

Making Everything Easier!™

Beginning HTML5 & CSS3

FOR
DUMMIES[®]
A Wiley Brand

Learn to:

- Work with rules, style sheets, and proper syntax
- Build dynamic web pages using solid structure and the latest tools
- Avoid common HTML5 and CSS3 challenges and mistakes

IN FULL COLOR!

**Ed Tittel
Chris Minnick**



Beginning HTML5 & CSS3

FOR
DUMMIES[®]
A Wiley Brand

by Ed Tittel and Chris Minnick

FOR
DUMMIES[®]
A Wiley Brand

Beginning HTML5 & CSS3 For Dummies®

Published by
John Wiley & Sons, Inc.
111 River Street
Hoboken, NJ 07030-5774
www.wiley.com

Copyright © 2013 by John Wiley & Sons, Inc., Hoboken, New Jersey

Published simultaneously in Canada

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without the prior written permission of the Publisher. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permissions>.

Trademarks: Wiley, For Dummies, the Dummies Man logo, Dummies.com, Making Everything Easier, and related trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and may not be used without written permission. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc. is not associated with any product or vendor mentioned in this book.

LIMIT OF LIABILITY/DISCLAIMER OF WARRANTY: THE PUBLISHER AND THE AUTHOR MAKE NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE ACCURACY OR COMPLETENESS OF THE CONTENTS OF THIS WORK AND SPECIFICALLY DISCLAIM ALL WARRANTIES, INCLUDING WITHOUT LIMITATION WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE. NO WARRANTY MAY BE CREATED OR EXTENDED BY SALES OR PROMOTIONAL MATERIALS. THE ADVICE AND STRATEGIES CONTAINED HEREIN MAY NOT BE SUITABLE FOR EVERY SITUATION. THIS WORK IS SOLD WITH THE UNDERSTANDING THAT THE PUBLISHER IS NOT ENGAGED IN RENDERING LEGAL, ACCOUNTING, OR OTHER PROFESSIONAL SERVICES. IF PROFESSIONAL ASSISTANCE IS REQUIRED, THE SERVICES OF A COMPETENT PROFESSIONAL PERSON SHOULD BE SOUGHT. NEITHER THE PUBLISHER NOR THE AUTHOR SHALL BE LIABLE FOR DAMAGES ARISING HEREFROM. THE FACT THAT AN ORGANIZATION OR WEBSITE IS REFERRED TO IN THIS WORK AS A CITATION AND/OR A POTENTIAL SOURCE OF FURTHER INFORMATION DOES NOT MEAN THAT THE AUTHOR OR THE PUBLISHER ENDORSES THE INFORMATION THE ORGANIZATION OR WEBSITE MAY PROVIDE OR RECOMMENDATIONS IT MAY MAKE. FURTHER, READERS SHOULD BE AWARE THAT INTERNET WEBSITES LISTED IN THIS WORK MAY HAVE CHANGED OR DISAPPEARED BETWEEN WHEN THIS WORK WAS WRITTEN AND WHEN IT IS READ.

For general information on our other products and services, please contact our Customer Care Department within the U.S. at 877-762-2974, outside the U.S. at 317-572-3993, or fax 317-572-4002. For technical support, please visit www.wiley.com/techsupport.

Wiley publishes in a variety of print and electronic formats and by print-on-demand. Some material included with standard print versions of this book may not be included in e-books or in print-on-demand. If this book refers to media such as a CD or DVD that is not included in the version you purchased, you may download this material at <http://booksupport.wiley.com>. For more information about Wiley products, visit www.wiley.com.

Library of Congress Control Number: 2013942775

ISBN 978-1-118-65720-1 (pbk); ISBN 978-1-118-69075-8 (ebk); ISBN 978-1-118-69070-3 (ebk)

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

Table of Contents

Introduction 1

- About this Book 2
- Foolish Assumptions 3
- Icons Used in This Book 4
- Beyond the Book 4
- Where to Go from Here 5

Part I: Getting Started with HTML and CSS on the Web... 7

Chapter 1: An Overview of HTML and CSS on the Web 9

- How and Where Web Pages Come to Life Online 10
 - HyperText 10
 - Content versus presentation 14
 - Web browsers 14
 - Getting to know Internet protocols 16
- Understanding HTML and Its Versions 17
 - Different versions of HTML 17
 - Creating HTML markup 18
 - Building HTML documents 19
- Understanding the Role of CSS 20
 - Different versions here, too 20
 - Creating CSS markup 21
- Dissecting a Simple Markup Example 22
 - Where's the HTML? 22
 - Where's the CSS? 23
 - A partnership of equals 23

Chapter 2: Meeting the Structure and Components of HTML. 25

- Like Any Language: Syntax and Rules 25
 - Color-coding the markup 26
 - Breaking down the elements 27
- Adding Attributes to Your HTML 29
- Examining Entities in Markup 30
 - Non-ASCII characters 30
 - Character codes 31
 - (Special) tag characters 32
- Organizing Web Pages 32
 - Organizing HTML text 34
 - Complementing and enhancing text 36

