

brief contents

PART 1	INTRODUCING XCODE AND SWIFT	1
1	▪ Your first iOS application	3
2	▪ Introduction to Swift playgrounds	29
3	▪ Swift objects	55
PART 2	BUILDING YOUR INTERFACE	81
4	▪ View controllers, views, and outlets	83
5	▪ User interaction	105
6	▪ Adaptive layout	133
7	▪ More adaptive layout	167
8	▪ Keyboard notifications, animation, and scrolling	199
PART 3	BUILDING YOUR APP	225
9	▪ Tables and navigation	227
10	▪ Collections, searching, sorting, and tab bars	263
11	▪ Local data persistence	297
12	▪ Data persistence in iCloud	337
13	▪ Graphics and media	371
14	▪ Networking	409
15	▪ Debugging and testing	439

PART 4	FINALIZING YOUR APP	479
16	▪ Distributing your app	481
17	▪ What's next?	513

contents

preface xv
acknowledgments xviii
about this book xix
about the author xxiii
about the cover illustration xxiv

PART 1 INTRODUCING XCODE AND SWIFT 1

1 Your first iOS application 3

1.1 Exploring iOS SDK 3

1.2 Creating an Xcode project 5

Templates 6 ▪ *Project options* 9

1.3 Exploring the Xcode interface 10

Toolbar area 11 ▪ *Utility area* 12 ▪ *Navigator area* 12

Editor area 13 ▪ *Debug area* 15

1.4 Editing your app's interface 15

Storyboards and nibs 15 ▪ *View controllers and views* 16

Interface Builder 16 ▪ *Object Library* 17 ▪ *Document*

Outline 18 ▪ *Inspectors* 19

1.5 Running your app 22

Running your app on a device 22 ▪ *Running your app in the simulator* 22 ▪ *Running your app* 23 ▪ *Simulator features* 23

- 1.6 Peeking at a completed app 24
 - Checking out a repository in Xcode* 24
 - Peeking at the completed app's storyboard* 24
 - Tweaking the code* 25
- 1.7 Summary 26

2 Introduction to Swift playgrounds 29

- 2.1 Xcode playground 30
 - Results sidebar* 31
 - Automatic compiling* 32
 - Console* 32
- 2.2 Type safety and type inference 33
 - Converting numeric types* 35
 - Concatenating strings* 36
- 2.3 Collections 36
 - Arrays* 36
 - Sets* 37
 - Dictionaries* 38
- 2.4 Control Flow 38
 - for-in* 39
 - switch statement* 40
- 2.5 Functions 41
 - Modifying external parameter names* 41
 - Omitting external parameter names* 42
 - Default parameter names* 42
- 2.6 Optionals 42
 - Declaring an optional* 44
 - Unwrapping an optional* 44
 - Optional chaining* 48
 - Final comments on optionals* 49
- 2.7 Tuples 49
 - Tuples as return values* 50
 - Tuple magic* 50
- 2.8 Higher-order functions 51
 - map* 51
 - Closures* 52
 - filter* 53
 - reduce* 53
 - sorted* 54
- 2.9 Summary 54

3 Swift objects 55

- 3.1 Classes 56
 - Defining a class* 56
 - Properties* 57
 - Initializers* 58
 - Methods* 59
 - Computed properties* 63
 - Class inheritance* 65
 - Protocols* 68
- 3.2 Structures 71
 - Structures vs. classes* 72

- 3.3 Extensions 76
 - Extensions of your type* 76 • *Extensions of their type* 77
 - Operator overloading* 78 • *Generics* 79
- 3.4 Summary 80

PART 2 BUILDING YOUR INTERFACE 81

4 *View controllers, views, and outlets* 83

- 4.1 View hierarchy 84
- 4.2 Model-view-controller 85
- 4.3 View controller 86
 - Creating a custom view controller* 87 • *Customizing a UIViewController subclass* 88 • *Initial view controller* 90
- 4.4 Managing views 94
 - Managing views in code* 94 • *Managing views in Interface Builder* 97
- 4.5 Summary 104

5 *User interaction* 105

- 5.1 Controls 106
 - Buttons* 106 • *Text field* 109 • *Other controls* 114
- 5.2 Touching views 115
 - Hit testing* 116 • *Overriding touch methods* 116
 - The responder chain* 117
- 5.3 Gesture recognizers 119
 - Pan gesture* 121 • *Pinch gesture* 125 • *Rotate gesture* 126 • *Simultaneous gesture recognizers* 126
 - Tap gesture in code* 129
- 5.4 Summary 131

