



# CSS *for* Print Designers

JD Graffam

A large, stylized, teal-colored graphic of the letters 'CSS' serves as the background for the entire page. The letters are thick and have a slightly irregular, hand-drawn feel.

# CSS *for* Print Designers

JD Graffam

## CSS for Print Designers

JD Graffam

New Riders  
1249 Eighth Street  
Berkeley, CA 94710  
510/524-2178  
510/524-2221 (fax)

Find us on the Web at: [www.newriders.com](http://www.newriders.com)

To report errors, please send a note to [errata@peachpit.com](mailto:errata@peachpit.com).

New Riders is an imprint of Peachpit, a division of Pearson Education.

Copyright © 2011 by JD Graffam

Acquisition Editor: Wendy Sharp  
Project Editor: Becca Freed  
Production Editor: Danielle Foster  
Development Editor: Judy Ziajka  
Copyeditor: Naomi Adler Dancis  
Proofreader: Suzie Nasol  
Compositor: Danielle Foster  
Indexer: Rebecca Plunkett  
Cover design: Aren Howell Straiger  
Interior design: Danielle Foster

### Notice of Rights

All rights reserved. No part of this book may be reproduced or transmitted in any form by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher. For information on getting permission for reprints and excerpts, contact [permissions@peachpit.com](mailto:permissions@peachpit.com).

### Notice of Liability

The information in this book is distributed on an “As Is” basis without warranty. While every precaution has been taken in the preparation of the book, neither the author nor Peachpit shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the instructions contained in this book or by the computer software and hardware products described in it.

### Trademarks

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and Peachpit was aware of a trademark claim, the designations appear as requested by the owner of the trademark. All other product names and services identified throughout this book are used in editorial fashion only and for the benefit of such companies with no intention of infringement of the trademark. No such use, or the use of any trade name, is intended to convey endorsement or other affiliation with this book.

ISBN-13 978-0-321-76588-8

ISBN-10 0-321-76588-5

9 8 7 6 5 4 3 2 1

Printed and bound in the United States of America

## Dedication

I'm renting a cabin near Pickwick Lake, where Tennessee, Mississippi, and Alabama touch. I'm here by myself to finish writing this book—to focus. The cabin is a one-room A-frame with a little back porch that overlooks a deep hollow.

Rather than waking up to the Rock 103 deejays, early morning bird songs get me up. I'm not chasing new business throughout the day; squirrels are chasing each other across the tin roof. And at midnight, instead of police sirens, packs of coyotes yip in the distance.

It's been nice to be alone, focusing on my writing. But today, I need to get out of this cabin and on the water to write. There are two reasons for this.

The first reason is practical: It's March and I need to take half a day to make sure our pontoon boat is ready for the summer season. After a quick wipe-down and a two-minute drive from dry storage to the state park, I put her in the water and turn the key—she fires right up without a problem.

The second reason is to dedicate this book to my family. I cannot think of a more appropriate place to write a dedication to my family than on the water, because the special memories I have with everyone in my family are surrounded by water: Lake D'Arbonne, Lake Claiborne, Lake Travis, and Lake Acworth; the Gulf of Mexico, the Pacific Ocean, and the Atlantic; the Ouachita River, Mississippi River, Red River, Tennessee River, and last, but not least, the Loutre Bottom.

I idle slowly across the choppy water into a cove, drop anchor, and start writing. It's windy. Ten minutes in, the boat is banging up against the shore and tangled in lost fishing line. I adjust, drop anchor again. This time the anchor holds.

I write the dedication: "To family, the single important thing in life."

The wind is picking up—waves are slapping against the rocks on the shore, floating docks are squeaking, and the birds are flying low. A thunderstorm will settle over the lake tonight. But for now, I'm enjoying the water. I'm hundreds of miles away from my family, but I'm connected to them. I fold up my laptop and float.

## Acknowledgments

I want to acknowledge these people for their inspiration and for helping make this book possible:

Denise Jacobs, author of *The CSS Detective's Guide*, for introducing me to Peachpit Press. She is the primary reason this book exists.

AIGA chapters and print designers across the country, for letting me lead my CSS for Print Designers workshops.

My Peachpit friends, for sticking with me as I went through the process of writing my first book. I learned a lot, mostly how patient y'all are.

And finally, the wonderful people I work with every day. To my team at Simple Focus, to our clients who trust us to make their Web sites the right way, and to the Memphis design community.



