LEARNING MADE EASY

HTML, CSS, & JavaScript®

ALL-IN-ONE

for dummies®

A Wiley Brand

5 Books in one!

Paul McFedries

Author of *Web Coding & Development All-in-One For Dummies*

HTML, CSS, & JavaScript®

ALL-IN-ONE

by Paul McFedries

for dummies®
A Wiley Brand

HTML, CSS, & JavaScript® All-in-One For Dummies®

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

Copyright © 2023 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. JavaScript is a registered trademark of Oracle America, Inc. 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: WHILE THE PUBLISHER AND AUTHORS HAVE USED THEIR BEST EFFORTS IN PREPARING THIS WORK, THEY 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 ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. NO WARRANTY MAY BE CREATED OR EXTENDED BY SALES REPRESENTATIVES, WRITTEN SALES MATERIALS OR PROMOTIONAL STATEMENTS FOR THIS WORK. THE FACT THAT AN ORGANIZATION, WEBSITE, OR PRODUCT IS REFERRED TO IN THIS WORK AS A CITATION AND/OR POTENTIAL SOURCE OF FURTHER INFORMATION DOES NOT MEAN THAT THE PUBLISHER AND AUTHORS ENDORSE THE INFORMATION OR SERVICES THE ORGANIZATION, WEBSITE, OR PRODUCT MAY PROVIDE OR RECOMMENDATIONS IT MAY MAKE. THIS WORK IS SOLD WITH THE UNDERSTANDING THAT THE PUBLISHER IS NOT ENGAGED IN RENDERING PROFESSIONAL SERVICES. THE ADVICE AND STRATEGIES CONTAINED HEREIN MAY NOT BE SUITABLE FOR YOUR SITUATION. YOU SHOULD CONSULT WITH A SPECIALIST WHERE APPROPRIATE. FURTHER, READERS SHOULD BE AWARE THAT WEBSITES LISTED IN THIS WORK MAY HAVE CHANGED OR DISAPPEARED BETWEEN WHEN THIS WORK WAS WRITTEN AND WHEN IT IS READ. NEITHER THE PUBLISHER NOR AUTHORS SHALL BE LIABLE FOR ANY LOSS OF PROFIT OR ANY OTHER COMMERCIAL DAMAGES, INCLUDING BUT NOT LIMITED TO SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR OTHER DAMAGES.

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 https://hub.wiley.com/community/support/dummies.

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: 2023940485

ISBN 978-1-394-16468-4 (pbk); ISBN 978-1-394-16497-4 (ebk); ISBN 978-1-394-16472-1 (ebk)

Contents at a Glance

Introduction 1

Book 1: Getting Started 5

CHAPTER 1: Getting Acquainted with HTML, CSS, and JavaScript 7
CHAPTER 2: Getting Ready to Code 39
CHAPTER 3: Finding and Setting Up a Web Host 47
CHAPTER 4: Testing and Validating Your Code 59

Book 2: Learning HTML Basics 77

CHAPTER 1: Building Good Bones: Structuring the Page 79
CHAPTER 2: Adding Links, Lists, and Other Text Tidbits 97
CHAPTER 3: Working with Images, Video, and other Media 123
CHAPTER 4: Building Tables with Your Bare Hands 143
CHAPTER 5: Using Forms to Make a Page Interactive 163
CHAPTER 6: Making Your Web Pages Accessible 181

Book 3: Learning CSS Basics 203

CHAPTER 1: Figuring Out the CSS Box Model 205
CHAPTER 2: Getting to Know the CSS Selectors 235
CHAPTER 3: Pseudo School: Learning Pseudo-Classes and Pseudo-Elements 251
CHAPTER 4: Making CSS Make Sense 287
CHAPTER 5: Taking the Measure of CSS 321
CHAPTER 6: Fancifying Pages with Colors and Backgrounds 337
CHAPTER 7: Taking Your Text Up a Notch with Web Typography 363

Book 4: Building Dynamic Pages with JavaScript 397

CHAPTER 1: JavaScript: The Bird's-Eye View 399
CHAPTER 2: Understanding Variables 411
CHAPTER 3: Building Expressions 427
CHAPTER 4: Controlling the Flow of JavaScript 457
CHAPTER 5: Harnessing the Power of Functions 481
CHAPTER 6: Playing with the Document Object Model 513
CHAPTER 7: Working with Arrays 553
CHAPTER 8: Manipulating Strings, Dates, and Numbers 583
CHAPTER 9: Storing User Data in the Browser 617
CHAPTER 10: Debugging Your Code 627
CHAPTER 11: Processing Form Data 649

