

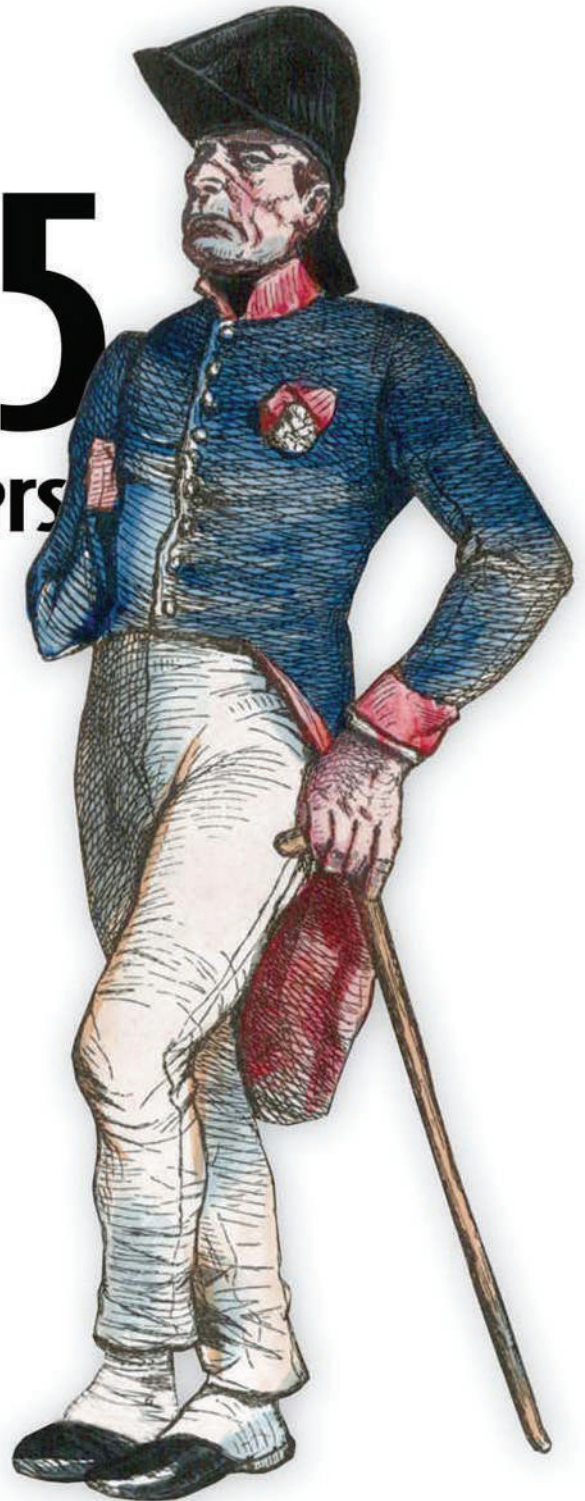
Single page web apps, JavaScript, and semantic markup

# HTML5

## for .NET Developers

Jim Jackson II  
Ian Gilman

FOREWORD BY  
Scott Hanselman



# contents

---

*foreword xv*  
*preface xvii*  
*acknowledgments xx*  
*about this book xxii*  
*about the cover illustration xxv*

## 1 **HTML5 and .NET 1**

### 1.1 New toys for developers thanks to HTML5 3

*New HTML5 tags and microdata 4* ▪ *HTML5 applications for devices 5* ▪ *Better, faster JavaScript 6*  
*Libraries, extensions, and frameworks 6* ▪ *New HTML5 JavaScript APIs 9* ▪ *Cascading Style Sheets 3 11*  
*MVC and Razor 12*

### 1.2 HTML5 applications end-to-end 12

*Page structure and page presentation 13* ▪ *Page content 14*  
*Application navigation 15* ▪ *Business logic 16*  
*Server communications 17* ▪ *The data layer 18*

### 1.3 Hello World in HTML5 19

*Creating the template 20* ▪ *Customizing the application 22*  
*Building the JavaScript library 24* ▪ *Building the server side 28*

### 1.4 Summary 32

- ## 2 ***A markup primer: classic HTML, semantic HTML, and CSS*** 33
- 2.1 Classic and semantic HTML markup: what's the difference? 34
  - 2.2 Basic structural elements of all HTML tags 35
    - Working with the basic HTML tags* 37
    - *Making content flow where you want with block and inline elements* 40
    - *Dividing data into grids with table elements* 42
    - *Using HTML form elements* 43
  - 2.3 Semantic HTML: The semantic blueprint 44
    - Grouping and dividing page content with content tags* 46
    - Going beyond semantics with application tags* 51
    - Using media tags for audio and video content* 53
  - 2.4 Styling HTML5: CSS basics 54
    - Understanding CSS syntax* 54
    - *Building selectors, the most critical CSS element* 55
    - *Assigning fonts* 56
    - *Assigning and manipulating colors* 57
    - *Changing the size of an element with the box model* 59
    - *Using columns and blocks for layout* 60
    - Changing screen layout based on changing conditions with media queries* 61
    - *Adjusting an element's presentation and location with transitions and transformations* 62
    - Changing styles as needed with pseudo-elements and pseudo-classes* 64
  - 2.5 Summary 65
- ## 3 ***Audio and video controls*** 66
- 3.1 Building a site to play audio and video 68
  - 3.2 Audio and video tags 71
    - Using audio and video tags without JavaScript* 72
    - *Using the audio tag as an HTML element* 73
    - *Using the video tag as an HTML element* 74
  - 3.3 Controlling audio and video playback with JavaScript 76
    - Building custom controls for audio and video* 76
    - *Building the main.js library structure* 78
    - *Creating a JavaScript media player object* 79
    - *Completing the media experience by adding volume controls* 80
  - 3.4 Updating media types for open source content 83
  - 3.5 Summary 85
  - 3.6 Complete code listings 86