Chapter 3: Creating and Viewing a Web Page 37

Before You Get Started	37
Creating a Page from Scratch.....	39
Step 0: Gather your tools	39
Step 1: Planning a simple design.....	40
Step 2: Writing some HTML.....	41
Step 3: Saving your page	44
Step 4: Viewing your page.....	46
Editing an Existing Web Page.....	47
Posting Your Page Online	49

Part II: Getting the Structure and Text Right 51**Chapter 4: HTML Documents Need Good Structure. 53**

Establishing a Document Structure.....	53
HTML Document Organization Revisited	54
HTML DOCTYPE Starts Things Off	55
The <html> Element	56
Anatomy of the <head>	56
Meeting the <head> himself.....	57
Handling metadata with <meta>	57
Redirecting users to another page	58
Naming your page with a <title>	61
The <body> Is a BIG Container	61

Chapter 5: Text and Lists. 63

Formatting Text	63
Paragraphs.....	65
Headings.....	66
Controlling Text Blocks	68
Block quotes	68
Preformatted text.....	69
Horizontal rules.....	71
Organizing Information	73
Numbered lists	73
Bulleted lists	75
Definition lists	77
Nesting lists	79

Chapter 6: Tip-Top Tables in HTML. 81

How <table> Got a Bad Name in HTML.....	81
What's in a Table? LOTS of Markup	82
Setting Up a Table Border	84



The Table Head (<thead>) and Its Elements.....	85
Managing Table Layouts.....	87
Making Good Table Bodies.....	89
Shaping a solid table.....	89
Sitting at the Footer of the Table.....	92
Exploring and Explaining a Table.....	92
Oh caption, my caption.....	93
Is the header dead yet?.....	93
Marching through the table body.....	94
Finishing with the footer.....	94

Chapter 7: Working with Forms in HTML..... 97

Exploring Types of Web Forms.....	97
Search forms.....	98
Data collection forms.....	99
Creating Forms.....	100
Structure.....	101
Input tags.....	102
Input fields.....	103
Form validation.....	113
Processing Data.....	115
Processing forms on your pages.....	115
Designing User-Friendly Forms.....	117
Other Noteworthy Forms-Related Markup.....	118
Form Frameworks.....	120

Part III: Adding Links, Images, and Other Media..... 123

Chapter 8: Getting Hyper with Links in HTML..... 125

Basic Links 101.....	125
Exploring link options.....	127
Avoiding common mistakes.....	129
Customizing Links.....	130
Opening new windows.....	130
Specifying locations in web pages.....	132
Linking to non-HTML resources.....	134

Chapter 9: Working with Images in HTML..... 139

The Role of Images in a Web Page.....	139
Creating Web-Friendly Images.....	140
Adding an Image to a Web Page.....	142
Image location.....	142
Using the element.....	143

Adding alternative and title text	143
Specifying image size.....	146
Image borders and alignment.....	149
Images That Link.....	149
Triggering links	149
Building image maps	150

Chapter 10: Managing Media and More in HTML 153

The Battle of the Media Formats	154
Meet the major audio formats.....	155
Meet the major video formats	156
Comparing Traditional and HTML5 Media Handling	157
Mastering HTML5 Media Markup	158
Making beautiful music with audio.....	158
Moving media with video.....	159
Undergoing the conversion experience.....	162
Working with Web Page Controls.....	163
Displaying a meter bar	163
Tracking progress on activities	165
Tracking and reporting on time	166
Updating HTML5 controls.....	168

Part IV: Adopting CSS Style 169

Chapter 11: Advantages of Style Sheets 171

Advantages of Style Sheets.....	172
The four steps to style	173
Understanding the C in CSS.....	174
What CSS can do for a web page.....	174
Styling a Document with CSS	175
Using HTML5 Boilerplate	176
Normalize before you stylize.....	176
What you can do with CSS.....	180
Putting CSS in Its Place	182
Pixels, points, and dots — Oh my!.....	182
Understanding the viewport.....	183
Property measurement values	184
About the CSS3 Standard.....	186

Chapter 12: CSS Structure and Syntax 191

Exploring CSS Structure and Syntax	191
Selectors and declarations	194
The selectors.....	195
Inheriting styles	204
Understanding the Cascade	205

Chapter 13: Using Different Kinds of Style Sheets 207

Applying Inline Styles.....	208
Getting to Know Internal Style Sheets	210
Understanding the <style> element.....	210
Figuring out internal style sheet scope.....	210
Working with External Style Sheets	212
CSS files	212
Link element attributes.....	213
Importing and when to use @import	214

Part V: Enhancing Your Pages' Look and Feel* 215*Chapter 14: Managing Layout and Positioning 217**

Managing Layout	217
Tiny boxes	217
Block versus inline elements.....	219
Normal flow	222
Managing Positioning.....	225
About coordinates and offsets.....	226
Relative positioning.....	226
Absolute positioning	227
Floating.....	228
Using a Layout Generator.....	230

