa, 144	video hosting services, 191
background graphics, 144, 172-	Picnik, 144	web pages, selecting providers,
173	thumbnail graphics, turning into	6-7, 28
banners, creating, 151-152	links, 169-172	web resources, 389
buttons, creating, 151-152	tiled backgrounds, 154-155	hot linking images, 163
clip art, 144	transparent graphics, 154	hr / tags, 394
color reduction, 153	web pages, adding graphics to,	hr /> tags (HTML), 35
color-correcting, 149-150	162-163	href attribute (a tags), 114-117
copyrights, 144	web resources, 387	href attribute (area tags), 176
Creative Commons licences, 144	wrapping text around, 166	href atttribute (a tags), 184
cropping, 146-147	groove value (CSS border-style property), 56	HTML (Hypertext Markup Language)
digital cameras, exporting images	property), 30	character entities, 90
to computer, 146		CSS, linking to, 52
extracting from web pages, 144	H	development of, 2, 41
Flickr website, 161-163		files, creating, 29
.GIF files, 153	h1 tags, 393, 396	forms
animated web graphics, 156	h2 tags, 393, 396	accepting text input, 344-345
tiled backgrounds, 154	h3 tags, 393, 396	check boxes, 346-348
transparent graphics, 154	h4 tags, 393, 396	creating, 340-344
Google Images website, 161	h5 tags, 393, 396	defining, 339
graphical editors, 28	h6 tags, 393, 396	form-processing scripts, 340,
hot linking, 163	head tags, 33, 394	345
imagemaps	headings	hidden data, 345
areas, 174-175	boldface text, 82	input controls, 346-349
creaating HTML for, 176-178	heading tags (HTML), 36-38	naming data, 345
mapping regions within	height	pull-down pick lists, 348
graphics, 174-175	column height, handling in	radio buttons, 348
uses for, 173-174	fixed/liquid hybrid layouts, 304-309	scrolling lists, 348
.JPEG files		selection lists, 348
compression, 150	height attribute	submitting data, 340, 349
tiled backgrounds, 154	object tags, 188	text areas, 349
links, 123, 169-172	table tags, 99	history of, 2, 41
permissions, 163	height property (CSS), 55, 415	.html file extension, 29
photographs, 146	images, specifying in, 165	imagemaps, creating HTML for, 176-178
.PNG files, 153-154	helper applications, 185	
resizing, 148	hexadecimal values (color), 135-136	JavaScript in, 326-327