- 4 Canvas 90**
- 4.1 Canvas quick-start 92
    - Creating the basic Canvas site structure* 93
    - *Assigning size to the canvas* 94
    - *Creating the stylesheet for the sample application* 95
    - *Drawing with the 2d context object* 96
    - Building the foundation object of the Canvas application* 96
  - 4.2 Creating and manipulating shapes, lines, images, and text 97
    - Understanding the basic drawing process* 98
    - Adding shapes* 100
    - *Adding lines* 101
    - Adding images* 103
    - *Manipulating pixels* 105
    - Adding text* 107
  - 4.3 Animating and adding special effects to canvas images 108
    - Adding animation* 108
    - *Adding special effects with curves and clipping* 111
    - *Managing canvas properties during screen resizing* 113
  - 4.4 Summary 114
  - 4.5 Complete code listing 114
- 5 The History API: Changing the game for MVC sites 118**
- 5.1 Building a History-ready MVC site 120
    - Launching the sample project in Visual Studio* 121
    - Adding controllers and views* 122
    - *History and MVC routing* 126
    - *Creating the application data model* 129
    - Loading content from the server on demand using partial views* 130
  - 5.2 Using HTML5 History 133
    - Adding JavaScript to handle History API navigation events* 135
    - Working with the page URL in JavaScript* 139
    - *Using History to update the page* 140
  - 5.3 Two more small steps ... 141
  - 5.4 Summary 144
  - 5.5 The complete JavaScript library 144
- 6 Geolocation and web mapping 147**
- 6.1 “Where am I?”: A (brief) geographic location primer 149
  - 6.2 Building a geolocation application 153
    - Basic application setup* 154
    - *Using the Bing Maps JavaScript API* 157

- 6.3 Using the Geolocation API 159
  - API functions for interacting with device location services* 161
  - Plotting a point on a map* 163
  - *Integrating geolocation and a map* 165
  - *Navigating the map using geolocation data* 167
- 6.4 Building a service to find address information 171
  - Modeling a point on the Earth in .NET* 171
  - Displaying routes between coordinates* 174
- 6.5 Summary 178
- 6.6 Complete code listings 178

## 7 *Web workers and drag and drop* 185

- 7.1 Getting started: Building an app that integrates Drag and Drop and Web Workers 187
- 7.2 Implementing drag and drop in JavaScript 190
  - The HTML5 Drag-and-Drop API* 191
  - *Using the dataTransfer object to pass data with drag-and-drop events* 193
  - *Building the object to transfer data during drag and drop* 194
- 7.3 HTML5 Web Workers 198
  - The basics: sending work to another thread* 199
  - *Integrating Web Workers into a JavaScript library* 203
- 7.4 Summary 208
- 7.5 The complete code listings 208

## 8 *Websockets* 214

- 8.1 HTTP and TCP—a quick primer 216
  - An HTTP overview* 216
  - *TCP communications in a nutshell* 217
- 8.2 Building a Websockets chat application 219
  - Separating interface logic from Websockets communications* 222
  - Implementing Websockets in JavaScript* 224
  - *Opening a Websockets server connection* 227
  - *Sending messages* 230
  - Receiving messages* 231
- 8.3 Using Node.js as a TCP server 234
- 8.4 Summary 241
- 8.5 The complete code listings 241

## 9 *Local storage and state management* 248

- 9.1 A LocalStorage example application 249
- 9.2 Structuring a JavaScript library to maintain state 253
  - Creating an application outline that supports local storage of objects* 253
  - *Building UI elements that can be stored locally* 255
- 9.3 Using the LocalStorage API 257
  - Adding and removing items to and from LocalStorage the not-so-easy way* 257
  - *Adding and removing items the easy way* 258
  - *Moving data from LocalStorage to the page* 259
  - *Deleting items from LocalStorage* 260
  - Clearing all items from LocalStorage* 260
  - *Using the LocalStorage storage event to detect changes* 261
- 9.4 Adding UI elements to complete the application 262
- 9.5 Other uses for LocalStorage 264
  - Using LocalStorage as a proxy for server data* 264
  - Using LocalStorage to save images* 266
- 9.6 Summary 267
- 9.7 The complete code listings 267

## 10 *Offline web applications* 273

- 10.1 Building an offline HTML5 application 274
  - Creating the basic site structure* 277
  - *Creating the offline JavaScript library* 279
- 10.2 The manifest file 280
  - Adding the application manifest to the sample project* 281
  - Exploring manifest sections* 281
- 10.3 Offline feature detection and event binding 285
- 10.4 The ApplicationCache object 287
- 10.5 Adding state management and displaying connected status 290
- 10.6 Building the server side of an offline application 295
- 10.7 Summary 300
- 10.8 The complete code listings 300

- appendix A A JavaScript overview* 307
- appendix B Using ASP.NET MVC* 346
- appendix C Installing IIS Express 7.5* 372
- index* 377