Book 5: Looking Good with Layouts 673
CHAPTER 1: Exploring Some Layout Basics 675
CHAPTER 2: Getting Fancy with Flexbox 703
CHAPTER 3: Laying Out a Page with CSS Grid 737
CHAPTER 4: Creating Responsive Layouts 765

Index 801

Table of Contents

INTRODUCTION 1
About This Book 2
Getting started with the book 2
My "No AI" guarantee 2
Foolish Assumptions 3
Icons Used in This Book 3
Beyond the Book 4

BOOK 1: GETTING STARTED 5

CHAPTER 1: Getting Acquainted with HTML, CSS, and JavaScript 7
What Do HTML, CSS, and JavaScript Do, Exactly? 8
Why Learn HTML, CSS, and JavaScript? 8
Seeing How It All Works 10
To get started, launch a new text file 11
Notes about saving HTML files 11
This book in four words: Edit. Save. Reload. Repeat. 12
Adding Structure with HTML 14
What is HTML? 15
Getting the hang of HTML elements and tags. 15
Understanding tag attributes 17
A barebones HTML page 18
Some notes on structure versus style 23
Adding Style with CSS 24
Figuring out cascading style sheets 24
Getting the hang of CSS rules and declarations 26
Adding styles to a page 28
Adding Dynamism with JavaScript 32
What is JavaScript? 33
Adding a script to a page 34
The <script> tag 34
Where do you put the <script> tag? 34
Example #1: Displaying a message to the user. 35
Example #2: Writing text to the page 37

CHAPTER 2: Getting Ready to Code 39
Setting Up Your Test Browsers 40
Creating Your Local Folders 40
First, create the local root folder 41
Now create the local subfolders, if any 41

Choosing Your Text Editor 42
What Else Do You Need? 44
Using the WebDev Workbench 45

CHAPTER 3: **Finding and Setting Up a Web Host** 47
Understanding Web Hosting Providers 48
Using your existing internet provider 48
Finding a free hosting provider 48
Signing up with a commercial hosting provider 49
A Buyer's Guide to Web Hosting 49
Finding a Web Host 52
Finding Your Way around Your New Web Home 53
Your directory and your web address 54
Making your hard disk mirror your web home 55
Uploading your site files 56
Making changes to your web files 57

CHAPTER 4: **Testing and Validating Your Code** 59
Cross-Browser Testing Your Code 60
Taking a look at browser market share 60
Your cross-browser testing suite 61
Cross-browser testing online 61
Understanding Validation 62
Don't Skip Validation! 64
Validating Your HTML Code 65
Validating HTML online 65
Validating HTML in your text editor 67
Validating Your CSS Code 69
Validating CSS online 69
Validating CSS in your text editor 72
Validating Your JavaScript Code 72
Validating JavaScript online 73
Validating JavaScript in your text editor 74

BOOK 2: LEARNING HTML BASICS 77

CHAPTER 1: **Building Good Bones: Structuring the Page** 79
Getting to Know HTML's Basic Structure Tags 80
Creating paragraphs 81
Dividing your text with headings 83
Adding line breaks 85
Separating stuff with a horizontal rule 86
Carving Up the Page 87
The <header> tag 88
The <nav> tag 89

The <main> tag 90
The <article> tag 91
The <section> tag 91
The <aside> tag 92
The <footer> tag 93
Handling nonsemantic content with 94
Handling words and characters with 95
Commenting Your Code 95

CHAPTER 2: **Adding Links, Lists, and Other Text Tidbits** 97
Applying the Basic Text Tags 98
Emphasizing text 98
Marking important text. 99
Nesting tags 100
Adding quotations 100
Creating Links. 102
Linking basics 102
Anchors aweigh: Internal links. 103
Building Bulleted and Numbered Lists. 105
Making your point with bulleted lists 105
Numbered lists: Easy as one, two, three 107
Inserting Special Characters 108
A Few More HTML Text Tags to Know 109
<abbr>: The abbreviation text tag. 110
<address>: The address text tag 110
<cite>: The citation text tag 111
<code>: The code text tag 112
<dfn>: The definition text tag 113
<dl>: The description list text tag 114
<kbd>: The keyboard input text tag 116
<mark>: The mark text tag. 116
<pre>: The preformatted text tag 117
<s>: The strikethrough text tag 118
<sub>: The subscript text tag. 120
<sup>: The superscript text tag. 120

CHAPTER 3: **Working with Images, Video, and other Media**. . . . 123
Inserting Images 124
First, a mercifully brief look at image formats 124
Inserting an image. 125
Inserting a figure with a caption 127
Turning an image into a link 128
Controlling an image's vertical alignment 129