images 437

links, email addresses, 121	br /> tags, 35	sub tags, 83
lists	closing tags, 32	sup tags, 83
building outlines via, 73-74	CSS, 51	table tags, 99, 365
bullets, 73-74	custom tags, 60	td tags, 96-102
creating image maps via,	dd tags, 71-73, 246	textarea tags, 349
251-254	default attribute values, 69	th tags, 96, 101-102
CSS box model and, 246-249	deprecated tags, 83	title tags, 33
defintion lists, 246	div tags, 165, 232-233,	tr tags, 96-97, 102
formatting, 246-249	284-289	tt tags, 83-84
formatting in Web browsers, 72	dl tags, 71-73, 246	u tags, 83
nesting, 73-77	dt tags, 71, 246	ul tags, 71, 75, 246-248
ordered lists, 71-73, 245	dummy tags, 63	viewing, 39
placing item indicators,	em tags, 82-83	web pages
249-250	embed tags, 187, 190	formatting text, 36
unordered lists, 71-73, 245	empty tags, 32	line breaks, 34-35
maintainable code, writing	font tags, 81, 85	paragraph breaks, 34-35
documenting code via	form tags, 340-341, 344	XHTML web pages, 32-33
comments, 364-365	frame tags, 202	web resources, 385
indenting code, 366	frameset tags, 200-202	HTML 5 (Hypertext Markup Language
positioning	head tags, 33	version 5), 42
absolute positioning, 235, 238-239	heading tags, 36-38	hue (images), adjusting, 149-150
relative positioning, 235-237	hr /> tags, 35	hyperlinks. See links
tables	html tag, 394	
	i tags, 81-83	
adding color to 104	1 tags, 01 00	_
adding color to, 104	iframe tags, 204	1
aligning within, 102-104	_	-
aligning within, 102-104 cells, 96	iframe tags, 204	i tags, 81-83, 398
aligning within, 102-104 cells, 96 creating, 96-99	iframe tags, 204 img tags, 101, 162-163,	i tags, 81-83, 398 id attribute, 114-116, 414
aligning within, 102-104 cells, 96 creating, 96-99 sizing, 99-101	iframe tags, 204 img tags, 101, 162-163, 166-167, 365	i tags, 81-83, 398 id attribute, 114-116, 414 iframe tags (HTML), 204
aligning within, 102-104 cells, 96 creating, 96-99 sizing, 99-101 spanning within, 102-104	iframe tags, 204 img tags, 101, 162-163, 166-167, 365 input tags, 344-345	i tags, 81-83, 398 id attribute, 114-116, 414
aligning within, 102-104 cells, 96 creating, 96-99 sizing, 99-101 spanning within, 102-104 web page layouts, 105	iframe tags, 204 img tags, 101, 162-163, 166-167, 365 input tags, 344-345 li tags, 75, 246 link tags, 316 map tags, 177	i tags, 81-83, 398 id attribute, 114-116, 414 iframe tags (HTML), 204 image maps, creating via HTML lists, 251-254
aligning within, 102-104 cells, 96 creating, 96-99 sizing, 99-101 spanning within, 102-104 web page layouts, 105 tags, 29-30	iframe tags, 204 img tags, 101, 162-163, 166-167, 365 input tags, 344-345 li tags, 75, 246 link tags, 316 map tags, 177 meta tags, 375-377	i tags, 81-83, 398 id attribute, 114-116, 414 iframe tags (HTML), 204 image maps, creating via HTML lists, 251-254 images
aligning within, 102-104 cells, 96 creating, 96-99 sizing, 99-101 spanning within, 102-104 web page layouts, 105 tags, 29-30 !—— tags, 364-365	iframe tags, 204 img tags, 101, 162-163, 166-167, 365 input tags, 344-345 li tags, 75, 246 link tags, 316 map tags, 177	i tags, 81-83, 398 id attribute, 114-116, 414 iframe tags (HTML), 204 image maps, creating via HTML lists, 251-254 images aligning
aligning within, 102-104 cells, 96 creating, 96-99 sizing, 99-101 spanning within, 102-104 web page layouts, 105 tags, 29-30 !— — tags, 364-365 a href tags, 169	iframe tags, 204 img tags, 101, 162-163, 166-167, 365 input tags, 344-345 li tags, 75, 246 link tags, 316 map tags, 177 meta tags, 375-377	i tags, 81-83, 398 id attribute, 114-116, 414 iframe tags (HTML), 204 image maps, creating via HTML lists, 251-254 images aligning horizontal alignment, 165-167
aligning within, 102-104 cells, 96 creating, 96-99 sizing, 99-101 spanning within, 102-104 web page layouts, 105 tags, 29-30 !— — tags, 364-365 a href tags, 169 a tags (anchors), 114-116,	iframe tags, 204 img tags, 101, 162-163, 166-167, 365 input tags, 344-345 li tags, 75, 246 link tags, 316 map tags, 177 meta tags, 375-377 name tags, 202	i tags, 81-83, 398 id attribute, 114-116, 414 iframe tags (HTML), 204 image maps, creating via HTML lists, 251-254 images aligning
aligning within, 102-104 cells, 96 creating, 96-99 sizing, 99-101 spanning within, 102-104 web page layouts, 105 tags, 29-30 !— — tags, 364-365 a href tags, 169 a tags (anchors), 114-116, 123, 184-185	iframe tags, 204 img tags, 101, 162-163, 166-167, 365 input tags, 344-345 li tags, 75, 246 link tags, 316 map tags, 177 meta tags, 375-377 name tags, 202 noframes tags, 201 ol tags, 71, 75, 246 opening tags, 32	i tags, 81-83, 398 id attribute, 114-116, 414 iframe tags (HTML), 204 image maps, creating via HTML lists, 251-254 images aligning horizontal alignment, 165-167 vertical alignment, 167-169
aligning within, 102-104 cells, 96 creating, 96-99 sizing, 99-101 spanning within, 102-104 web page layouts, 105 tags, 29-30 !— — tags, 364-365 a href tags, 169 a tags (anchors), 114-116, 123, 184-185 area tags, 176-177	iframe tags, 204 img tags, 101, 162-163, 166-167, 365 input tags, 344-345 li tags, 75, 246 link tags, 316 map tags, 177 meta tags, 375-377 name tags, 202 noframes tags, 201 ol tags, 71, 75, 246 opening tags, 32 option tags, 348	i tags, 81-83, 398 id attribute, 114-116, 414 iframe tags (HTML), 204 image maps, creating via HTML lists, 251-254 images aligning horizontal alignment, 165-167 vertical alignment, 167-169 alternative text descriptions, 163 animated web images, 156
aligning within, 102-104 cells, 96 creating, 96-99 sizing, 99-101 spanning within, 102-104 web page layouts, 105 tags, 29-30 !— — tags, 364-365 a href tags, 169 a tags (anchors), 114-116, 123, 184-185 area tags, 176-177 attributes, 68, 111	iframe tags, 204 img tags, 101, 162-163, 166-167, 365 input tags, 344-345 li tags, 75, 246 link tags, 316 map tags, 177 meta tags, 375-377 name tags, 202 noframes tags, 201 ol tags, 71, 75, 246 opening tags, 32 option tags, 348 p tags, 33, 68-69, 98	i tags, 81-83, 398 id attribute, 114-116, 414 iframe tags (HTML), 204 image maps, creating via HTML lists, 251-254 images aligning horizontal alignment, 165-167 vertical alignment, 167-169 alternative text descriptions, 163