# CONTENTS

	<b>Introduction</b>	<b>vii</b>
<b>CHAPTER 1</b>	<b>Coding in Plain English</b>	<b>1</b>
	CSS Is Easy to Memorize . . . . .	2
	You Already Know How to Read CSS. . . . .	2
<b>CHAPTER 2</b>	<b>From Picas to Pixels</b>	<b>5</b>
	Three Steps to Thinking Like a Web Designer . . . . .	6
	Setting Up Your Web Design Files . . . . .	11
<b>CHAPTER 3</b>	<b>Dump Drag and Drop</b>	<b>17</b>
	Why You Shouldn't Rely on Software to Set Up Your Paths . . . .	18
	Web Sites Have Folders Like Your Computer. . . . .	18
	How FTP Works (the Oversimplified Version) . . . . .	20
	Navigating Folders on the Internet . . . . .	21
	It's Like Packaging InDesign Projects . . . . .	21
	Absolute Paths . . . . .	23
	Relative Paths . . . . .	24
	Moving Within the Same Folder . . . . .	25
	Moving Into Deeper Folders . . . . .	25
	Moving Into Higher Folders. . . . .	25
	Putting It Together . . . . .	25
	Starting at Home . . . . .	26
	Leaving Off Index . . . . .	26
	Trailing Slashes . . . . .	26
	This Chapter Will Fix 89.6 Percent of Your Problems— Read It Again . . . . .	26
<b>CHAPTER 4</b>	<b>You Have to Read the Words</b>	<b>27</b>
	It's the Whole Purpose of Coding . . . . .	28
	So What Does HTML Do, Exactly? . . . . .	29

<b>CHAPTER 5</b>	<b>Boxes Inside Boxes</b>	<b>33</b>
	Learning CSS Happens Fast, Once You Know the Secret. . . . .	34
	HTML Is Made Up of Tags . . . . .	35
	A Dozen Tags You Need to Know. . . . .	36
	New HTML vs. Old HTML . . . . .	38
	Tag Groups (Also Known as Lists) . . . . .	39
	Nesting Tags . . . . .	44
	Adding Attributes to Tags . . . . .	46
	Five Attributes You Need to Know . . . . .	47
	Self-Closing Tags . . . . .	49
	Formatting Code Is Like Setting Type. . . . .	50
	Block and Inline Tags . . . . .	53
	HTML Looks Like a Word Doc . . . . .	56
<b>CHAPTER 6</b>	<b>Sculpting with CSS</b>	<b>57</b>
	First Things First: Syntax . . . . .	58
	Formatting Your CSS . . . . .	60
	Getting Fancy with Selectors . . . . .	62
	Reading Selectors in Plain English . . . . .	66
	Commenting Your CSS . . . . .	67
	Let's Write Some CSS. . . . .	68
	Setting Type with CSS. . . . .	68
	Laying Out a Web Page. . . . .	74
	Designing for Interaction . . . . .	85
	Let's Make a Web Page Together. . . . .	88
	You Already Know a Lot . . . . .	92
<b>CHAPTER 7</b>	<b>Designing with CSS</b>	<b>93</b>
	Using Background Images . . . . .	94
	Cropping Images with CSS . . . . .	101
	Making Columns with Background Images. . . . .	108
	Designing with CSS3—Without Images. . . . .	112
	Putting It All Together . . . . .	119
	Making Design Happen. . . . .	122

<b>CHAPTER 8</b>	<b>Improving Lives with CSS</b>	<b>123</b>
	Empathy Through CSS . . . . .	124
	Designing for Print (with CSS) . . . . .	124
	Designing for Accessibility . . . . .	127
	Writing CSS That Loads Fast . . . . .	129
	Thinking Beyond Visual Design . . . . .	132
<b>CHAPTER 9</b>	<b>Starting from Scratch</b>	<b>133</b>
	Copying and Pasting Saves Time . . . . .	134
	More HTML Tags . . . . .	134
	The HTML Framework . . . . .	137
<b>CHAPTER 10</b>	<b>Prepress for the Web</b>	<b>139</b>
	Image Production for Web Design . . . . .	140
	Starting by Planning . . . . .	140
	Understanding Image Formats . . . . .	144
	Saving Your Images . . . . .	148
	Measure Twice, Slice Once . . . . .	149
<b>CHAPTER 11</b>	<b>Tools of the Trade</b>	<b>151</b>
	Web Design Doesn't Require Many Tools . . . . .	152
	The Web Designer's Toolbox . . . . .	152
	Putting It All Together . . . . .	155
<b>CHAPTER 12</b>	<b>Any Questions?</b>	<b>157</b>
	Frequently Asked Questions . . . . .	158
	Coding Is an Art . . . . .	162
	There Is No Wrong Way to Learn. . . . .	162
	<b>Index</b>	<b>163</b>

# VIDEO CONTENTS

<b>CHAPTER 2</b>	Things You Can Do with Web Design That You Can't Do With Print
<b>CHAPTER 3</b>	How Paths Work on Your Computer and In a Browser
<b>CHAPTER 5</b>	How to Start Writing HTML from Scratch
<b>CHAPTER 6</b>	How to Lay Out a Web Page with CSS
<b>CHAPTER 7</b>	How to Add Style to a Web Page with CSS
<b>CHAPTER 8</b>	Making a Web Page Accessible and Printable with CSS
<b>CHAPTER 9</b>	Similarities Between InDesign, HTML, and CSS
<b>CHAPTER 10</b>	How to Slice and Dice a Photoshop File
<b>CHAPTER 11</b>	How to Update a Live Web Site

# INTRODUCTION

A while back, I volunteered to lead a CSS workshop in Memphis. I came up with the idea because it seemed like I kept having the same conversation with print designers, especially after a couple rounds of beer at AIGA mixers. The conversations would go something like this:

Me: “This beer is pretty good.”