Embedding Videos 131
Embedding a third-party video 131
Embedding your own videos 132
Embedding Audio Snippets 137
Embedding third-party audio 137
Embedding your own audio 139

CHAPTER 4: **Building Tables with Your Bare Hands** 143
What Is a Table? 144
Web Woodworking: How to Build a Table 146
The simplest case: a one-row table 146
Adding more rows 148
Adding More Table Elements 149
Creating a header row 149
Including a caption 150
Creating a header column 151
Creating a table footer 152
Table Refinishing: Styling Your Tables 154
Aligning text within cells 154
Bring on the borders 159
Putting your data in a padded cell 161
Changing the location of the caption 162

CHAPTER 5: **Using Forms to Make a Page Interactive** 163
What Is a Web Form? 164
Cruising along Server Street 165
Taking a walk down Local Lane 166
Building a Web Form 166
Setting up the form 167
Adding a form button 167
Working with text fields 168
Adding checkboxes 171
Working with radio buttons 173
Adding selection lists 175
Bringing in pickers 178

CHAPTER 6: **Making Your Web Pages Accessible** 181
Why to Make Your Pages Accessible 182
Accessibility is a right 182
Accessibility brings other benefits 183
Understanding Web Accessibility 183
Understanding who needs accessibility 184
Learning about assistive technologies for web surfing 185
Making Your Page Structure Accessible 187
Using headings hierarchically 187
Using semantic sectioning elements 188

Making Text Accessible 188
Making Media Accessible 189
Specifying alt text for images 190
Making other media accessible 190
Buffing Up Your Page Accessibility Semantics 191
Adding form field labels 192
Understanding ARIA roles, states, and properties 192
Making Your Pages Keyboard-Friendly 198
Adding an element to the tab order 199
Removing an element from the tab order 200
Ensuring Sufficient Color Contrast 200
Validating the Accessibility of a Page 202

BOOK 3: LEARNING CSS BASICS 203

CHAPTER 1: Figuring Out the CSS Box Model 205
Thinking Outside (but Also Inside) the Box Model 206
Understanding the components of the box model 206
Understanding block and inline boxes 208
Changing the display type for a box 209
Eyeballing an element's box 210
A Brief Digression on Whitespace 211
Adding Padding 213
Putting a Border on It 215
Applying a border 215
Rounding your borders 218
Manipulating Element Margins 223
Taking advantage of negative margins 225
Collapsing margins ahead! 225
Styling Element Sizes 228
Setting a minimum or maximum height or width 231
Making width and height make sense 232
Magically converting an inline element to a block-level element 233

CHAPTER 2: Getting to Know the CSS Selectors 235
Introducing Yourself to the Web Page Family 236
What's All This About a Selector? 238
Seeing some examples of selectors 238
Working with selector lists 239
Learning the Standard Selectors 240
The type selector 240
The class selector (.) 240
The id selector (#) 242
The universal selector (*) 243
Selecting Descendants, Children, and Siblings 243

The descendant combinator ( ) 244
The child combinator (>) 245
The subsequent-sibling combinator (~) 245
The next-sibling combinator (+) 246
Selecting Elements by Attribute 247

CHAPTER 3: **Pseudo School: Learning Pseudo-Classes and Pseudo-Elements** 251

Scratching Your Head Over Pseudo-Classes 252
Introducing the pseudo-class 253
Styling elements with pseudo-classes 253
Matching child elements 255
Matching child elements by type 260
Matching form elements by state 264
Matching elements by user action 268
Matching links 270
Working with functional pseudo-classes 271
Getting Up to Speed with Pseudo-Elements 278
Working with pseudo-elements 278
Generating a last child with ::after 279
Generating a first child with ::before 281
Styling the opening letter with ::first-letter 283
Styling the opening line with ::first-line 285

CHAPTER 4: **Making CSS Make Sense** 287

Commenting Your CSS Code 288
Commenting non-obvious code 288
Marking stylesheet sections with comments 289
Using a CSS Reset 290
Debugging CSS 292
Displaying the web development tools 292
Inspecting an element 294
Editing a property value 299
Disabling a declaration 301
Adding an inline declaration to an element 302
Adding an element declaration to the inspector stylesheet 302
Adding a class to an element 304
Simulating a pseudo-class state 305
Popping the Hood: How Styles Get Applied 306
Unravelling inheritance 306
Flowing along with the cascade 309
How the cascade works 310
Understanding declaration types 310