6 *Adaptive layout* 133

- 6.1 The problems 134
- 6.2 Auto layout 135
 - Auto layout tips* 137 • *Auto layout in Interface Builder* 137
 - Auto layout in code* 153

- 6.3 Autoresizing 157
 - Autoresizing in code* 158 ▪ *Autoresizing in Interface Builder* 160 ▪ *Autoresizing considerations* 160
 - 6.4 Manual adaptive layout 161
 - Receiving transition events* 161 ▪ *Receiving layout events* 163
 - 6.5 Choosing an approach 165
 - 6.6 Summary 166
- 7 More adaptive layout 167**
- 7.1 Size classes 167
 - Size classes in code* 170 ▪ *Size classes in Interface Builder* 176
 - 7.2 Stack views 185
 - The problem with auto layout* 185 ▪ *Stack view properties* 187
 - Simple stack view in Interface Builder* 189 ▪ *Nested stack views in Interface Builder* 191 ▪ *Adding or removing views from a stack view* 194 ▪ *Stack views in code* 195
 - 7.3 Summary 197
- 8 Keyboard notifications, animation, and scrolling 199**
- 8.1 The problem with the keyboard 200
 - 8.2 Dismissing the keyboard 201
 - Dismissing the keyboard by resigning the first responder* 202
 - Detecting when to dismiss the keyboard* 202
 - 8.3 Observing keyboard notifications 205
 - What is a notification?* 205 ▪ *Observing a keyboard frame change notification* 206 ▪ *Unregistering a notification* 207
 - Extracting keyboard information from the notification* 208
 - Getting a reference to the first responder* 209 ▪ *Calculating the offset to animate* 210
 - 8.4 Animating views 211
 - Animating the view from under the keyboard* 212 ▪ *Diving deeper into animating views with a sample bar chart* 213
 - 8.5 Scroll views 218
 - Scroll view with form content and keyboard* 218 ▪ *Diving deeper into scroll views with image content* 222
 - 8.6 Summary 223

PART 3 BUILDING YOUR APP 225

9 Tables and navigation 227

- 9.1 Displaying data in table views 228
 - Setting up a table view controller in the storyboard* 229
 - Displaying data in the table view* 234
- 9.2 Adding a row 242
 - Embedding a navigation controller* 243
 - Creating a segue* 245
 - Embedding second navigation controller* 248
 - Communicating with the books scene using your own delegate* 249
 - Adding data to the table* 252
- 9.3 Editing a row 255
 - Creating a segue from a row* 255
 - Passing in the book object to edit* 257
 - Removing the view controller* 258
 - Updating the book object* 259
- 9.4 Using large titles 259
- 9.5 Deleting a row 260
- 9.6 Summary 262

10 Collections, searching, sorting, and tab bars 263

- 10.1 Sorting the data 264
 - Creating a sort method to sort the books array* 264
 - Changing sort order* 265
- 10.2 Searching the data 270
 - Creating a search controller* 271
 - Adding the search controller to the view controller* 272
 - Filtering the data* 273
 - Removing and updating rows with filtered data* 275
- 10.3 Displaying data in collection views 278
 - Creating custom collection cells* 280
 - Displaying data in a custom collection view cell* 281
 - Implementing a flow layout* 283
 - Adding a search bar to the collection view* 283
 - Creating a second section* 285
 - Implementing the flow layout delegate* 286
- 10.4 Creating sections with a tab bar controller 288
 - Sharing data between tabs* 290
- 10.5 Summary 295

11 *Local data persistence* 297

- 11.1 Preserving user preferences and state 298
 - Preserving and restoring state* 298 ▪ *Preserving user preferences on the device* 300
- 11.2 Storing data locally 302
 - Storage setup* 302 ▪ *Structured data files* 306 ▪ *Archiving objects* 312 ▪ *SQLite* 315 ▪ *Core Data* 322
- 11.3 Summary 335