Chapter 15: Building with Boxes, Borders, and Buttons 233

Meeting the Box Model.....	233
Putting the Box Model into Practice	235
Specifying padding and margin widths.....	239
Adding borders	243
Aligning text.....	246
Indenting text	247
Creating buttons with CSS	247

Chapter 16: Using Colors and Backgrounds 251

Defining Color Values.....	251
Color names.....	251
Color numbers.....	253
Defining Color Definitions.....	255
Text.....	256
Links	256
Backgrounds.....	258
Advanced backgrounds	259

Chapter 17: Web Typography	261
Finding Out about Fonts	261
Font family	262
Sizing	265
Trying Out Text Treatments	268
Embolden with bold	268
Emphasizing with italic	269
Changing capitalization.....	270
Getting fancy with the text-decoration property.....	271
Checking Out the Catchall Font Property	272
Experimenting with Web Fonts.....	273
Font file formats	273
Finding fonts	274
Linking fonts	274
Using Google Fonts	275
Chapter 18: CSS Text and Shadow Effects	281
Creating Shadows	282
text-shadow	282
box-shadow.....	283
Creating Inset Text	284
Creating 3D Text	285
Creating a Letterpress Effect.....	286
Drop Shadows	287
Text Rotation.....	289
Chapter 19: Multimedia and Animation with CSS	291
Using CSS with Multimedia.....	291
Visual media styles	293
Paged media styles	299
Getting Animated	300
Using the animation properties	302
Creating animations with @keyframes.....	303
Animating color.....	303
Part VI: The Part of Tens	305
Chapter 20: Ten Keys to Mobile Web Design	307
Design for Different Mobile Devices.....	307
Design for People.....	310
Design for Small Screens	310
Design for Low Bandwidth	311

Design for Touch	311
Design for Distracted Surfers.....	313
Test on Many Mobile Devices	313
Design for Simplicity	314
Set Up Mobile Web Addresses.....	314
Include a Link to the Desktop Site.....	315
Chapter 21: Ten HTML Do's and Don'ts	317
Don't Lose Sight of Your Content	317
Do Structure Your Documents and Your Site.....	318
Do Make the Most from the Least	318
Do Build Attractive Pages.....	319
Don't Lose Track of Those Tags	319
Do Avoid Browser Dependencies	320
Don't Make It Hard to Navigate Your Wild and Woolly Web	321
Don't Think Revolution, Think Evolution.....	322
Don't Get Stuck in the Two-Dimensional-Text Trap.....	323
Don't Let Inertia Overcome You.....	323
Chapter 22: Ten Ways to Kill Web Bugs Dead	325
Make a List and Check It — Twice.....	325
Master Text Mechanics.....	326
Lack of Live Links — a Lousy Legacy.....	327
When Old Links Must Linger	328
Make Your Content Mirror Your World.....	328
Look for Trouble in All the Right Places.....	328
Cover All the Bases with Peer Reviews	329
Use the Best Tools of the Testing Trade	330
Schedule Site Reviews.....	330
Foster User Feedback.....	331
If You Give to Them, They'll Give to You!	332
Chapter 23: Ten Cool HTML Tools and Technologies	333
WYSIWYG HTML Editors	334
Dreamweaver.....	334
Other WYSIWYG editors	335
Helper HTML Editors	335
Aptana Studio	335
Other helper editors.....	336
Inexpensive Graphics Editors	337
Professional Graphics Editors	337
Adobe Photoshop	338
Adobe Fireworks	338



Beginning HTML5 & CSS3 For Dummies

W3C Link Checker.....	339
Other Link Checkers.....	339
HTML Validators.....	340
FTP Clients.....	341
Miscellaneous Helpful Web Tools	341

Part VII: Appendixes..... 343

Appendix A: Twitterati 345

Appendix B: About the Dummies HTML Site. 349

About WordPress	349
The dashboard	349
Appearance and themes	350
Pages and posts	351
Widgets	351
Responsive Design.....	352
HTML5 Cafe	352
The home page.....	352
About Us.....	354
The Menu	354
Contact Us.....	354
HTML5 Boilerplate.....	355

Index..... 357

Introduction



Anyone can create or edit web pages. Crafting such pages doesn't require an especially high IQ or an advanced degree. Creating or editing web pages simply requires a desire to learn and enough gumption to see the process through to its natural end — a page visible on the web.

In this book, we reveal the ins and outs of the markup languages that are the web's lifeblood — the HyperText Markup Language (HTML) used to capture text, graphics, and other content, and the Cascading Style Sheets (CSS) language used to make web pages look good wherever they appear. Because HTML and CSS are basic building blocks for creating web pages, knowing how to use them adds you to the fold of web authors and content developers.

If you've tried to build your own web pages but found it too daunting, it's okay to relax now. If you can dial a telephone or find your keys in the morning, you too can create web pages. No kidding!

This book keeps the technobabble to a minimum and sticks with plain English whenever possible. Besides plain talk about hypertext, HTML, and the web, we include lots of examples, plus tag-by-tag instructions to help you build web pages with minimal fuss and bother. We also provide examples about what to do with your web pages after you've built them, so you can publish them online. We explain the differences between various flavors of HTML (HTML4, HTML5, and even something called XHTML) so you can pick the style that works best for you. Spoiler alert: We think you should choose HTML5, but that choice is entirely up to you.