Understanding origin types 311
Declaration type, origin type, and weight 311
Figuring out specificity 313
Specificity and :is(), :where(), :not(), and :has() 317
The ultimate tiebreaker: source code order 317
Putting it all together: the cascade algorithm 318

CHAPTER 5: **Taking the Measure of CSS** 321
Getting Comfy with CSS Measurement Units 322
Checking Out the Absolute Measurement Units 323
Getting to Know the Relative Measurement Units 324
Using percentages 325
Getting a handle on the em unit 325
Meeting your CSS measurement best friend: the rem unit 329
Here's Looking at View(port Measurement Units) 332
Calculating with CSS 333
Calculating with calc() 333
Finding the smallest value with min() 334
Finding the largest value with max() 335
Setting bounds with clamp() 335

CHAPTER 6: **Fancifying Pages with Colors and Backgrounds** 337
Specifying Colors in CSS 338
Papayawhip, peachpuff, and more: color keywords 338
Rolling your own colors with the rgb() function 339
Specifying colors nerd-style: RGB codes 340
Going pro with the hsl() function 341
Using Color to Spruce Up Your Text 344
Applying a text color 344
Setting the text decoration color 344
Creating a text shadow 345
Styling an Element's Background 346
Applying a coat of background color 346
Wallpapering an element with a background image 347
Taking advantage of the background shorthand property 352
Impressing Your Friends with Color Gradients 354
Applying a linear gradient 354
Applying a radial gradient 356
Applying a conic gradient 358
Setting Border Colors 359
Playing with Colors in the Dev Tools 359
Changing the color 360
Choosing accessible colors 361

CHAPTER 7: **Taking Your Text Up a Notch with Web Typography** 363
Taking Care of Your Text 364
Getting to know typefaces (and fonts, too) 364
All in the font family: Applying a typeface 365
Introducing the font stack 365
Generic font families 366
System font families 367
Web font families 368
The Typographic Trinity: Setting Type Size, Line Height, and Line Length 376
Setting the type size 377
Setting the line height 380
Setting the line length 382
How the typographical trinity are related 384
Applying Text Styles 384
Styling the text weight 384
Styling text with italics 386
Getting more styles with variable fonts 386
More Typographical Trickery 389
Controlling capitalization 389
Setting the spacing between letters 390
Displaying text as small caps 390
Using alternative numeric glyphs 391
The font shorthand property 392
Giving Your Links a Makeover 392
Messing with Alignment 394
Aligning text 394
Hyphenating text 395
Indenting a paragraph's first line 396

BOOK 4: BUILDING DYNAMIC PAGES WITH JAVASCRIPT 397

CHAPTER 1: **JavaScript: The Bird's-Eye View** 399
What Is Web Coding? 400
What Is a Programming Language? 401
Is JavaScript Hard to Learn? 402
What You Can Do with JavaScript 403
What You Can't Do with JavaScript 404
What Do You Need to Get Started? 404
Dealing with a Couple of Exceptional Cases 405
Handling browsers with JavaScript turned off 405
Handling very old browsers 407
Adding Comments to Your Code 408
Creating External JavaScript Files 408

CHAPTER 2: **Understanding Variables** 411
Understanding Variables 412
Declaring a variable with let 412
Storing a value in a variable 413
Checking out another way to declare a variable: const 414
Using variables in statements 415
Naming Variables: Rules and Best Practices 417
Rules for naming variables 417
Ideas for good variable names 418
Understanding Literal Data Types 419
Working with numeric literals 419
Working with string literals 421
Working with Boolean literals 423
JavaScript Reserved Words 423
JavaScript and HTML Keywords 424

CHAPTER 3: **Building Expressions** 427
Understanding Expression Structure 428
Building Numeric Expressions 429
A quick look at the arithmetic operators 429
Using the addition (+) operator 430
Using the increment (++) operator 430
Using the subtraction and negation (-) operators 431
Using the decrement (--) operator 433
Using the multiplication (*) operator 433
Using the division (/) operator 434
Using the modulus (%) operator 435
Using the arithmetic assignment operators 436
Building String Expressions 436
Building Comparison Expressions 439
The comparison operators 439
Using the equality (==) operator 439
Using the inequality (!=) operator 440
Using the greater than (>) operator 441
Using the less than (<) operator 441
Using the greater than or equal (>=) operator 442
Using the less than or equal (<=) operator 442
The comparison operators and data conversion 442
Using the strict equality (===) operator 444
Using the strict inequality (!==) operator 444
Using strings in comparison expressions 445
Using the ternary (?:) operator 446
Building Logical Expressions 447
The logical operators 447
Using the AND (&&) operator 447