12 *Data persistence in iCloud* 337

- 12.1 Setting up your app for iCloud 338
- 12.2 Persisting data with ubiquitous key-value store 339
- 12.3 Storing data using CloudKit 342
 - Updating the model for CloudKit* 344 ▪ *Adding a book record to CloudKit* 346 ▪ *Updating a book record in CloudKit* 352 ▪ *Loading book records in CloudKit* 353 ▪ *Deleting a book record in CloudKit* 355 ▪ *Managing CloudKit errors* 356 ▪ *Refreshing CloudKit data* 360 ▪ *Subscribing to changes* 362
- 12.4 Summary 369

13 *Graphics and media* 371

- 13.1 Adding images to your app with an asset catalog 372
 - Adding image sets* 373 ▪ *Adding app icons* 377
- 13.2 Displaying a launch screen 379
- 13.3 Drawing with Core Graphics 381
 - Overriding the draw method* 382 ▪ *Describing a path* 382 ▪ *Drawing into the graphics context* 383 ▪ *Saving and restoring graphics state* 384 ▪ *Drawing paths with UIBezierPath drawing methods* 384 ▪ *Rendering views in Interface Builder* 385 ▪ *Creating a star-rating view* 386
- 13.4 Drawing with Core Animation 389
- 13.5 Using the camera 391
 - Taking photos with the image picker controller* 392 ▪ *Selecting photos from photo library with the image picker controller* 397 ▪ *Taking photos with AVFoundation* 398
- 13.6 Playing sounds 405
- 13.7 Summary 407

14 Networking 409

- 14.1 Using a web service 410
- 14.2 Setting up a books service 411
- 14.3 Communicating with the web service 413
- 14.4 Creating a URL Session 414
 - URLSessionConfiguration* 414 ▪ *URLSession* 414
- 14.5 Setting up the URL request 416
- 14.6 Requesting data from a web service 418
- 14.7 Examining the data 418
- 14.8 Parsing JSON data with JSONSerialization 420
- 14.9 Parsing JSON data with JSONDecoder 421
- 14.10 Parsing JSON data with SwiftyJSON 424
 - Integrating SwiftyJSON with Carthage* 426 ▪ *Using SwiftyJSON* 429
- 14.11 Downloading data from a web service 431
 - Accessing insecure domains* 433
- 14.12 Displaying the network activity indicator 436
- 14.13 Cancelling a task 436
- 14.14 Summary 437

15 Debugging and testing 439

- 15.1 The setup 440
- 15.2 Debugging mode 440
- 15.3 Debugging crash logs in the console 443
 - Solving a crash caused by an outlet* 445 ▪ *Solving a crash caused by an action* 447
- 15.4 Examining variables and breakpoints 448
 - Examining a variable with print* 449 ▪ *Pausing your app with a breakpoint* 450 ▪ *Examining a variable with the variables view* 452 ▪ *Controlling the app's execution using the debug bar* 452 ▪ *Examining a variable with Quick Look* 454 ▪ *Examining a variable with print description* 455 ▪ *Examining a variable with LLDB* 456 ▪ *Examining a variable with data tips* 457 ▪ *Solving the save problem* 458 ▪ *Examining a variable in summary* 458
- 15.5 Debugging playback with gauges and instruments 459
 - Debugging playback with debug gauges* 459 ▪ *Debugging playback with instruments* 461 ▪ *Solving the playback problem* 463

- 15.6 Debugging the user interface 463
 - Debugging the user interface with the Debug View Hierarchy* 465
 - Debugging the user interface with runtime issues* 467
 - Solving the user interface problem* 468
- 15.7 Testing your app 469
 - Testing for functionality* 470 ▪ *Testing for performance* 473
 - Testing your user interface* 475
- 15.8 Summary 478

PART 4 FINALIZING YOUR APP 479

16 *Distributing your app* 481

- 16.1 Joining the Apple Developer Program 482
 - Signing into Xcode* 482 ▪ *Code signing your app* 483
- 16.2 Setting up an app in iTunes Connect 490
- 16.3 Uploading your build to iTunes Connect 493
- 16.4 Distributing your app to beta testers 495
 - Distributing to beta testers manually* 496 ▪ *Distributing to beta testers with TestFlight* 497
- 16.5 Distributing your app to the App Store 505
- 16.6 Summary 511

17 *What's next?* 513

- 17.1 Further learning 513
- 17.2 One more thing! 515

- appendix A Project settings* 517
- appendix B Swift syntax cheat sheets* 525
- index* 529