Print Designer: “Yeah, I’m a little buzzed. Geez, I wish I could code Web sites by hand.”

Me: “I bet you could learn it pretty quickly. It’s easy. I’m buzzed, too.”

Print Designer: “No way. My mind just doesn’t work like yours. I’m more of a visual kind of person.”

Me: “No, seriously—I used to be a print designer. I could show you the basics in a few minutes. Coding is a lot more visual than you might think.”

Print Designer: “Yeah, right.”

Me: “I’m not kidding.”

Print Designer: “You should do a workshop about this, then. I bet it’d do really well. And you should serve beer at it.”

After a while, I started thinking a little more about the idea of teaching CSS to print designers. I was convinced that with three hours I could teach CSS to someone who’d never hand-coded before. It would have to be basic. It’d have to move fast. Print designers are pretty tech-savvy people—they can learn this stuff.

So with the help of AIGA Memphis, I organized a workshop. We called it CSS for Print Designers. It sold out after only a few days of promotion, so we organized a second one, which sold out within 48 hours. A few weeks later, I got an email about doing a workshop in a different city. Then I got another email, and another, and another.

I think the timing of this book is perfect. Five years ago there were print designers who didn’t touch Web projects, and Web designers who didn’t code. Now there’s hardly a single print designer who hasn’t been asked to design something for the Web, and almost every Web designer knows how to code CSS by hand.

The gap between designer and coder is closing, and fast. There's an entire generation of pixel-native designers coming up. To them, CSS is just another design tool. When you're done reading this book, I want you to feel the same way.

## Who This Book Is For

This book covers the basics of how to code Web sites by hand with a plain text editor. It's deliberately short and written in a casual voice, without jargon or geek-speak.

Who should read it?

- A designer or visual thinker who's ready to take the first step toward finally learning how to code by hand
- A designer who has tried to read other books or online tutorials about coding Web sites, but came away overwhelmed and frustrated
- A designer looking for a way to communicate better with coders when handing off projects to them
- An art or design student who wants to make Web sites but isn't interested in a computer science degree
- A design professor who needs to teach students the latest Web standards and technology

## What This Book Is Not

This book is not a comprehensive resource. For example, there are over 100 HTML elements, but this book covers only the dozen or so that are used the most and instructs the reader to search for the rest. It doesn't cover everything there is to know about Web design—that's what the Internet is for.

This book is not philosophical. While this book is perfect for students and professors, it does not cover the *why* of code. Instead, it focuses on the *how*. It's just practical like that.

This book is not a workbook. While there are practical examples scattered throughout, this book doesn't start with Chapter 1 and walk you through step-by-step how to build a Web site. Instead, it teaches concepts in a logical order with contextual examples that help explain those concepts.

This book is not for computer novices. Print designers work with specialized software and know all about the Internet, so the readers of this book need to be comfortable working with applications like Photoshop, InDesign, or Illustrator and making their way around online.