This book has its own companion website with HTML and CSS examples from all of its chapters in usable form. In addition to the book content, we share web-only content and live pointers to all of the widgets, websites, and other cool stuff to which we refer, so you can use the techniques we show you to embellish your own web pages and amaze your friends. Please visit www.dummieshtml.com/html5cafe and start browsing from there. (Appendix B in this book covers all of that material in more detail.)

About this Book

Think of this book as a friendly, approachable guide to taking up HTML and CSS and building readable, attractive web pages. These things aren't hard to pick up, but they pack lots of details. Topics covered in this book include the following:

- ✓ Understanding web page structure and organization
- ✓ Uploading and publishing web pages for the whole world to see
- ✓ Checking and validating your web pages
- ✓ Diving deep into markup with HTML5 and CSS3

You too can build web pages without years of arduous training, advanced aesthetic abilities, or ritual ablutions in ice-cold streams. If you can tell a friend how to prepare your favorite mac-'n'-cheese, you can build a useful web document. The purpose of this book isn't to turn you into a rocket scientist (or for that matter, to turn rocket science into HTML). Its purpose is to show you the structural and technical elements needed for good-looking, readable web pages and to give you the confidence to build some!

This book explains how to use HTML and CSS to get your pages up and running on the World Wide Web. We tell you what's involved in structuring and building effective web documents that can bring your ideas and information to the online world — if that's what you want to do — and maybe even have some high-tech fun communicating them to others.

To make this book easier to read, keep in mind the following things about working with the markup:

- ✓ As a convention for this book, all HTML and CSS markup appears in monospaced type like this:

```
<head><title>What's in a Title?</title></head>
```
- ✓ When you type HTML markup, CSS, or other related stuff, copy the information exactly as you see it, including the angle brackets (< and >) because they're part of the magic that makes HTML and CSS work.
- ✓ The margins on a book page don't have the same room as do the vast reaches of cyberspace. Therefore, long lines of HTML and CSS markup, or designations for web sites (called *URLs*, or *Uniform Resource Locators*), may break across multiple lines. Remember, your computer sees such lines as a *single line of HTML or CSS*, or as a single URL — so if you type all of that text, be sure to put it all on one line. Don't insert any hard returns (or press the Enter key) if you see the line wrap. We show

you that everything is supposed to be *all on one line* by breaking at a punctuation character or space and then indenting any overage, like so:

```
www.infocadabra.transylvania.co/nexlus /plexus/lexus/  
praxis/okay/this-is-all make-believe-but-real-  
ones-get LONG.html
```

- ✓ HTML4 doesn't care whether you type tag text in uppercase, lowercase, or both (except for character entities, also known as character codes). HTML5 and CSS, however, want tag text in lowercase only. Thus, to make your work look as much like ours as possible, enter all HTML and CSS tag text, and all other markup, *in lowercase only*.
- ✓ Our code listings may be color-coded, where specific colors signify different kinds of markup. We explain this in Chapter 2 in the section about color-coding. (Note: All illustrations use pretty colors, too!)

One more thing: Readers may notice that we refer to the web, websites, and so forth in this book, even though we also call it the World Wide Web. We've decided to follow common usage, which no longer treats "web" as a proper name. Finally, the wheels of progress have turned long enough to wear off the top of the initial capital "W" in web!

Foolish Assumptions

Some say that making assumptions makes a fool out of both the person who makes them and the person who falls subject to them. (And just who are *they* anyway? We *assume* we know but . . . never mind.)

You don't need to be a wizard in the arcane arts of programming, nor do you require a PhD in computer science. You don't even need a detailed sense of what's going on in the innards of your computer to deal with the material in this book.

Even so, practicality demands that we make a few assumptions about you, our gentle reader: You can turn your computer on and off, you know how to use a mouse and a keyboard, and you want to build your own web pages for fun, profit, or some reason entirely of your own. We also assume you have a working Internet connection and a web browser.

If you can write a sentence and know the difference between a heading and a paragraph, you can build and publish your own documents on the web. The rest consists of details — and we help you with those.

Icons Used in This Book



Here's a list of the icons we use in this book to flag text and information that's especially noteworthy.

This icon signals technical details that are informative or interesting but aren't absolutely essential for writing or understanding HTML and CSS.



This icon flags useful information that makes HTML markup or other important stuff even less complicated than you feared it might be.



This icon points to stuff you shouldn't skip — don't overlook these reminders. (The sanity or web page you save could be your own.)



Watch out when you see this icon. It warns you against things you shouldn't attempt. Consequences can be severe if you ignore these admonitions.



This icon points you to resources available online. Most notably, we steer you to www.dummieshtml.com/html5cafe when we discuss example files you can find there.