aligning within, 102-104 cells, 96 creating, 96-99 sizing, 99-101 spanning within, 102-104 web page layouts, 105 tags, 29-30 !—— tags, 364-365 a href tags, 169 a tags (anchors), 114-116, 123, 184-185 area tags, 176-177 attributes, 68, 111 b tags, 81-83	iframe tags, 204 img tags, 101, 162-163, 166-167, 365 input tags, 344-345 li tags, 75, 246 link tags, 316 map tags, 177 meta tags, 375-377 name tags, 202 noframes tags, 201 ol tags, 71, 75, 246 opening tags, 32 option tags, 348 p tags, 33, 68-69, 98 pre tags, 83-84	i tags, 81-83, 398 id attribute, 114-116, 414 iframe tags (HTML), 204 image maps, creating via HTML lists, 251-254 images aligning horizontal alignment, 165-167 vertical alignment, 167-169 alternative text descriptions, 163 animated web images, 156 backgrounds, 144, 154, 172-173
aligning within, 102-104 cells, 96 creating, 96-99 sizing, 99-101 spanning within, 102-104 web page layouts, 105 tags, 29-30 !— — tags, 364-365 a href tags, 169 a tags (anchors), 114-116, 123, 184-185 area tags, 176-177 attributes, 68, 111 b tags, 81-83 big tags, 83	iframe tags, 204 img tags, 101, 162-163, 166-167, 365 input tags, 344-345 li tags, 75, 246 link tags, 316 map tags, 177 meta tags, 375-377 name tags, 202 noframes tags, 201 ol tags, 71, 75, 246 opening tags, 32 option tags, 348 p tags, 33, 68-69, 98 pre tags, 83-84 select tags, 348	i tags, 81-83, 398 id attribute, 114-116, 414 iframe tags (HTML), 204 image maps, creating via HTML lists, 251-254 images aligning horizontal alignment, 165-167 vertical alignment, 167-169 alternative text descriptions, 163 animated web images, 156 backgrounds, 144, 154, 172-173 banners, creating, 151-152
aligning within, 102-104 cells, 96 creating, 96-99 sizing, 99-101 spanning within, 102-104 web page layouts, 105 tags, 29-30 !— — tags, 364-365 a href tags, 169 a tags (anchors), 114-116, 123, 184-185 area tags, 176-177 attributes, 68, 111 b tags, 81-83 big tags, 83 blockquote tags, 73	iframe tags, 204 img tags, 101, 162-163, 166-167, 365 input tags, 344-345 li tags, 75, 246 link tags, 316 map tags, 177 meta tags, 375-377 name tags, 202 noframes tags, 201 ol tags, 71, 75, 246 opening tags, 32 option tags, 348 p tags, 33, 68-69, 98 pre tags, 83-84 select tags, 348 small tags, 83	i tags, 81-83, 398 id attribute, 114-116, 414 iframe tags (HTML), 204 image maps, creating via HTML lists, 251-254 images aligning horizontal alignment, 165-167 vertical alignment, 167-169 alternative text descriptions, 163 animated web images, 156 backgrounds, 144, 154, 172-173 banners, creating, 151-152 buttons, creating, 151-152
aligning within, 102-104 cells, 96 creating, 96-99 sizing, 99-101 spanning within, 102-104 web page layouts, 105 tags, 29-30 !—— tags, 364-365 a href tags, 169 a tags (anchors), 114-116, 123, 184-185 area tags, 176-177 attributes, 68, 111 b tags, 81-83 big tags, 83 blockquote tags, 73 body tags, 33	iframe tags, 204 img tags, 101, 162-163, 166-167, 365 input tags, 344-345 li tags, 75, 246 link tags, 316 map tags, 177 meta tags, 375-377 name tags, 202 noframes tags, 201 ol tags, 71, 75, 246 opening tags, 32 option tags, 348 p tags, 33, 68-69, 98 pre tags, 83-84 select tags, 348 small tags, 83 strike tags, 83	i tags, 81-83, 398 id attribute, 114-116, 414 iframe tags (HTML), 204 image maps, creating via HTML lists, 251-254 images aligning horizontal alignment, 165-167 vertical alignment, 167-169 alternative text descriptions, 163 animated web images, 156 backgrounds, 144, 154, 172-173 banners, creating, 151-152 buttons, creating, 151-152 changing based on user interac-
aligning within, 102-104 cells, 96 creating, 96-99 sizing, 99-101 spanning within, 102-104 web page layouts, 105 tags, 29-30 !— — tags, 364-365 a href tags, 169 a tags (anchors), 114-116, 123, 184-185 area tags, 176-177 attributes, 68, 111 b tags, 81-83 big tags, 83 blockquote tags, 73	iframe tags, 204 img tags, 101, 162-163, 166-167, 365 input tags, 344-345 li tags, 75, 246 link tags, 316 map tags, 177 meta tags, 375-377 name tags, 202 noframes tags, 201 ol tags, 71, 75, 246 opening tags, 32 option tags, 348 p tags, 33, 68-69, 98 pre tags, 83-84 select tags, 348 small tags, 83 strike tags, 83 strong tags, 82-83	i tags, 81-83, 398 id attribute, 114-116, 414 iframe tags (HTML), 204 image maps, creating via HTML lists, 251-254 images aligning horizontal alignment, 165-167 vertical alignment, 167-169 alternative text descriptions, 163 animated web images, 156 backgrounds, 144, 154, 172-173 banners, creating, 151-152 buttons, creating, 151-152 changing based on user interaction via JavaScript, 333
aligning within, 102-104 cells, 96 creating, 96-99 sizing, 99-101 spanning within, 102-104 web page layouts, 105 tags, 29-30 !—— tags, 364-365 a href tags, 169 a tags (anchors), 114-116, 123, 184-185 area tags, 176-177 attributes, 68, 111 b tags, 81-83 big tags, 83 blockquote tags, 73 body tags, 33	iframe tags, 204 img tags, 101, 162-163, 166-167, 365 input tags, 344-345 li tags, 75, 246 link tags, 316 map tags, 177 meta tags, 375-377 name tags, 202 noframes tags, 201 ol tags, 71, 75, 246 opening tags, 32 option tags, 348 p tags, 33, 68-69, 98 pre tags, 83-84 select tags, 348 small tags, 83 strike tags, 83	i tags, 81-83, 398 id attribute, 114-116, 414 iframe tags (HTML), 204 image maps, creating via HTML lists, 251-254 images aligning horizontal alignment, 165-167 vertical alignment, 167-169 alternative text descriptions, 163 animated web images, 156 backgrounds, 144, 154, 172-173 banners, creating, 151-152 buttons, creating, 151-152 changing based on user interaction via JavaScript, 333 clip art, 144

