

brief contents

PART 1 MEET JQUERY UI1

- 1 ■ Introducing jQuery UI 3
- 2 ■ Enhancing UIs with widgets 18

PART 2 JQUERY UI CORE41

- 3 ■ Building complex web forms with jQuery UI 43
- 4 ■ Enhancing interfaces with layout and utility widgets 77
- 5 ■ Adding interaction to your interfaces 107
- 6 ■ Creating rich animations with effects 135
- 7 ■ Theming and styling applications with jQuery UI 162

PART 3 CUSTOMIZATION AND ADVANCED USAGE183

- 8 ■ Using the widget factory to build stateful plugins 185
- 9 ■ Extending widgets with the widget factory 213
- 10 ■ Preparing your application for production 238
- 11 ■ Building a flight-search application 259
- 12 ■ Under the hood of jQuery UI 287

contents

foreword xiii
preface xv
acknowledgments xvi
about this book xviii
about the cover illustration xxi

PART 1 MEET JQUERY UI.....1

1 *Introducing jQuery UI* 3

- 1.1 What is in jQuery UI? 4
- 1.2 The benefits of using jQuery UI 6
 - Cohesive and consistent APIs* 6 ▪ *Comprehensive browser support* 7
 - Open source and free to use* 7 ▪ *Thorough documentation* 7
 - Powerful theming mechanism* 7 ▪ *Emphasis on accessibility* 8
 - Stable and maintenance friendly* 9
- 1.3 The limitations of jQuery UI 9
 - Lack of widgets* 9 ▪ *jQuery UI and mobile devices* 10
- 1.4 Getting started with the library 11
 - Versions of the library* 11 ▪ *Downloading from the jQuery UI website* 11 ▪ *Downloading from CDNs* 12
- 1.5 The first example 12

- 1.6 Using an online testing tool 15
- 1.7 Summary 17

2 *Enhancing UIs with widgets* 18

- 2.1 Creating widgets 19
- 2.2 Customizing widgets with options 20
- 2.3 Modifying widgets with methods 24
 - Invoking methods* 24 ▪ *Using option() to modify widgets* 27
 - Using dialogs to edit lists* 28
- 2.4 Responding to widget changes with events 31
 - Subscribing to widget events* 32 ▪ *Event handlers vs. callbacks* 34 ▪ *Event parameters* 35
- 2.5 Summary 39

PART 2 JQUERY UI CORE.....41

3 *Building complex web forms with jQuery UI* 43

- 3.1 The challenges of building modern web forms 44
- 3.2 Autocomplete: suggesting input options to users 46
 - Using local data* 47 ▪ *Loading from a remote source* 49
 - Using autocomplete with third-party services and APIs* 52
- 3.3 Button: enhancing native buttons, inputs, and links 55
- 3.4 Selectmenu: enhancing native <select> elements 59
- 3.5 Datepicker: selecting dates from a pop-up calendar 62
 - Parsing and formatting dates* 64 ▪ *Handling date globalization* 67
- 3.6 Spinner: enhancing native <input> elements to collect numeric data 69
- 3.7 Completing the appointment form 71
- 3.8 HTML5 elements vs. jQuery UI widgets 74
- 3.9 Summary 76

4 *Enhancing interfaces with layout and utility widgets* 77

- 4.1 Accordion: creating toggleable content panels 78
 - Configuring the accordion widget* 79 ▪ *Adding and removing panels* 81

- 4.2 Tabs: toggling between content areas 82
 - Loading remote content 83* ▪ *Loading movie information in a tabs widget 83*
- 4.3 Menu: creating web menus with semantic markup 88
- 4.4 Dialog: displaying content in a pop-up container 91
- 4.5 Progressbar: displaying the progress of a task 94
- 4.6 Slider: selecting a value using moveable handles 97
 - Building range sliders 98* ▪ *Adding a font size range 99*
- 4.7 Tooltip: enhancing native tooltips with a customizable control 101
 - Using custom tooltip content 103* ▪ *Displaying a preview in a tooltip 104*
- 4.8 Summary 106

5 *Adding interaction to your interfaces 107*

- 5.1 Draggable: allowing users to move elements 108
- 5.2 Droppable: creating containers that accept draggables 110
 - Building a drag-and-drop game 110* ▪ *Building a shopping cart 114*
- 5.3 Sortable: rearranging elements in a list 118
 - Building connected lists 121* ▪ *Building a fruit and vegetable sorting game 121*
- 5.4 Resizable: allowing users to change the size of elements 125
 - Using custom resize handles 126* ▪ *Building an appointment scheduler 127*
- 5.5 Selectable: allowing users to select elements from a group 130
- 5.6 Creating multidevice interactions: the importance of touch 132
 - Why doesn't jQuery UI support touch events? 132* ▪ *Introducing jQuery UI Touch Punch 133*
- 5.7 Summary 134

6 *Creating rich animations with effects 135*

- 6.1 Using effects and the effect() method 136
 - Customizing effects with easings 138* ▪ *Making visual associations with the transfer effect 140*