Using the OR (||) operator. 448
Using the NOT (!) Operator 449
Advanced notes on the && and || operators. 450
Understanding Operator Precedence 452
The order of precedence 452
Controlling the order of precedence 453

CHAPTER 4: **Controlling the Flow of JavaScript** 457
Making True/False Decisions with if Statements 458
Branching with if. . .else Statements 460
Making Multiple Decisions 462
Using the AND (??) and OR (||) operators 462
Stringing together multiple if statements 463
Using the switch statement 464
Understanding Code Looping 466
Using while Loops 467
Using for Loops 470
Using do. . .while Loops 474
Controlling Loop Execution 476
Exiting a loop using the break statement 476
Bypassing loop statements using the continue statement. 479
Avoiding Infinite Loops 480

CHAPTER 5: **Harnessing the Power of Functions** 481
What Is a Function? 482
The Structure of a Function 482
Where Do You Put a Function? 484
Calling a Function 484
Calling a function when the <script> tag is parsed. 484
Calling a function after the page is loaded 486
Calling a function in response to an event 487
Passing Values to Functions. 488
Passing a single value to a function 489
Passing multiple values to a function. 490
Making an argument optional 492
Returning a Value from a Function 494
Getting Your Head around Anonymous Functions 495
Assigning an anonymous function to a variable. 496
Replacing a function call with an anonymous function 497
Moving to Arrow Functions 498
Running Functions in the Future. 500
Using a timeout to perform a future action once. 501
Using an interval to perform a future action repeatedly. 502

Understanding Variable Scope 504
Working with block scope. 504
Working with function scope 505
Working with global scope 507
Using Recursive Functions 508

CHAPTER 6: **Playing with the Document Object Model** 513
Working with Objects 514
What is an object, anyway? 514
Manipulating object properties 515
Working with object methods 518
Rolling your own objects 519
The Web APIs: Some special objects. 521
Getting to Know the Document Object Model 522
Specifying Elements 525
Specifying an element by id 525
Specifying elements by tag name 526
Specifying elements by class name 526
Specifying elements by selector 527
Working with collections of elements. 527
Traversing the DOM 529
Getting the children of a parent element 530
Getting the parent of a child element 534
Getting the siblings of an element 534
Manipulating Elements. 535
Adding an element to the page 535
Inserting text or HTML into an element. 538
Removing an element. 539
Modifying CSS with JavaScript 540
Changing an element's styles. 540
Adding a class to an element. 541
Tweaking HTML Attributes with JavaScript 544
Reading an attribute value. 545
Setting an attribute value. 545
Removing an attribute 546
Building Reactive Pages with Events. 546
What's an event? 547
Understanding the event types 547
Getting data about the event. 549
Preventing the default event action 551

CHAPTER 7: **Working with Arrays** 553
What Is an Array?. 554
Declaring an Array. 554
Populating an Array with Data. 555

Declaring and populating an array at the same time. 557
Using a loop to populate an array. 558
How Do I Iterate Thee? Let Me Count the Ways 559
Iterating an array: forEach() 560
Iterating an array: for. . .of 561
Iterating to test an array's elements: every() and some() 562
Iterating to create a new array: map() 565
Iterating an array down to a value: reduce() 566
Iterating to locate an element: find() 568
Creating Multidimensional Arrays 569
Manipulating Arrays 570
The length property 570
Concatenating to create a new array: concat() 571
Creating a string from an array's elements: join(). 573
Removing an array's last element: pop() 574
Adding elements to the end of an array: push() 575
Reversing the order of an array's elements: reverse() 576
Removing an array's first element: shift() 576
Returning a subset of an array: slice() 577
Ordering array elements: sort(). 578
Removing, replacing, and inserting elements: splice(). 580
Inserting elements at the beginning of an array: unshift() 582

CHAPTER 8: **Manipulating Strings, Dates, and Numbers** 583
Manipulating Text with the String Object 584
Working with string templates 584
Determining the length of a string 586
Searching for substrings. 587
Methods that extract substrings 591
Dealing with Dates and Times 599
Arguments used with the Date object 600
Working with the Date object 600
Extracting information about a date. 601
Setting the date 605
Performing date calculations. 608
Working with Numbers: The Math Object 611
Converting between strings and numbers 612
The Math object's properties and methods. 614

CHAPTER 9: **Storing User Data in the Browser** 617
Understanding Web Storage 618
Introducing JSON 618
Learning the JSON syntax. 619
Declaring and using JSON variables 620
Converting a JavaScript object to JSON. 621
Converting a JSON string to a JavaScript object 622

Adding Data to Web Storage 623
Getting Data from Web Storage 625
Removing Data from Web Storage 626