Beyond the Book

This section describes where readers can find the book's companion content. Some of it is available at www.dummies.com, and some of it — including all the markup examples in the book — is available at www.dummieshtml.com/html5cafe:

- ✓ **Cheat Sheet:** Visit www.dummies.com/cheatsheet/beginninghtml5css3 to see a quick compendium of HTML and CSS markup, plus some handy-dandy color charts.
- ✓ **Extras:** We've posted articles that extend the content covered in the book, with one extra short article for Parts II through IV of this book. Parts II and III deal with HTML, and Parts III and IV with CSS. The URL for this material is www.dummies.com/extras/beginninghtml5css3.
- ✓ **Updates:** Each *For Dummies* technical book explains where readers can find updates in case the book changes substantially. This is where any

updates or corrections that we make to the book's content and coverage will appear, along with any errata we find and fix. The URL for this stuff is also www.dummies.com/extras/beginninghtml5css3.

For example, our book is chock-full of HTML5 and CSS 3 markup, and the specifications for both HTML5 and CSS3 are still in development, so changes are bound to occur in the months and years ahead.

- ✓ **Companion files:** Our book site offers per-chapter downloads with the source HTML and/or CSS files for each chapter, and a one-shot-gets-everything download for the whole book, all at www.dummieshtml/html5cafe. See Appendix B for details about the Dummies HTML website.

Where to Go from Here

This is where you hit the road. Where you start doesn't matter. Don't worry — you can handle it. We know you're ready to have the time of your life. Enjoy!

6

Beginning HTML5 & CSS3 For Dummies

Part I

Getting Started with HTML and CSS on the Web



Visit www.dummies.com for more great *For Dummies* content online. Also, there's a website just for this book online at www.dummieshtml.com.

In this part . . .

- ✓ Taking in HTML from 10,000 feet (an overview)
- ✓ Understanding the role that Cascading Style Sheets (CSS) play on the web
- ✓ Digging into HTML-speak: markup, elements, tags, entities, and more
- ✓ Getting your web pages organized
- ✓ Creating and viewing your very first web page
- ✓ Moving pages from your PC to a web server online

An Overview of HTML and CSS on the Web

In This Chapter

- ▶ Bringing web pages to life
 - ▶ Understanding the role that HTML plays on web pages
 - ▶ Appreciating what CSS does to give web pages style
 - ▶ Exploring and analyzing simple markup examples
-

Welcome to the wonderful world of the web, HTML, and CSS. With just a little knowledge, some practice, and something to say, you can create your own little virtual acre of cyberspace or improve on existing work.



We use the term HTML throughout this book. Using this term lets us refer to the HyperText Markup Language in general, including both HTML4 and HTML5, plus XHTML), all in one go. Although HTML4 and HTML5 are different (and XHTML differs from both of them, too), they're all enough alike for this reference to make sense.

This book is your down-and-dirty guide to understanding web documents, sprucing up existing web pages, and crafting complex and exciting pages that use intricate designs, multimedia, and scripting.

The best way to start working with HTML is to jump right in, so that's what this chapter does: It explains the basics of how HTML and CSS work behind the scenes inside web pages, and it introduces you to their underlying building blocks. When you're done with this chapter, you'll know how HTML and CSS work so you can start creating or editing web pages right away — albeit very, very simple ones.

How and Where Web Pages Come to Life Online

Web pages can accommodate many kinds of content, such as text, graphics, forms, audio and video files, streaming media, and even interactive games.

Browse the web for only a moment or two, and you see a smorgasbord of information and content displayed in many ways. Every website is different, but all have one thing in common: HyperText Markup Language (also known as HTML). You also run into Cascading Style Sheets (CSS) regularly.

Regardless of what information a web page contains, every page is created using some form of HTML. HTML is the mortar that holds web pages together: graphics, text, and other information are the bricks. CSS tells web pages how they should look (and to some extent, behave) when on display.



HTML files that produce web pages are simple text files, whether those files contain HTML4, HTML5, or even XHTML. Same thing goes for CSS. Reliance on simple text files, or documents, explains why the web works as well as it does. Text is a universal way of representing data for computers. Any text file you create on a Windows PC — including any HTML or CSS file — works equally well on a Mac, Linux/Unix, or any other operating system.

But web pages aren't *merely* text documents. Web pages are made using special, attention-starved, sugar-loaded text called HTML or CSS. Each web page uses its own specific sets of instructions and directives that you include (along with your content) inside text files to specify what's on the page and how that page should look and behave. Stick with us to uncover everything you need to know about HTML and CSS!

HyperText

Special instructions in HTML permit lines of text to point to (that is, *link*) something else in cyberspace. Such pointers are called *hyperlinks*. Hyperlinks are the glue that holds the World Wide Web together. In your web browser, hyperlinks usually appear in blue and are underlined. When you click a hyperlink, it takes you somewhere else.



Hypertext or not, a web page is a text file, which means you can create and edit a web page in any application that creates plain text (such as Notepad or TextEdit). Some software tools offer fancy options (covered in Chapter 23) to help you create web pages, but they generate the same text files you can create using a plain-text editor. We recommend you start with a simple, free web page editor named Aptana Studio. Visit www.aptana.com, where you can download the program. (You can also find instructions for Windows, Mac OS, and Linux.)