# 4

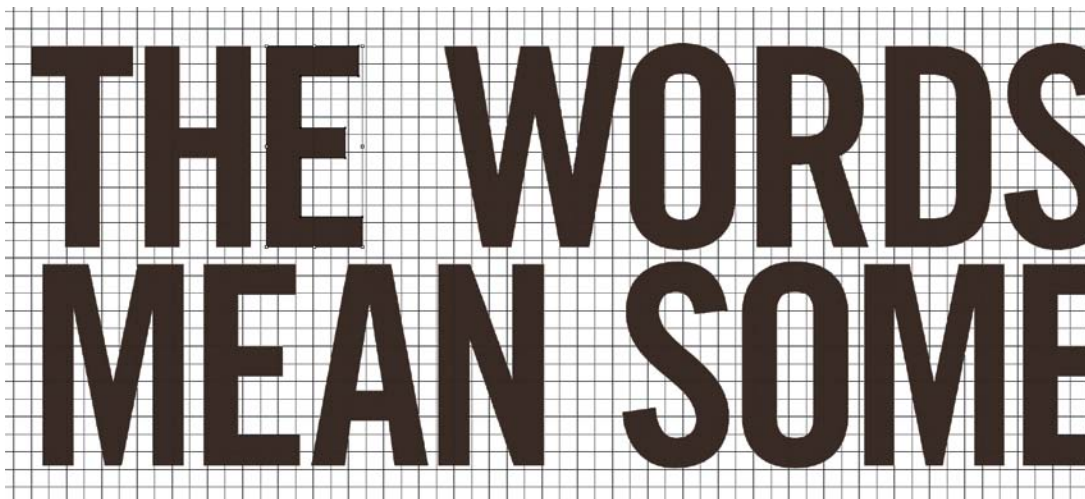
## You Have to Read the Words

*This Is Not About Proofreading*

**THERE'S A REASON** we designers are not allowed to send anything to the printer without having it proofread by a qualified individual. For designers, words and letters aren't much more than shapes interacting with one another. When we're in the zone, kerning our little hearts out, our minds aren't focused on meaning (**FIGURE 4.1**).

I'm not saying designers don't read—there's no way to art-direct a magazine layout or annual report without understanding the information being communicated. But let's be honest: Sometimes we skim. I mean, sure, headlines can be fun, but long copy is *boring*.

When we're setting type, we're thinking more about the aesthetic and readability of the words than we are about their grammatical significance.



**FIGURE 4.1** Kerning is fun. Reading is not.

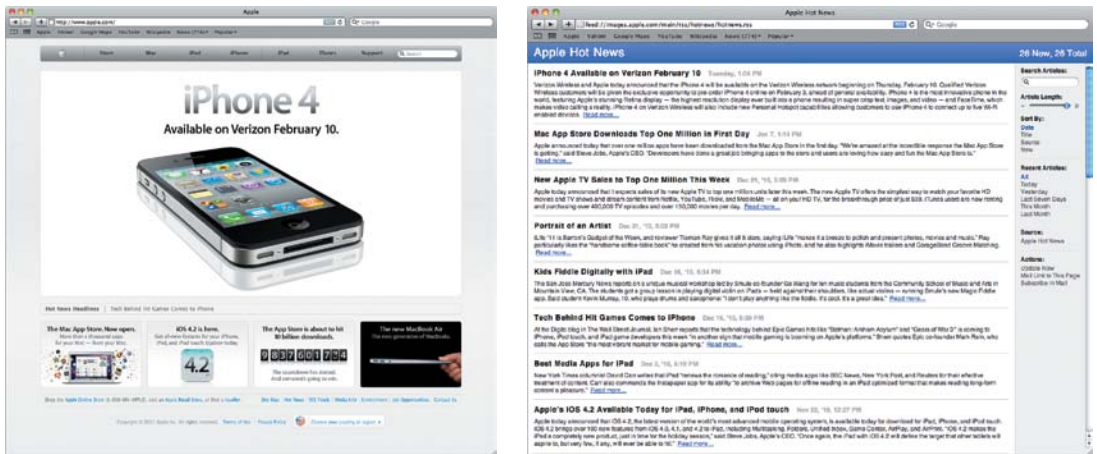
## It's the Whole Purpose of Coding

Before we start telling our Web pages what to look like with CSS, we code Web sites with HTML so the content we're designing can be shared in different ways. Additionally, CSS (the language that controls look and feel), is built on the structure of HTML—until you have a way of saying, “This is body copy,” you can't write CSS to tell the body copy what to look like. HTML is that framework.

Let me explain this by demonstrating a problem you have probably faced from the print design world. If you need to lay out two brochures at different dimensions, you create a separate InDesign file and bring the content into each file separately. From then on, when your proofreader discovers a typo, you have to fix it in two places.

There's a concept in the Web design world where we separate content from the way it looks. You might hear of this as the separation of presentation from content. In short, it means that we maintain the content in one place and just reskin it with CSS to reuse it in different forms (**FIGURE 4.2**).






**FIGURE 4.2** The content on Apple's homepage looks a certain way when you browse the Web site, but it looks different if you view it in an RSS reader.

## So What Does HTML Do, Exactly?

HTML just tells us what sort of content we're looking at. To write HTML, you need to read the content from your copywriter so that you can identify what each chunk of content is. It's actually pretty simple, so don't freak out; here's a list of the main types of content you'll need to identify:

- **Paragraph:** A chunk of text; ideally it's a series of related sentences that create a unified thought.
  - HTML also lets you put breaks between chunks of content to mimic a soft return. Don't be fooled: Adding space below a chunk of text with a soft break is not the same as identifying it as a paragraph. HTML is not about what something *looks like*; it's about what it is.
- **Headline:** Summarizes the chunk of content that follows it.
  - Main headlines summarize the overall meaning of the following content, while the lesser headlines progressively summarize smaller, less global ideas.
  - HTML gives you six levels of headlines to pick from.
  - Typically, the more important the headline, the bigger it is. This is not always the case, though, which is why HTML exists. HTML helps you establish what the content *is* before you tell it what to *look like*. For example, you should work hard to ensure that `<h3>` tags are not above `<h2>` tags.