438 images

copyrights, 144	indenting	lang attribute, 415
Creative Commons licenses, 144	HTML code, 366	large (complex) websites, organizing, 360-362
cropping, 146-147	text-indent property (CSS), 57	layouts
digital cameras, exporting images to computer, 146	index files, web servers, 20-22	CSS layout style properties
extracting from web pages, 144	inline styles, internal style sheets (CSS), 62-63	display property, 54
Flickr website, 161-163	inline value (CSS display property), 54	height property, 55
.GIF files, 153	input controls (forms)	quick reference, 424-427
animated web images, 156	check boxes, 346-348	width property, 55
tiled backgrounds, 154	pull-down pick lists, 348	fixed layouts, 294
transparent images, 154	radio buttons, 348	fixed/liquid hybrid layouts
Google Images website, 161	scrolling lists, 348	basic structure of, 298-299
hot linking, 163	selection lists, 348	defining columns in, 300-302
imagemaps	text areas, 349	handling column height in,
areas, 174-175	input / tags, 411-412	304-309
creating HTML for, 176-178	input tags (HTML), 344-345	setting minimum width of a
mapping regions within	inset value (CSS border-style	layout, 302
graphics, 174-175	property), 56	liquid layouts, 293-295, 298
uses for, 173-174	internal style sheets (CSS), 48, 62-63	leading (text), 58
.JPEG files	Internet connections, 27	left property (CSS), 235, 426
compression, 150	Internet Explorer, viewing web pages in Print Preview, 320	letter-spacing property (CSS), 418 li tags, 75, 246, 400
tiled backgrounds, 154	ISPs (Internet service providers),	lightness (images), adjusting,
links	comparing, 27	149-150
turning images into, 169-172 video, 185	italic text, 82-85	line breaks, 34-35, 84
permissions, 163	item indicators, placement in HTML	line-height property (CSS), 58, 416
.PNG files, 153-154	lists, 249-250	links
resizing, 148		a tags (anchors), 184-185
resolution, 145		identifying web page locations,
scaling, 148	J - K - L	114
scanners, getting images from,	JavaScript scripting language	linking to web page locations, 114-116
146	changing images based on user	psuedoclasses, 123
sizing, 165	interaction, 333	absolute addresses, 112
thumbnail images, turning into	DOM, 332	CSS CSS
links, 169-172	event attributes, 283	CSS to HTML documents, 52
tiled backgrounds, 154-155	HTML and, 326-327	styling in, 123-126
transparent images, 154	random content web pages, 328-332	email addresses, 120-121
web pages, adding images to, 162-163	JPEG files	frames to windows, 202-203
wrapping text around, 166	compression (images), 150	graphics as links, 123
img tags, 101, 162-163, 166-167,	tiled backgrounds, 154	hot linking images, 163
365	JSP (Java Server Pages), 326	images, turning into links,
img / tags, 123, 407-408		169-172
@import command, 316		images to video, 185
inclusions. See embedded content	label tags, 412	link tags, 316, 401