Steer clear of word processors such as WordPad or Microsoft Word when creating HTML. These tools introduce all kinds of extra markup to web pages that you don't want gunking up your work. If you don't believe us, try creating a web page inside Word and then look at all the stuff it adds inside some other editor. You won't believe your eyes!

The World Wide Web comes by its name honestly. It's literally a web of online pages hosted on web servers around the world, connected in trillions of ways by hyperlinks that tie individual pages together. Without such links, the web would be just a bunch of isolated, stand-alone documents. Boo hoo!

Much of the web's value comes from its ability to link pages and other resources (such as images, downloadable files, and media of all kinds) on a single website, or across many websites. For example, USA.gov (www.usa.gov) is a *gateway* website — its primary function is to provide access to other websites. If you aren't sure which government agency handles first-time loans for homebuyers, or you want to take a tour of the Capitol, visit the site shown in Figure 1-1 for information.



Figure 1-1: USA.gov uses hyperlinks to help visitors locate government information.

Web browsers were created specifically for the purpose of reading HTML markup and displaying the resulting web pages such markup describes. Markup lives in a text file (along with your content) to give orders to a browser. For example, look at the web page shown in Figure 1-2. You can

see how the page is made up by examining its underlying HTML; its underlying CSS governs its formatting, layout, and appearance.

This page includes a graphic, a title that describes the page (HTML5 Cafe: Home), a brief welcome, navigation text, and not much else.

Here, different components of the page use different formatting:

- ✔ The title for the page appears in its browser tab.
- ✔ A brief and simple text navigation bar (HOME | ABOUT US | MENU | CONTACT US) appears at the top border.
- ✔ The welcome statement is a text heading in large-format type, followed by a brief description of what's there.
- ✔ A coffee cup image appears next, followed by our favorite morning slogan (powered by coffee).

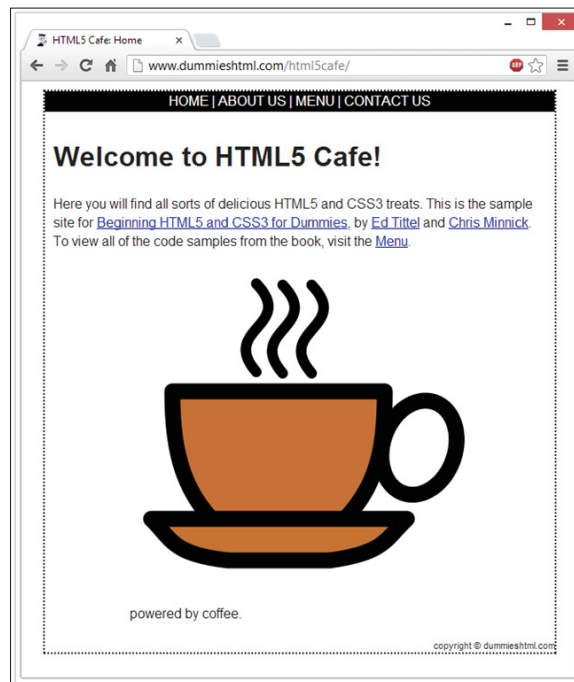


Figure 1-2: This page incorporates multiple parts and numerous bits of HTML and CSS.

The browser knows to display these components of the page in specific ways thanks to the somewhat simplified HTML markup for this page we present in Listing 1-1. Eventually we get around to all the real stuff that's on the actual web page, but for the moment, we present a stick-figure equivalent.

Listing 1-1: The HTML5 Cafe Home Page

```
<!DOCTYPE html>
  <head>
    <meta charset="utf-8">
    <title>HTML5 Cafe: Home</title>
    <meta name="description" content="sample site for 9781118657201">
    <meta name="viewport" content="width=device-width">
    <link rel="stylesheet" href="css/normalize.css">
    <link rel="stylesheet" href="css/main.css">
  </head>
  <body>
    <div id="container">
      <nav id="topnav">
        <a href="index.html">HOME</a> |
        <a href="about.html">ABOUT US</a> |
        <a href="menu.html">MENU</a> |
        <a href="contact.html">CONTACT US</a>
      </nav>
      <div id="content">
        <h1>Welcome to HTML5 Cafe!</h1>
        <p>Here you will find all sorts of delicious HTML5 and CSS3 treats.
          This is the sample site for <a href=
            "http://www.amazon.com/Beginning-HTML5-CSS3-Dummies-
            Computer/dp/1118657209">Beginning HTML5 and CSS3 for Dummies</a>,
          by <a href="http://www.edtittel.com">Ed Tittel</a> and
          <a href="http://www.chrisminnick.com">Chris Minnick</a>. To view
          all of the code samples from the book, visit the
          <a href="menu.html">Menu</a>.
        </p>
        <figure id="home-image">
          
          <figcaption class="warning">powered by coffee.</figcaption>
        </figure>
      </div>
      <footer>
        copyright &copy; dummieshtml.com
      </footer>
    </div>
  </body>
</html>
```

Nearly all text enclosed between angle brackets (less-than and greater-than signs, or `< >`) is an HTML *tag* (often called *markup*). For example, the *p* within brackets (`<p></p>` tags) identifies text inside paragraphs. The markup between `<head>` and `</head>` at the beginning of the document defines data that describes the entire document, including the character set it uses (`charset="utf-8"`), the title that appears on the browser tab, description and display information, and links to some standard style sheets to manage the look and feel. The markup between `<body>` and `</body>` contains everything you can actually see on the page (and some values that control how big the included coffee cup image appears). That's really all there is to it. You embed the markup in a text file, along with text for readers to see, to instruct the browser how to display your web page.