 **TIP** When you cheat with HTML to get the look you want, you handcuff yourself because CSS isn't able to control the look and feel any more. One of the biggest advantages of using CSS to control look and feel is that look and feel is controlled in a single place.

- **List:** A series of related items.
  - HTML lets you work with a few different types of lists:
    - Unordered lists are what we typically call bulleted lists. But don't be fooled; not all unordered lists have bullet points. They might have arrows, or dashes, or squares. They might not even have these graphics at all. For example, navigation, which is a list of page links, does not. In case I haven't convinced you already, HTML isn't about what something *looks like*. It's about what it *is*.
    - Ordered lists are what we might call numbered lists. They can be 1., 2., 3... or i., ii., iii., etc. Use them when you're ordering items deliberately.
    - Definition lists have an item that's followed by some sort of definition or clarification. For example, a glossary of terms would use a definition list.
- **Other stuff:** There are other little HTML chunks that you'll need to learn about that will let you emphasize, link, or group things together. But for now, paragraphs, headlines, and lists are the main ones I want you thinking about.

Using just the tags described above, your content can be displayed in an RSS feed or mobile-optimized Web site that has its own look and feel. It won't use the same typography or colors as your Web site, but it will be readable because it has HTML telling it which chunks of text are headings, paragraphs, and lists.

When your copywriter gives you a Word document, the document typically reads from top to bottom. It isn't pretty, but it's logical. Next time you get content from your copywriter, refer back to this section and try to identify each chunk of text the way HTML asks us to identify it. Just mark in the margins what each chunk of text is (**FIGURE 4.3**).



**FIGURE 4.3** When you start writing HTML, it might help to put pen to paper and mark up the chunks of content.

If you ever get stuck trying to figure out how to identify a chunk of content, it wouldn't be a bad idea to check with your copywriter. Copywriters often use styles to format their documents, which you know can be mapped to InDesign. Similarly, this sort of structure could make your HTML writing easier.

Just like there's no *flow content* button that works for a 60-page technical manual you're designing with InDesign (boy, wouldn't that be nice!), there's no magic button that does this for coding Web sites—you have to use your brain.

*This page intentionally left blank*

# INDEX

## Symbols

\ (backslash), 23, 25, 26  
: (colon), 58, 59  
{ } (curly brackets), 58, 59  
" " (curly quotes), 47, 161  
# (hash), 63  
- (hyphens), 69  
. (period), 63  
; (semicolon), 58, 59, 60  
tick marks ("), 47, 70, 96

## A

absolute paths, 23  
accessibility, 127–129  
active command, 86  
attributes  
  adding class and id, 63  
  HTML's accessibility, 127  
  <media>, 136  
  using, 46–49

## B

background images, 94–101  
  cropping, 100  
  defining background color for, 94–95  
  <img> tag vs. background-image property, 94  
  incorporating in design, 119–122  
  making columns with, 108–112  
  positioning, 98–100  
  repeating, 96–97, 109–112  
  shorthand for CSS coding, 100–101  
  specifying paths for, 95–96  
block tags, 53–56, 78  
<body> tag, 136–137  
borders, 78–81  
boxes  
  adding drop-shadows to, 116–118  
  nesting tags as boxes within, 44–45  
  visualizing image production with, 141–142  
browsers  
  interacting with CSS3, 113–115  
  PNG transparency and Internet Explorer 6, 148  
  resetting default CSS for, 84–85  
  selecting development, 154  
  supporting Internet Explorer 6, 160–161  
  using newer HTML with older, 38  
  viewing HTML files in, 56  
bulleted lists, 30, 37, 39–40

## C

Cascading Style Sheets. *See* CSS  
class attribute, 63  
closing tags (<forewardslash >), 35, 36  
coding  
  background images, 100–101  
  comments for organizing, 52–53  
  consistency in, 50  
  HTML before adding CSS, 28–29  
  margins and padding, 83–84  
  marking up page in HTML, 8, 142–143  
  reading CSS, 2  
  RGB color, 72  
  sample CSS web page, 90–91, 119–122  
  tables, 150  
color  
  cautions for, 15  
  choosing text, 71–72  
  defining background for background images, 94–95  
  GIF images with matte, 146–147  
  identifying in CSS, 60, 72  
  optimizing for GIF files, 145  
  RGB, 15–16, 72  
  selecting for print style sheets, 126  
  using with interactive commands, 87–88  
columns, 75–79, 108–112  
comments, 52–53, 67  
content  
  copying and pasting in HTML, 134  
  designing for “fold” in site design, 160  
  identifying for layout, 142  
  setting minimum height for, 109  
  using tables for laying out, 159  
  viewing on page vs. RSS reader, 29  
cropping images, 101–108  
CSS (Cascading Style Sheets). *See also* background images;  
  selectors  
  accessibility features with, 127–129  
  adding shadows in CSS3, 116–119  
  adjusting line height, 72  
  calling with HTML tags, 136  
  changing case of HTML text with, 73  
  coding in HTML before adding, 8, 28–29, 34–35  
  commenting in, 67  
  controlling page printing, 124–126  
  creating interactive features, 85–88  
  cropping images with, 101–108  
  .css file extension, 136, 153  
  designing page layout, 74–85, 124  
  downloading speeds for, 129–132