multimedia, 184	ordered lists, 71-73, 245	mouse actions
images to video, 185	placing item indicators,	event handling, 283-284
playing linked content, 187	249-250	rollover text, displaying via CSS,
psuedoclasses, displaying, 125	unordered lists, 71-73, 245	281-282
relative addresses, 112	locally publishing web content, 23	tooltips, creating via CSS, 277-280
testing, 119 troubleshooting, 128		multi-level vertical navigation (CSS), 265-269
web browsers, opening links in	M	multimedia
new windows, 122		audiovidual file formats, choosing,
web content, linking	main (home) pages (websites), index files, 22	194
between content, 117-119	maintainable HTML code, writing	automatically playing, 191
to external content, 120	documenting code via comments,	creating, 183-184
web pages, 372	364-365	embedded files, 187-190
creating for, 113	indenting code, 366	file sizes, 191
email addresses, 120-121	map tags, 177-178, 408	file storage, 192
linking between content,	margins	finding files, 183-184
117-119	CSS	linking, 184
linking to external content, 120	CSS box model, 232	images to video, 185
opening links in new windows,	margin property, 212-214, 423	playing linked content, 187
122	margin style properties, quick	MIME types, 189-190
XHTML, quick reference, 400-401	reference, 423	players, choosing, 191
liquid layouts, 295	margin-bottom property, 212,	QuickTime, downloading, 187
fixed/liquid hybrid layouts	423	streaming files, 187, 194
basic structure of, 298-299	margin-left property, 212, 423	text and, 191
defining columns in, 300-302	margin-right property, 212, 423	video hosting services, 191
handling column height in,	margin-top property, 212, 423	web resources, 183-184, 388
304-309	properties of, 212	multiple attribute (select tags), 349
setting minimum width of a layout, 302	styling, 217-219	
web resources, 293	writing in shorthand, 213-214	N
list-item value (CSS display property), 54	HTML lists, 249	N
list-style property (CSS), 427	markers (bullets), CSS marker style	name attribute
list-style-image property (CSS), 427	properties quick reference, 427-428 max-height property (CSS), 416	a tags, 114
list-style-position property (CSS), 428	max-width property (CSS), 416	frame tags, 202
list-style-type property (CSS), 428	maxlength attribute (input tags), 345	param tags, 188
list-style-type style (ul tags), 75	media-specific style sheets, 315-317	name tags (HTML), 202
lists	meta tags, 375-377, 395	navigation (CSS)
CSS list style properties, quick	method attribute (form tags), 340	horizontal navigation, 270-273
reference, 427-428	middle value (vertical-align property),	navigation lists, 259
HTML, 71	224	primary navigation, 260
creating images via, 251-254	MIME (Multi-purpose Internet Mail	secondary navigation, 260
CSS box model and, 246-249	Extensions) types, 189-190	vertical navigation, 260-262
definition lists, 246	min-height property (CSS), 416	multi-level navigation, 265-269
formatting, 246-249	min-width property (CSS), 416	single-level navigation,
nesting, 73-77	monospaced text, 83-84	263-264

text-align: right style, 69

Flickr website, 161-163 nesting, HTML lists, 73-77 padding. See also spacing text CSS. 219-222 .GIF files, 153 noframes tags (HTML), 201 none value (CSS) CSS box model, 232 animated web pictures, 156 border-style property, 56 padding property, 58, 220, 424 tiled backgrounds, 154 display property, 54 padding style properties, quick transparent pictures, 154 reference, 424 noscript tags, 414 Google Images website, 161 padding-bottom property, Notepad, HTML files hot linking, 163 220, 424 creating, 28 imagemaps padding-left property, 220, 424 saving, 29 areas, 174-175 padding-right property. creating HTML for, 176-178 220, 424 mapping regions within padding-top property, 220, 424 graphics, 174-175 0 HTML lists, 249 uses for, 173-174 object tags, 187-190, 408-409 paragraphs .JPEG files ol tags, 71, 75, 246, 400 aligning, 69 compression, 150 onclick event, 283-289, 415 breaks, HTML web pages, 34-35 tiled backgrounds, 154 ondblclick event. 283, 415 param / tags, 409 links, turning pictures into, onkeydown event, 283, 415 param tags (object tags), 188 169-172 onkeypress event, 415 Perl scripting language, 326 permissions, 163 onkeyup event, 283, 415 permissions, images, 163 .PNG files, 153-154 onload event attribute (JavaScript), 283 photographs, 146 resizing, 148 onmousedown event, 283, 415 Photoshop (Adobe), 143 resolution, 145 onmousemove event, 284, 415 PHP (PHP Hypertext Preprocessor), scaling, 148 onmouseout event, 284, 415 326, 340 scanners, getting pictures from, Picasa, 144 onmouseover event, 284, 415 146 onmouseup event, 283, 415 Picnik, 144 sizing, 165 pictures opening tags (HTML), 32 thumbnail pictures, turning into links, 169-172 option tags, 348, 412 aligning ordered lists (HTML), 71-73, 245 horizontal alignment, 165-167 tiled backgrounds, 154-155 vertical alignment, 167-169 transparent pictures, 154 outlines, building via HTML lists, 73-74 web pages, adding pictures to, alternative text descriptions, 163 outset value (CSS border-style 162-163 animated web pictures, 156 property), 56 wraping text around, 166 backgrounds, overflow property (CSS), 242, 426 144, 154-155, 172-173 plug-ins, 185 overflow sheets, float property, 242 banners, creating, 151-152 .PNG (Portable Network Graphics) files, 153-154 overlapping elements, CSS, 235 buttons, creating, 151-152 points (text), 52 clip art, 144 pop-up windows, opening links in, 122 color reduction, 153 position property (CSS), 235, 426 color-correcting, 149-150 positioning copyrights, 144 p tags, 33, 98, 397 absolute positioning, 54, 235, Creative Commons licenses, 144 style attribute, 68 238-239 cropping, 146-147 text-align: center style, 69 relative positioning, 54, 235-237 digital cameras, exporting images text-align: left style, 69 pre tags, 83-84, 397 to computer, 146