Tags and the content between (and inside) them are also called *elements*. Angle brackets `< >` enclose HTML markup; curly braces `{ }` enclose CSS markup. (You haven't seen those yet, but they show up in the next chapter.)

Content versus presentation

Simply put, *content* is stuff you can see on a web page. When developers talk about “web page content,” they often mean text information that appears on a web page. But images are content, too, as is any of the various types of multimedia that you find on many web pages nowadays, such as music, videos, animations, slide shows, and all kinds of other stuff. In general, HTML handles and packages content on web pages.

Equally simply, presentation is what stuff on a web page looks like when you see it. When web developers talk about “presentation,” they're referring to a multitude of characteristics. These include a plethora of typography controls for text (font family, font weight, font size, font color, and much more) but also precise positioning controls that can determine exactly where elements will appear as they're displayed. CSS includes hundreds of presentation controls, which define how web content looks and behaves when it's displayed somewhere, or printed, or even spoken (for those people making use of text-to-speech rendering facilities).

Web browsers

The user's primary tool in the web puzzle is called a *web browser*. Web browsers are programs that read HTML and CSS instructions and then use those instructions to make web page content appear on a screen.



Always write your HTML with the idea that people will view the content using a web browser. Just remember that there's more than one kind of browser out there, and each one comes in several versions.

Usually, web browsers request and display web pages that come from a web server on the Internet. But you can also display HTML pages you've saved on your own device before making them available on an Internet web server. When you develop your own HTML documents (web pages), you view those pages (called *local* pages) in your browser. You can use local pages to get a good idea of what people will see after those pages go live on the Internet.



Each web browser interprets HTML in its own way (though HTML5 is designed to improve this situation). Thus, the same HTML may not look exactly alike from one browser to the next. When you work with basic HTML, variations will be minor, but as you add other elements (such as scripting and multimedia), rendering markup can get hairy. Again, HTML5 is supposed to fix many such problems, but HTML5 isn't completely finished yet as we write this book, so it's still too early to tell whether that promise in theory will be kept in practice.



Chapter 3 explains how to use a web browser to view a local copy of your very first web page, in case you don't already know how to do this.

Some people use text-only web browsers such as Lynx because either:

- ✓ They're visually impaired and can't use a graphical display.
- ✓ They like a lean, fast web browser that displays only text.

A bevy of browsers

The web may be viewed through browsers of many types, each in its own versions, and each with its own feature sets. Some of the most popular web browsers include Microsoft Internet Explorer, Mozilla Firefox, Apple Safari, and Google Chrome. Other browsers, such as Lynx and Opera, are also widely used. As an HTML developer, you must think beyond your own browser experience and preferences. That's because every user has his or her own personal browser preferences and settings, and they are by no means all alike — not even close!

Each browser renders HTML a bit differently. Every browser handles JavaScript, multimedia, style sheets, and other add-ins differently, too. Throw in different operating systems, and a mix of smartphones and tablets, plus notebook and desktop PCs, and things get really interesting.

Usually differences between browsers are minor. But sometimes a combination of HTML, text, and media can bring a specific browser to its knees. When you work with HTML, test your pages on as many different browsers as you can. Install at least four browsers on your system for testing. We recommend the latest versions of Internet Explorer, Safari, Chrome, and Firefox.

If you want information about more browsers, Yahoo! maintains a fairly complete list (over 60 items altogether):

```
http://dir.yahoo.com/computers_
and_internet/software/
internet/world_wide_web/
browsers
```

Getting to know Internet protocols

Under the hood, the Internet works because of extraordinarily durable and capable sets of rules and formats for networked communication. These things are called protocols, and they define the ways in which computers can talk to each other across the Internet.

In fact the web is made up of billions of resources, each of them linkable. A resource's exact location is the key to linking to it. Without an exact address (a *Uniform Resource Locator*, or URL), you can't use the address bar in a web browser to visit a web page directly.



URLs are the standard addressing system for web resources. Each resource (web page, site, or individual file) has a unique URL. URLs work a lot like your postal address. Figure 1-3 identifies the components of a URL.