CSS (Cascading Style Sheets) *continued*

- formatting, 60–61
- hooking into with class and id attributes, 63
- how to read, 2–3
- identifying colors in, 60, 72
- image replacement with, 106–107
- kerning with, 73
- linking home page to CSS styles, 21
- lowercase specificity in, 69
- navigating without a mouse, 128–129
- overriding declarations in, 104, 131–132
- properties demonstrated for, 92
- pros and cons of CSS3, 112–115
- resetting browser defaults, 84–85
- rounded corners in CSS3, 115–116
- sculpting HTML with, 57–58
- setting content height, 109
- shorthand for margins and padding, 83–84
- specifying fonts with, 69–72, 125–126
- sprites in, 101, 102, 105, 119–120
- syntax elements for, 58–59
- typesetting with, 2, 68–74
- units of measurement with, 70
- white space and file size, 62
- writing Web page in HTML and, 88–92

CSS3 generators, 116

**D**

definition lists, 30, 41–43

## designers

- accepting design variation, 6–8, 9–11, 14
- adapting to CSS formatting, 60
- adjusting to new workflow, 13–14
- color design in RGB, 15–16, 72
- designing for Web vs. paper, 5–6
- developing interactive features, 85–86
- faking vertical alignment on Web, 108–112
- frequently asked questions from, 158–162
- learning art of Web design, 162
- points to consider in CSS, 124
- producing images for Web, 139–140
- pros and cons of CSS3 for, 112–115
- selling clients on CSS3, 113–114
- understanding screen resolution, 14–15
- using HTML, 8–9
- writing text in HTML, 27–31

division tag (<div>), 37, 160

## downloading

- defined, 20
- increasing CSS speeds for, 129–132
- videoclips for book, 26

**E**

## ems

- <em> tag, 36
- measuring, 70
- using em dashes, 161

**F**

favicon, 136

files. *See also* paths

- .css extension for, 136, 153
- favicon, 136
- GIF, 144–147
- .html extension for, 56, 135, 153
- HTML Framework, The, 137
- .ico extension for, 136
- JPEG, 146, 147
- packaging Web, 21–23
- paths to, 18–20
- Photoshop settings for design, 16
- PNG, 147–148
- uploading and downloading, 20
- viewing HTML, 56
- white space and size of, 62

Fireworks, 12, 152

## floats

- clearing, 77
- making three-column layout with, 78
- planning production by visualizing, 141–142
- setting up columns with, 75–77
- using float property, 74–75

## folders

- absolute and relative paths to, 23–24
- adding trailing slash on path to, 26
- jumping to root, 26
- moving in and between path, 25
- navigating Internet, 21
- organizing files into, 23
- saving production images in separate, 149
- setting up paths to, 18–20

## fonts

- color of, 71–72
- selecting for print style sheets, 125–126
- size of, 70
- specifying with CSS, 69–70
- using temporary, 69
- weight and style of, 71

<footer> tags, 36

## formatting

- code in HTML, 50–51
- CSS, 60–61

FTP (File Transfer Protocol), 20–21, 153

**G**

GIF file format, 144–147

glossary listings, 41–43

**H**

## headlines

- about, 29, 37
- <h1> tag for, 36, 158–159
- <header> tag for, 36
- hexadecimal codes, 60, 72, 117

home folder, 26  
 horizontal positioning keywords, 98  
 hover command, 86  
 HTML Framework, The, 137  
 HTML (HyperText Markup Language). *See also* tags  
   about, 8–9  
   accessibility attributes in, 127  
   avoiding HTML4 DOCTYPEs, 159  
   calling CSS with tags, 136  
   case in code for, 73, 74  
   coding pages in, 28–29, 88–89  
   copying and pasting, 134  
   formatting code in, 50–51  
   .html file extension, 135, 153  
   linking with anchors in, 37  
   lists in, 30, 37, 39–43  
   marking up page in, 8, 142–143  
   navigating to index file, 26  
   organizing code with comments, 52–53  
   saving and viewing, 56  
   sculpting with CSS, 57–58  
   special characters in, 161–162  
   styling with CSS, 29  
   tables in, 150  
   tags in, 29–31, 34, 35–37, 136  
   targeting nested selectors in, 64–66  
   timesaving with HTML Framework files, 137  
   understanding before writing CSS, 34–35  
   using version HTML5, 38  
   writing semantic, 48–49