extracting from web pages, 144

primary navigation (CSS), 260

strike tags 441

Print Preview (Internet Explorer),	S	213-214
viewing web pages in, 320		simple websites, organizing, 357-359
print-friendly web pages, 311	saturation (images), adjusting, 149-150	single-level vertical navigation (CSS),
designing, 312-315		263-264
media-specific style sheets,	saving files, 29	single-page websites, 356
315-317	scaling images, 148	size attribute
style sheet design, 317-320	scanners getting images from, 146	font tags, 85
viewing, 320	scripting	input tags, 345
pseudoclasses	client-side scripting, 326	select tags, 348-349
a tags (anchors), 123	form-processing scripts, 340, 345	sizing
link pseudoclasses, displaying, 125	script tags, 414	HTML tables, 99-101
publicizing web pages, 371-372	server-side scripting, 326	images, 165
pull-down pick lists, HTML forms, 348	XHTML, quick reference, 414	skeleton web pages, 33
Python scripting language, 326	scrolling lists, HTML forms, 348	small tags, 83, 398
	search engines, 372	software
0 B	AllTheWeb, 374	graphics
Q - R	AltaVista, 374	Adobe Photoshop, 143
QuickTime, 187, 194	Bing, 374	Corel DRAW, 144
	Google, 374	GIMP, 144-152
	listing web pages on, 373-374	Picasa, 144
radio buttons, HTML forms, 348	optimizing website searchability, 374-380	Picnik, 144
random content, displaying in		multimedia, creating, 184
websites via JavaScript, 328-332	providing hints for, 374-379 spam, 375	web resources, 386-387
registered trademark symbol (®), 90	Yahoo! Search, 374	solid value (CSS border-style
registering web pages with search		property), 56
engines, 373-374	secondary navigation (CSS), 260 select tags, 348, 413	sound files, streaming, 194
relative addresses, 112-113	<u> </u>	source editors, blogs, 23
relative positioning, 54, 235-237	selected attribute (select tags), 349	spacing text, 84. See also padding
relative value (position property), 235	selection lists, HTML forms, 348	spam, search engines, 375
resizing images, 148	selectors, CSS, 59	span tags, 395
resolution (images), 145	SEO (search engine optimization), 374-380	spanning within HTML tables,
ridge value (CSS border-style	servers	102-104
property), 56	server-side scripting, 326	special characters, 89-91
right property (CSS), 235, 426	web servers	special formatting, 82-85
robots, 372, 376	document roots, 17-19	spiders, 372, 376
rollover text, displaying via CSS,	file transfers, 17	src attribute
281-282	index files, 20-22	frame tags, 202
rows (tables)	organizing directories, 18-20	iframe tags, 204
adding color to, 104	publishing web content	img tags, 162-163
spanning, 104	without, 22	stacking elements, CSS, 239-241
rows attribute	selecting, 6-7	Starbucks.com, website organization,
frameset tags, 200	server-side scripting, 326	361-362
textarea tags, 349	shape attribute (area tags), 176	storing multimedia files, 192
rowspan attribute (th tags), 104	shorthand, writing margins in (CSS),	streaming files, 187, 194
Ruby scripting language, 326	5	strike tags, 83