CHAPTER 10: **Debugging Your Code** 627
Understanding JavaScript's Error Types 628
Syntax errors 628
Runtime errors 628
Logic errors 629
Getting to Know Your Debugging Tools 630
Debugging with the Console 632
Displaying the console in various browsers 632
Logging data to the Console 633
Executing code in the Console 633
Pausing Your Code 634
Entering break mode 634
Exiting break mode 637
Stepping Through Your Code 637
Stepping one statement at a time 638
Stepping into some code 638
Stepping over some code 639
Stepping out of some code 639
Monitoring Script Values 640
Viewing a single variable value 640
Viewing all variable values 641
Adding a watch expression 642
More Debugging Strategies 643
The Ten Most Common JavaScript Errors 644
The Ten Most Common JavaScript Error Messages 647

CHAPTER 11: **Processing Form Data** 649
Looking at the HTMLFormElement Object 650
Taking a Peek at the HTMLInputElement Object 650
Programming Text Fields 651
Referencing text fields by field type 651
Getting a text field value 652
Setting a text field value 653
Coding Checkboxes 654
Referencing checkboxes 654
Getting the checkbox state 655
Setting the checkbox state 656
Dealing with Radio Buttons 657
Referencing radio buttons 657
Getting a radio button state 658
Setting the radio button state 659

Programming Selection Lists 660
Checking out the HTMLSelectElement object 660
Checking out the HTMLOptionElement object 661
Referencing selection list options 662
Getting the selected list option 662
Changing the selected option 664
Handling and Triggering Form Events 664
Setting the focus 665
Monitoring the focus event 666
Monitoring the blur event 666
Listening for element changes 667
Creating Keyboard Shortcuts for Form Controls 668
Dealing with the Form Data 670

BOOK 5: LOOKING GOOD WITH LAYOUTS 673

CHAPTER 1: Exploring Some Layout Basics 675
Getting a Grip on Page Flow 676
Floating Elements 677
Example: Creating a pull quote 679
Clearing your floats 681
Collapsing containers ahead! 683
Positioning Elements 685
Using relative positioning 687
Giving absolute positioning a whirl 688
Trying out fixed positioning 690
Making elements stick (temporarily) 692
Stacking Elements 696
Layering elements with z-index 696
Getting your head around stacking contexts 698

CHAPTER 2: Getting Fancy with Flexbox 703
Introducing Flexbox 704
Do I Still Need Flexbox Now That CSS Grid Is Here? 705
Setting Up and Configuring Flex Containers 705
Setting up the flex container 706
Touring the landscape of a flex container 706
Setting the flex direction 706
Aligning flex items along the primary axis 709
Aligning flex items along the secondary axis 711
Centering an element horizontally and vertically 713
Allowing flex items to wrap 714
Aligning rows or columns along the secondary axis 716
Adding gaps between items 718

Taking Control of Flex Items 719
Allowing flex items to grow 720
Allowing flex items to shrink 722
Suggesting a flex item size 726
Using the flex shorthand property 730
Laying out content columns with Flexbox 731
Changing the order of flex items. 734

CHAPTER 3: **Laying Out a Page with CSS Grid** 737
Introducing CSS Grid. 738
Setting Up the Grid Container 739
Specifying the Grid Rows and Columns 740
Setting your own columns and rows: the explicit grid. 741
Letting the browser do some of the work: the implicit grid 745
Creating grid gaps 749
Taking Control: Assigning Grid Items 750
Assigning grid items to columns and rows 750
Assigning grid items to named grid areas 757
Getting Your Grid Ducks in a Row (or a Column): Aligning Things . . . 761

CHAPTER 4: **Creating Responsive Layouts**. 765
What is a Responsive Layout? 766
Going with the Flow: Fluid Layouts 769
How Flexbox makes a page fluid. 770
How CSS Grid makes a page fluid 773
Taking advantage of viewport units 776
Making typography fluid. 778
Introducing your best fluid friend: clamp() 779
Querying Your Way to Responsiveness: Adaptive Layouts 779
Interrogating the screen with media queries 780
Laying out trees instead of forests with container queries. 783
Respecting your visitors with user preference queries 789
Working with Images Responsively. 791
Making images responsive. 792
Delivering images responsively 792
Exploring the Principles of Mobile-First Development. 795
What is mobile-first web development?. 795
Mobile first means content first 796
Pick a testing width that makes sense for your site 797
Get your content to scale with the device 798
Build your CSS the mobile-first way 798
Pick a “non-mobile” breakpoint that makes sense for your content. 799

INDEX 801

Introduction

Let me start off this book by letting you in on a little secret. If you talk to or read things written by people who make websites for a living, it's all "HTML this" and "CSS that." They go on and on about "tags" and "properties" and "collapsing margins" and blah blah blah. It can be more than a little intimidating, so you can easily come away with the idea that crafting a web page is *really* hard. You may end up believing that creating stuff for the web is a for-geeks-with-CS-graduate-degrees-only business.