## I

.ico file extension, 136  
 id attribute, 63  
 images. *See also* background images; Web image  
   production  
     about <img> tag, 36, 94  
     creating simple Web page using, 119–122  
     cropping with CSS, 101–108  
     designing rollovers with, 102–104  
     floating and wrapping text around, 74–75  
     making columns with background, 108–112  
     preserving original Photoshop, 149  
     replacing for rollovers, 105–107  
     size and speeds of downloading, 129–130  
     slicing, 149–150  
     sprites, 101, 102, 105, 119–120  
 InDesign, 12, 21–23  
 index files, 26  
 inline tags, 53, 55–56  
 interactive features, 85–88  
   creating rollovers with images, 102–104  
   designing for, 85–86  
   using interactive CSS commands, 86–88  
 Internet  
   finding CSS3 support on, 115  
   navigating folders on, 21, 25

  packaging linked files for, 21–23  
   variation of Web design on, 6–8  
 Internet Explorer 6, 148, 160–161

## J

JavaScript, 38  
 JPEG file format, 146, 147

## K

  kerning, 73  
   keywords  
     positioning images with, 98–99  
     SEO, 136

## L

  layout, 74–85  
     about Web, 74  
     adding padding, 81–82  
     adjusting margins, 82–83  
     background images in, 94–101  
     borders in, 80–81  
     centering with margin declarations, 84  
     considering size of, 9, 10–11  
     designing with tables, 159  
     identifying content for, 142  
     inheriting border color, 81  
     planning for image production, 140–142  
     removing default spacing with CSS reset, 84–85  
     setting up columns with floats, 75–77, 78  
     spacing with border, padding, and margin, 78–79  
     using div, 160  
     wrapping text with floats, 74–75  
   leading, 72  
   linking  
     CSS with HTML via <link> tag, 136  
     with HTML anchor tags, 37  
     page files on Internet, 21–23

## lists

  definition, 30, 41–43  
   navigating using series of floating <li> tags, 78  
   ordered (numbered), 30, 37, 40–41  
   tagging, 29, 30, 39–43  
   unordered (bulleted), 30, 37, 39–40

## M

  margins  
     centering design using, 84  
     CSS shorthand for, 83–84  
     padding vs., 82  
     setting for layout, 82–83  
     spacing page with border, padding, and, 78–79  
   marking up pages in HTML, 8, 142–143  
 MeasureIt, 155  
 <media> attribute, 136  
 mouse  
   designing interactive buttons for, 86–88  
   users who don't use a, 128

## N

## navigating

- in and between folders on path, 25
- cueing users about, 86
- designing Web page for, 144
- to index, 26
- Internet folders, 21
- multiple folders with ./ and ../, 25
- using series of floating <li> tags, 78
- without a mouse, 128–129

## nested tags

- applying CSS within HTML, 61
- can't nest block tag within inline tag, 55
- targeting nested selectors in, 64–66
- using, 44–46

## numbered lists, 30, 37, 40–41

## O

## opening tags (&lt; &gt;), 35, 36

## ordered lists, 30, 37, 40–41

## overriding declarations in CSS, 104, 131–132

## P

<p> tag. *See* selectors

## padding

- adding, 81–82, 103
- CSS shorthand for, 83–84
- margins vs., 82
- spacing with padding property, 78–79

## page title for Web pages, 135

## paper vs. Web design, 5–6

## paragraph tags (&lt;p&gt;), 29, 36

## parent and child tags, 76–77

## paths

- absolute, 23
- importance of setting up, 18, 26
- jumping to root folder, 26
- linking page files on InDesign, 21–23
- navigating folders on, 21, 25
- relative, 24
- specifying background image, 95–96
- trailing slashes after folder name, 26
- understanding Web folders, 18–20

## percentages for image positioning, 100

## Photoshop

- adding matte color to GIF images in, 146–147
- Color Picker in, 71
- designing with, 11, 152
- Fireworks vs., 12, 152
- preserving originals in, 149
- Save for Web & Devices dialog, 145, 146, 147, 148
- saving selected images for production, 148–149
- setting up design projects in, 16
- slicing images, 149–150

## pixels, 10, 99–100, 149

## plug-ins, 154–155

## PNG file format, 147–148

## positioning background images, 98–100, 102

## printing

- identifying CSS file for, 136
- Web pages, 124–126

## pseudo-selectors, 87

## R

## relative paths, 24

## rendering engines, 114–115

## resolution

- horizontal and vertical, 10
- understanding computer, 14–15

## RGB color, 15–16, 72

## rollovers

- accessibility concerns for, 128
- changing default state of, 103–104
- designing with images, 102–104
- replacing images in, 105–107