442 strong tags

strong tags, 82-83, 397	text-decoration property, 58	style ID, 61
style attribute, 414	text-indent property, 57	style rules, 59
div tags, 165	inline styles, internal style sheets,	text flow, managing, 242-243
html tags, 96	62-63	tooltips, creating, 277-280
p tags, 68	internal style sheets, 48, 62-63	validating, 63
table tags, 104	layout properties	z-index property, 235, 239-241
style classes (CSS), 58-61	display property, 54	style tags, 62, 410
style ID (CSS), 61	height property, 55	sub tags, 83, 397
style properties (CSS)	width property, 55	sup tags, 83, 397
defining, 48	links	
formatting properties, 55-58	displaying pseudoclasses, 125	
layout properties, 54-55	styling, 123-126	T
style rules (CSS), 59	margins	tabindex attribute, 414
defining, 48	box model, 232	tablices attribute, 414
viewing in web browsers, 53	margin property, 212-214	CSS table style properties, quick
style sheets (CSS)	margin-bottom property, 212	reference, 428-429
alignment, 223-224	margin-left property, 212	HTML tables
borders, 232	margin-right property, 212	adding color to, 104
box model, 231-234	margin-top property, 212	aligning within, 102-104
clear property, 228, 242	properties of, 212	columns, 104
creating, 53	styling, 217-219	creating, 96-99
current line, 242	writing in shorthand, 213-214	defining cells, 96
event handling, 283-284	navigation	rows, 104
external style sheets, 48-51	horizontal navigation, 270-273	sizing, 99-101
float property, 224-227, 242	navigation lists, 259	spanning, 102-104
formatting properties	primary navigation, 260	web page layouts, 105
background-color property, 56	secondary navigation, 260	table of contents (TOC), websites,
border property, 55	vertical navigation, 260-269	356-357
border-bottom property, 55-56	overlapping elements, 235	table tags, 96, 99, 365, 403
border-color property, 55-56	padding, 219-222, 232	table-layout property (CSS), 429
border-left property, 55-56	positioning	XHTML tables, quick reference,
border-right property, 55-56	absolute positioning, 235,	402-406
border-style property, 55-56	238-239	target attribute
border-top property, 55-56	bottom property, 235	frame tags, 202
border-width property, 55-56	left property, 235	a tags, 123
color property, 56, 58	position property, 235	tbody tags, 404
font property, 58	relative positioning, 235-237	td tags, 96-102, 404-405
font-family property, 57	right property, 235	testing links, 119
font-size property, 57	top property, 235	text. See also HTML lists
font-style property, 57	print-friendly web pages	aligning, 68
font-weight property, 57	media-specific style sheets,	page sections, 69-71
line-height property, 58	315-317	paragraphs, 69
padding property, 58	style sheet design, 317-320	alternative text, images, 163
text-align property, 57-58	rollover text, displaying, 281-282	color, 56, 134-138
3 1 1 3/1 3	stacking elements, 239-241	

stacking elements, 239-241 style classes, 58-61

top value (vertical-align property),

224

CSS non-European characters, 93 spacing, 84 creating tooltips, 277-280 font property, 57-58, 417 special characters, 89-91 font style properties, quick refspecial formatting, 82-85 erence, 416-418 HTML font-family property, 57, 417 creating files via TextEdit, 29 font-size property, 57, 417 forms, 344-345, 349 font-style property, 57, 417 tags, 29-30 font-variant property, 417 leading, 58 font-weight property, 57, 417 multimedia and, 191 managing flow in, 242-243 points, 52 text style properties, quick rollover text, displaying via CSS, reference, 416-418 281-282 text-align property, 57-58, 418 text-align property, 57-58, 223, 418 text-decoration property, 58, 418 text-align: center style (p tags), 69 text-indent property, 57, 418 text-align: left style text-transform property, 418 div tags, 165-167 editors, 28 p tags, 69 fonts (text) text-align: right style CSS font style properties, div tags, 165-167 quick reference, 416-418 p tags, 69 font property (CSS), 57-58, text-bottom value (vertical-align 417 property), 224 font tags, 81, 85 text-top value (vertical-align font-family property (CSS), 57, property), 224 417 textarea tags, 349, 413 font-family style (font tags), 85 TextEdit, creating HTML files, 29 font-size property (CSS), 57, tooltips, creating via CSS, 277-417 280 font-size style (font tags), 85 wrapping around images, 166 font-style property (CSS), 57, tfoot tags, 405 417 th tags, 96, 101-102, 405 font-variant property (CSS), thead tags, 406 417 thumbnail images as image links, font-weight property (CSS), 57, 169-172 417 tiled backgrounds, 154-155 size, 85-88 titles formatting headings versus titles, 38 boldface text, 82-85 title attribute, 414, 162 color, 85-88 title tags, 33, 395 CSS. 88 TOC (table of contents), websites, font size, 85-88 356-357 HTML web pages, 36 tooltips, creating via CSS, 277-280 italic text, 82-85 top property (CSS), 235, 426 line breaks, 84

monospaced text, 83-84

tr tags, 96-97, 102, 406
trademark symbol (™), 90
transferring files. **See** FTP
(File Transfer Protocol)
transparent images, 154
triadic color schemes, 132
troubleshooting links, 128
tt tags, 83-84, 399
type attribute
input tags, 345
object tags, 188

U - V

u tags, 83 ul tags, 71, 75, 246-248, 400 units of measurements, CSS height/width properties, 55 unordered lists (HTML), 71-73, 245 usemap attribut (img tags), 177

validating web pages, 39 value attribute

input tags, 349 param tags, 188

VBScript (Visual Basic Scripting Edition), 326

vertical navigation (CSS), 260-262 multi-level navigation, 265-269 single-level navigation, 263-264 vertical-align attribute (tables), 102 vertical-align property, 223, 427 vertically aligning images, 167-169 video

embedding, 190 hosting services, 191 streaming, 187, 194 visibility property (CSS), 427 visual editors, blogs, 23