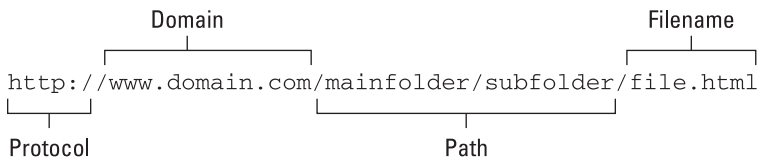


Figure 1-3: The components of a URL help it define an exact location for a file on the web.



The devil is in the protocol details

A collection of related protocols is often called a *protocol suite*. For the Internet, that protocol suite is TCP/IP taken from the abbreviation for the names of two of its most important protocols — namely the Transmission Control Protocol (TCP) and the Internet Protocol (IP). Together, in fact, both TCP and IP transport web communications safely across the Internet. They also support the HyperText Transfer Protocol, also known as HTTP, which is what moves web pages and ancillary materials (images, graphics, media, and so forth) around the Internet.

HTTP isn't the only protocol at work on the Internet riding atop TCP and IP. The Simple Mail Transfer Protocol (SMTP) and the Post

Office Protocol (POP) make e-mail possible, and the File Transfer Protocol (FTP) allows you to upload, download, move, copy, and delete files and folders across the Internet. The good news is that web browsers and web servers do all the HTTP work for you, so you need only put your pages on a server or type a URL into a browser.

To see how HTTP works, check out David Gourley and Brian Totty's chapter on HTTP Messages, available through Google Books. Go to <http://books.google.com>, search for "understanding http transactions," double-click *HTTP: The Definitive Guide* in the results, and browse around inside this excellent reference.

Each URL component helps to define the location of a web page or resource:

- ✓ **Protocol:** Specifies the protocol the browser should use to request the resource. This is usually HTTP but could be HTTPS (Secure HTTP), FTP, or something else.
- ✓ **Domain:** Points to the General website, such as `www.usa.gov`, where the resource resides. A domain may host a few files (like a personal website) or thousands of files (like a large government or corporate site, such as `www.usa.gov` or `www.ibm.com`).
- ✓ **Path:** Names the sequence of folders through which to navigate to get to a specific file or resource.

For example, to get to a file in the `services` folder that resides in the `system` folder, use the `/system/services/` path.

- ✓ **Filename:** Specifies which file in a directory path the browser is to access.

Although the URL shown in Figure 1-3 is not publicly accessible, it points to a domain and defines a path that leads to a specific resource named `file.html`:

```
http://www.domain.com/mainfolder/subfolder/file.html
```

Chapter 8 provides the complete details on how to use HTML and URLs to add hyperlinks to your web pages, and Chapter 3 shows how to obtain a URL for your website after you're ready to move it to a web server.

Understanding HTML and Its Versions

You already know that HTML's primary job is to label and accommodate content on web pages. But HTML comes in various versions, each of which handles content, but each of which is slightly different from the other. The basic rules and components stay more or less the same, but some important details differ. The following sections explore those versions and explain what makes them different.

Different Versions of HTML

HTML stands for HyperText Markup Language, markup developed in the late 1980s and early 1990s to describe web pages. HTML is now enshrined in numerous standard descriptions called *specifications* from the World Wide Web Consortium (W3C) and the Web HyperText Application Technology Working Group (WHATWG). Work on HTML specifications for versions 1–4 ended in 1999.

When you add an X in front of HTML, you get XHTML, a reworked version of HTML based on the *eXtensible Markup Language* (XML). XML was designed to work and behave well with computers, software, and the Internet.

The original versions (1–4, that is) of HTML included some irregularities that could cause heartburn for software that reads HTML documents. XHTML was designed to use an extremely regular and predictable syntax that’s easier for software to handle. XHTML was supposed to replace HTML, but increasing technical complexity in later versions caused it to fall by the wayside. (XHTML 2.0 was so complicated, it was neither widely adopted nor used very much at all.)

In 2004, the WHATWG began work on what is called a “Living Standard” for what is called HTML5 today. It’s been in process for a long time now, and the standard is finally nearing completion as we write this book. Some areas of HTML5 are still under development or subject to unresolved controversy. We steer clear of them in this book so that we can provide you with a solid foundation for your web pages for the foreseeable future.

HTML5 already appears to be succeeding where XHTML did not. Even though the standard is still under construction, HTML5 is widely adopted and used on the web today. In fact, the HTML5 specification is in what’s called “Candidate Recommendation” form as of December 2012. That’s one step before final Recommendation status is reached; most experts expect that final version to be approved and ratified in late 2013 or early 2014.

This book concentrates on the safe parts of HTML5, which use the same kind of regular and straightforward syntax that XHTML offered, but is much simpler to understand and use. Earlier books we’ve written show how to create both HTML and XHTML; in this book, we stick to HTML5, period.

Creating HTML markup

HTML is a straightforward language for describing web page contents. Its components are easy to use and come in three basic types:

- ✓ **Elements:** Identify different parts of an HTML document using tags.
- ✓ **Attributes:** Provide additional information about a particular instance of an element.
- ✓ **Entities:** Non-ASCII text characters, such as the copyright symbol (©) and accented letters (É). Entities come from the Standard Generic Markup Language (SGML) used to define early HTML versions.