- 6.2 Animating visibility changes 142
 - Building form validation messages* 142
 - *Building portlets with jQuery UI* 144
- 6.3 Using effects with the jQuery UI widgets 147
 - The show and hide options* 147
 - *Showing a message in a dialog* 148
- 6.4 Animating CSS class name changes 151
 - Enhancing addClass(), removeClass(), and toggleClass()* 151
 - Building an off-canvas navigation menu for mobile* 153
- 6.5 Effects vs. CSS3 animations and transitions 156
 - CSS3 transitions vs. the jQuery UI class name methods* 156
 - *CSS animations vs. effects* 158
- 6.6 Summary 161

7 *Theming and styling applications with jQuery UI* 162

- 7.1 Using built-in and custom themes 163
- 7.2 Using the jQuery UI CSS framework to customize applications 166
 - Styling widget containers* 167
 - *Styling interaction states* 168
 - Styling interaction cues* 169
 - *Building a styled confirmation dialog* 172
- 7.3 Styling with widget class names 174
 - Building vertical tabs* 176
 - *Building a mobile-friendly datepicker* 178
 - *Adding arrows to tooltips with CSS* 179
- 7.4 Summary 181

PART 3 CUSTOMIZATION AND ADVANCED USAGE183

8 *Using the widget factory to build stateful plugins* 185

- 8.1 Building a widget 186
 - Constructing widgets with \$.widget()* 187
 - *Choosing a markup structure* 189
 - *Overriding _create() to initialize widgets* 190
 - *Making widgets themeable* 193
 - *Listening for events with _on()* 195
- 8.2 Customizing widgets with options, methods, and events 196
 - Making widgets configurable with options* 197
 - *Changing the widget's state with methods* 200
 - *Triggering widget events with _trigger()* 202

- 8.3 Enabling, disabling, and destroying widgets 205
 - Enabling and disabling a widget* 205
 - *Undoing a widget's effects with `_destroy()`* 207
- 8.4 Summary 212

9 *Extending widgets with the widget factory* 213

- 9.1 Building widget extensions 214
 - Changing existing and adding new options to a widget* 214
 - Redefining widgets with the widget factory* 218
 - *Extending a custom widget* 220
- 9.2 Customizing widgets with extension points 225
 - Using undocumented extension points* 227
 - *Adding your own extension points* 230
- 9.3 Extending the datepicker widget 231
 - Building a mobile-friendly datepicker extension* 234
- 9.4 Summary 236

10 *Preparing your application for production* 238

- 10.1 The problem with third-party CDNs 239
- 10.2 Downloading jQuery UI from Download Builder 241
- 10.3 Managing JavaScript dependencies with AMD 243
 - Setting up RequireJS for development* 245
 - *Loading jQuery UI components with RequireJS* 246
- 10.4 Building your application's assets with the optimizer 249
 - Optimizing JavaScript assets* 249
 - *Optimizing CSS dependencies* 251
- 10.5 Supporting AMD in custom widgets 254
- 10.6 Summary 257

11 *Building a flight-search application* 259

- 11.1 Structuring your application 260
- 11.2 Collecting user input 261
 - Building an airport code autocomplete* 263
 - *Polyfilling HTML5 inputs with jQuery UI* 265
 - *Validating user input with HTML5* 268
- 11.3 Connecting to a RESTful API 271
 - Looking up flights with `$.ajax()`* 271
 - *Parsing XML with jQuery* 272

- 11.4 Displaying the results on the screen 274
 - Storing and resolving templates with RequireJS* 276
 - Showing a processing indicator while data loads* 277
- 11.5 Adding a responsive design 279
- 11.6 Preparing the application for production 283
- 11.7 Getting the optimal performance with almond 284
- 11.8 Summary 286

12 *Under the hood of jQuery UI* 287

- 12.1 Positioning elements with the position utility 287
 - Building a UI walkthrough with the position utility and dialog widget* 290
 - Handling collisions elegantly* 293
 - Controlling the collision detection* 295
- 12.2 Using the utility functionality in jQuery UI Core 297
 - Generating unique ids* 297
 - Using key code constants* 299
- 12.3 Accessing and managing widget instances 300
 - Detecting widget instances with :data()* 301
- 12.4 Advanced widget prototype methods and properties 302
 - A widget's prototype chain explained* 303
 - Using a widget's default element to streamline initialization* 305
 - Supporting embedded-window use in widgets* 307
 - Displaying elements with _show() and _hide()* 308
 - Customizing options on the fly* 310
- 12.5 Using autoinitialization to remove boilerplate code 311
 - How jQuery Mobile's autoinitialization works* 312
 - jQuery Mobile's widget extension* 314
 - Autoinitializing jQuery UI widgets* 315
- 12.6 Summary 318

- appendix A Learning jQuery* 319
- appendix B How jQuery UI tests jQuery UI* 325
- appendix C Using jQuery UI with Backbone* 332
- appendix D Creating decimal, currency, and time pickers with Globalize* 337
- appendix E Contributing to jQuery UI* 343
- appendix F Polyfilling HTML5 with jQuery UI* 348
- index* 355