## rounded corners in CSS3, 115–116

## S

## saving

- CSS files, 136, 153
- HTML files, 56, 135
- images for production, 148–149

## screen readers, 127

## screen size variation, 6–8, 9–11, 14

## scrolling pages, 10, 109

## selectors

- checking, 67
- defined, 59
- hooks and tags with, 63–64
- making interactive, 86
- overriding declarations with, 104, 131–132
- reading, 66
- stringing together, 130–131
- targeting nested, 64–66
- using, 62–66

## self-closing tags, 49–50

## SEO (search-engine optimization), 134, 135, 136

## servers, 20

## shadows, 116–119

## slicing images, 149–150

## spaces in CSS, 69

## &lt;span&gt; tag, 36

## special characters, 161–162

## sprites, 101, 102, 105, 119–120

## states

- changing rollover's, 103–104
- creating multiple rollover, 105, 119–120

## &lt;strong&gt; tag, 36

## syntax

- comment, 67
- elements for CSS, 58–59
- HTML attribute, 47



## T

tables for layout, 159

tags. *See also* lists

- adding attributes to, 46–49
- basic HTML, 29–31, 36–37
- block and inline, 53–56
- `<body>`, 136–137
- calling CSS with, 136
- centering with margin declarations, 84
- comment, 52–53
- definition list, 41–43
- errors in opening/closing, 45–46
- floating children, 76–77
- frequency of `<h1>` tags per page, 158–159
- importance in HTML, 34
- including Web site description in, 136
- lists, 29, 30, 37, 39–43
- nesting, 44–46
- page title for Web pages, 135
- parts of HTML, 35–36
- self-closing, 49–50
- structuring HTML files with, 134–137
- using selectors with hooks and, 63–64
- vertical height of columns and HTML, 108

## text

- case of HTML, 73, 74
- changing leading, 72
- clearing floats, 77
- coding special characters, 161–162
- color selection for, 71–72
- CSS cognates for, 2
- designating units of measurement, 70
- editing software for, 153
- Kerning, 73
- reusing and reformatting, 28
- setting with CSS, 68
- shadows for, 118–119
- tips for writing in HTML, 27–31
- working with fonts, 69–72, 125–126
- wrapping with floats, 74–75

## text editors, 153

## transparencies

- adding transparent shadows, 117
- PNG files supporting, 147, 148

## troubleshooting

- cautions about image replacement, 107
- checking opening/closing tags, 45–46
- inline tags nested in block tag, 55
- organizing files for Web pages, 21–23
- using Firebug for, 154–155

type. *See* text

## U

- units of measurement, 70
- unordered lists, 30, 37, 39–40
- uploading files, 20

## V

- vertical alignment, 99, 108–112
- vertical resolution, 10

## viewing

- content on page vs. RSS reader, 29
- HTML files in browsers, 56
- identifying CSS file for, 136
- visited command, 86

## W

Web design. *See also* Web image production; Web pages

- art of, 162

- challenges for designers, 5–6
- color design in RGB, 15–16
- considering layout size, 9, 10–11
- content design for “fold,” 160
- controlling page printing, 124–126
- creating files for, 11–16
- div layouts for, 160
- frequently asked questions about, 158–162
- linking page files on Internet, 21–23
- Photoshop settings for files, 16
- points to consider in CSS, 124
- recommended books on, 162
- tools needed for, 11–16, 151–155
- understanding computer resolution, 14–15
- using HTML for, 8–9
- using tables for layout, 159
- variations in screen size and, 6–8, 9–11, 14
- workflow for, 13–14

## Web image production, 139–150

- choosing final images for, 143–144
- GIF images for, 144–147
- JPEG images for, 146, 147
- planning layouts for, 140–142
- PNG images for, 147–148
- saving selected images for, 148–149
- slicing images, 149–150

Web pages. *See also* coding; Web image production

- accessibility of, 127–129
- browsing code for bugs, 154–155
- creating files for, 11–16
- CSS coding for sample, 90–91, 119–122
- designing with InDesign, 12
- errors in opening/closing tags, 45–46
- formatting in HTML, 50–51
- frequency of `<h1>` tags per, 158–159
- HTML tags needed for, 134–135
- linking files for, 21–23
- marking up mockups in HTML, 8, 142–143
- measuring, 155
- page title for, 135
- planning final layouts for, 140–144
- resources for, 81, 158
- saving and viewing HTML, 56
- scrolling, 10, 109
- structure of HTML, 134
- white space and file size of, 62
- writing example, 88–92

## X

- X- and Y-axis, 97