Okay, it's time for that secret I just promised you. Ready? Come closer. Closer. Okay:

whispers *Learning how to build web pages is not hard.*

Sure, it *sounds* hard; and if you've ever taken a peek at some web page code, it certainly *looks* hard; and, I'll admit, building a huge and insanely complex site like Amazon or Instagram really *is* hard. But creating a personal website? Not hard. Fabricating a site to showcase a hobby? Not hard. Crafting some pages for a local charity, team, or small business? You got it: Not hard!

Still don't believe me? That's okay, I get it: HTML, CSS, and JavaScript — the technologies that enable anyone to assemble web pages — have a reputation problem. After all, so the thinking goes, people have used HTML, CSS, and JavaScript to sculpt some truly sensational sites, so *of course* such sophistication must come with a near-vertical learning curve. Duh.

For years now I've talked to many smart people who believed all that and who therefore wouldn't even dream of building a web page from scratch. How many awesome websites never got built because their would-be builders thought HTML, CSS, and JavaScript were well beyond their capabilities? Why is no one talking about how accessible these technologies really are?

After asking myself these questions over and over, I finally decided to do something about it. I decided to write this book, the aim of which is to prove to everyone — yes, even skeptical you — that the technologies behind the web are approachable, straightforward, and readily learnable.

About This Book

Welcome, then, to *HTML, CSS, and JavaScript All-in-One For Dummies*. This book gives you a complete education on the technologies that enable anyone to craft professional-looking web pages. You learn how to set up the tools you need, how to use HTML and CSS to design and build your site, and how to use JavaScript to program your pages. My goal is to show you that these technologies aren't hard to learn, and that even the greenest rookie web designer can learn how to put together pages that will amaze their family and friends (and themselves).

If you're looking for lots of programming history, computer science theory, and long-winded explanations of concepts, I'm sorry, but you won't find it here. My philosophy throughout this book comes from Linus Torvalds, the creator of the Linux operating system: "Talk is cheap. Show me the code." I explain what needs to be explained and then I move on without further ado (or, most of the time, without any ado at all) to examples and scripts that do more to illuminate a concept that any verbose explanations I could muster (and believe me, I can muster verbosity with the best of them).

Getting started with the book

How you approach this book depends on your current level of web coding expertise (or lack thereof):

- If you're just starting out, begin at the beginning with Book 1 and work at your own pace sequentially through to Books 2 and 3. This will give you all the knowledge you need to pick and choose what you want to learn throughout the rest of the book.
- If you know HTML and CSS, you can probably get away with taking a fast look at Books 2 and 3, and then settle in with Book 4 and beyond.
- If you've done some JavaScript coding already, I suggest working quickly through the material in Book 4, and then exploring the rest of the minibooks as you see fit.

My "No AI" guarantee

As I began updating this edition of the book, the world was awash in posts and talk and endless speculation about artificial intelligence, to the point where it seemed we'd soon be welcoming our new AI overlords. That's not likely to happen anytime soon, but AI is here to stay and has already established itself as a significant part of many people's workday routines.

I've been as enamored of ChatGPT and its ilk as the biggest AI boosters. I use AI for entertainment and curiosity, but I don't use it for work. That is to say that not one word of the text, code, or examples used in this book has been generated by AI. Everything you read here is, for good or ill, the product of my warped-from-birth brain.

Foolish Assumptions

This book is not a primer on the internet or on using the World Wide Web. This is a book on building web pages, pure and simple. This means I assume the following:

- You know how to operate a basic text editor, and how to get around the operating system and file system on your computer.
- You have an internet connection.
- You know how to use your web browser.

Yep, that's it.

I should mention here that this book is all about coding what web nerds call the *front end*, which means the stuff you see in your web browser. The web also has a *back end*, which refers to what happens on the web server. I don't get into any back-end coding in this book. If you're interested in that part of the web, may I not-so-humbly suggest my book *Web Coding and Development All-in-One For Dummies* (Wiley).

Icons Used in This Book

REMEMBER

This icon points out juicy tidbits that are likely to be repeatedly useful to you — so please don't forget them.

TIP

Think of these icons as the fodder of advice columns. They offer (hopefully) wise advice or a bit more information about a topic under discussion.