N	testing, 24	print-friendly, 311
V3C CSS Validator, 63	validating, 39	designing, 312-315
V3C Markup Validation Service,	web hosting providers, selecting, 6-7, 28	media-specific style sheets,
39-41	web pages	315-317
vallpaper. See backgrounds	backgrounds, extracting from web	style sheet design, 317-320
veb addresses, 111	pages, 144	viewing, 320
absolute addresses, 112-113	bare-bones web pages, 33	publicizing, 371-372
document root, 112	bookmarks, 33	search engines
relative addresses, 112-113	creating, text tags, 29-30	listing web pages on, 373-374
veb browsers	designing, web resources, 386	optimizing website searchability, 374-380
client-side scripting, 326	fixed layouts, 294	providing hints for, 374-379
comparing, 28	fixed/liquid hybrid layouts	spam, 375
CSS, viewing style rules, 53	basic structure of, 298-299	validating, 39
distributing, 23	defining columns in, 300-302	
helper applications, 185	handling column height in,	viewing, 31
HTML lists, formatting, 72	304-309	XHTML web pages, HTML tags, 32-33
links, opening in new windows, 122	setting minimum width of a layout, 302	web resources
plug-ins, 185	hiding sections via comments, 365	advanced developer resources,
selecting, 8, 28	hosting, finding providers, 28	388
testing, 8	HTML web pages	color, 387
web content	formatting text, 36	CSS, 385
delivering, 4	HTML tables, 105	graphics, 387
retrieving, 5	line breaks, 34-35	HTML, 385
web pages, viewing, 31	organizing content via heading	multimedia, 388
web resources, 386	tags, 36-38	software, 386-387
veb content	paragraph breaks, 34-35	web browsers, 386
creating, 2-3	viewing tags, 39	web pages, 386
delivering, process of, 4	images	websites, hosting, 389
linking	adding to, 162-163	XHTML, 385
between content, 117-119	extracting from web pages, 144	web servers
external content, 120	internal style sheets (CSS), 62	directories, organizing, 18-20
organizing via HTML heading tags,	links, 113, 372	document roots, 17-19
36-38	email addresses, 120-121	file transfers, 17
publishing	linking between content,	index files, 20-22
blogs, 23	117-119	publishing web content without, 22
locally, 23	linking to external content, 120	selecting, 6-7
operating a FTP client, 16-18	opening in new windows, 122	server-side scripting, 326
organizing web server	troubleshooting, 128	web-based forms (HTML)
directories, 18-20	liquid layouts, 293-295, 298	accepting text input, 344-345
selecting a FTP client, 14-15	page locations	creating, 340-344
web server index files, 20-22	identifying via a tags	defining, 339
without web servers, 22	(anchors), 114	form-processing scripts, 340, 345
retrieving, 5	linking to via a tags (anchors),	hidden data, 345
searching, 372-373	114-116	input controls

check boxes, 346-348		
pull-down pick lists, 348		
radio buttons, 348		
scrolling lists, 348		
selection lists, 348		
text areas. 349		
naming data, 345		
submitting data, 340, 349		
websites		
color		
background color, 133-134		
formatting, 131-136		
hexadecimal values, 135-136		
selecting, 131-132		
text color, 134		
dynamic websites		
changing images based on		
user interaction via		
JavaScript, 333		
client-side scripting, 326		
defining, 325		
displaying random content via		
JavaScript, 328-332		
DOM, 332		
JavaScript in HTML, 326-327		
server-side scripting, 326		
formatting, color, 131-136		
home (main) pages, index files, 22		
hosting, web resources, 389		
organizing		
Amazon.com, 361		
ESPN.com, 359		
large (complex) websites, 360-362		
simple websites, 357-359		
Starbucks.com, 361-362		
TOC, 356-357		
search engines, 372		
AllTheWeb, 374		
AltaVista, 374		
Bing, 374		
Google, 374		
listing web pages on, 373-374		
optimizing website		
searchability, 374-380		

providing hints for, 374-379 spam. 375 Yahoo! Search, 374 services, web resources, 389 single-page websites, 356 web browsers, testing websites on, 8 white-space property (CSS), 418 width fixed/liquid hybrid layouts, setting minimum width in, 302 images, specifying in, 165 width attribute map tags, 177 table tags, 99 width property (CSS), 55, 416 windows, linking with frames, 202-203 Windows Media Player, streaming multimedia files, 187 word-spacing property (CSS), 418 WordPress Theme Gallery, liquid layouts, 293 wrapping text around images, 166 WWW (World Wide Web), development of, 2

tags
iframe tags, 204
object tags, 187-190
web pages, HTML tags, 32-33
web resources, 385
XML (Extensible Markup Language),
applications of, 41-42

Y - Z

Yahoo! Search, 374
YouTube, 192

z-index property (CSS),

235, 239-241, 427



XHTML (Extensible Hypertext Markup Language), 42

character entities, 90
quick reference
common attributes, 414-415
embedded content, 407-409
forms, 410-413
links, 400-401
lists, 399
paragraphs, 395-397
scripts, 414
structure elements, 392-395
style sheets, 410
tables, 402-406
text formatting, 398-399
text phrases, 395-397