WARNING

Look out! In this book, you see this icon when I'm trying to help you avoid mistakes that can cost you time, money, or embarrassment.

TECHNICAL STUFF

When you see this icon, you've come across material that isn't critical to understand but will satisfy the curious. Think "inquiring minds want to know" when you see this icon.

Beyond the Book

Some extra content for this book is available on the web. Go online to find the following:

- **The examples used in the book:** You can find these in either of the following places:
  - **My website:** `https://paulmcfedries.com/htmlcssjsfd`
  - **GitHub:** `https://github.com/paulmcfe/html-css-js-fd`

  The examples are organized by book and then by chapter within each minibook. For each example, you can view the code, copy it to your computer's clipboard, and run the code in the browser.
- **The Web Dev Workbench:** To try your own HTML, CSS, and JavaScript code and see instant results, fire up the following site:

  `https://webdevworkshop.io/wb`

  You won't break anything, so feel free to use the site to run some experiments and play around with HTML, CSS, and JavaScript.
- **Bonus chapters:** I had much more to tell you than I could fit in this book, so go to `www.dummies.com/go/htmlcss&javascriptaiofd` to download additional chapters that will show you some amazingly powerful techniques and animations to raise your CSS game.
- **Cheat Sheet:** Go to `www.dummies.com` and search *HTML, CSS, & JavaScript All-in-One For Dummies* to find the Cheat Sheet for this book. Here you'll find a ready-to-roll HTML template for a basic web page, a set of 15 essential CSS selectors, and ten powerful JavaScript debugging strategies.

1 Getting Started

Contents at a Glance

CHAPTER 1: **Getting Acquainted with HTML, CSS, and JavaScript** 7

CHAPTER 2: **Getting Ready to Code** 39

CHAPTER 3: **Finding and Setting Up a Web Host** 47

CHAPTER 4: **Testing and Validating Your Code** 59

IN THIS CHAPTER

- Finding out a bit about what you're getting yourself into
- Befriending HTML
- Introducing yourself to CSS
- Shaking hands with JavaScript

Chapter 1

Getting Acquainted with HTML, CSS, and JavaScript

Well begun is half done.

— ANCIENT GREEK PROVERB

In the early days of the internet in general and of the web in particular, people often used the abbreviation RTFM, which stood for (in the bowdlerized version), *read the freaking manual*. In days of yore, software programs came with little booklets — called *manuals* — that described the workings of the program. Look-before-you-leap types would carefully read the manual and would therefore know how to use the program. But a sizable proportion of the population would rather leap than look, meaning they would just start using the software, poking around willy-nilly to try to make things happen. Those dedicated leapers would inevitably end up on message boards or forums, desperately seeking solutions to the problems their haphazard experimenting caused them. The answer, more than often than not, was a simple one: "RTFM!"

This book is a sort of manual writ large for using HTML, CSS, and JavaScript. However, and this is particularly true if you're just getting started with coding

web pages, if there's any part of the book that fits the RTFM credo, it's this chapter. *Everything* you learn in this chapter acts as a kind of home base for the explorations that come later, especially in Book 2 and beyond.

In this chapter, you learn the basic concepts behind HTML, CSS, and JavaScript, get a better understanding of how they work, and get started exploring these powerful technologies.

What Do HTML, CSS, and JavaScript Do, Exactly?

If you're new to the world of weaving web pages, you may be asking yourself a very basic — but a very *astute* — question about HTML, CSS, and JavaScript: What do they do?

The bird's-eye view is that HTML, CSS, and JavaScript are the technologies behind what appears when you visit a page on the web. Sure, your trusty web browser shows you the page text and images, but the way in which the text and images are presented to you is a function of the page's underlying HTML, CSS, and JavaScript code. These technologies have three separate but interrelated functions:

- **Structure:** The basic scaffolding of the page, such as the page headings, the text paragraphs, and where the images appear. This is the realm of HTML.
- **Style:** How the page looks, including the fonts, colors, and margins. This is the bailiwick of CSS.
- **Dynamism:** Extras that make the page perform actions such as interacting with the user, "listening" for mouse clicks and keypresses, and writing content to the page based on certain conditions. This is the job of JavaScript.

That's the big picture. I get into all this in a bit more detail later in this chapter (starting with HTML in the "Adding Structure with HTML" section).

Why Learn HTML, CSS, and JavaScript?

I mention in the Introduction that learning HTML, CSS, and JavaScript isn't hard. That's still true, but I must admit that it doesn't tell the entire story. Yes, learning these technologies isn't hard, but it's certainly not trivial, either. Although you