

Table of Contents

Headings formatted in *bold-italic* have changed since the last version.

- [Preface](#)
 - Welcome to the Book! xliii
 - The Book’s Structure xliii
 - ***The Trails*** ***xliv***
 - About the Updates 1
 - ***What’s New in Version 8.3?*** ***1***
 - Warescription 1
 - About the APK Edition li
 - Book Bug Bounty lii
 - Source Code and Its License liii
 - ***Creative Commons and the Four-to-Free (42F) Guarantee*** ***liv***
 - Acknowledgments liv
- [Key Android Concepts](#)
 - Android Applications 1
 - Android Devices 7
 - Don’t Be Scared 11
- [Choosing Your Development Toolchain](#)
 - Android Studio 13
 - Eclipse 13
 - IntelliJ IDEA 14
 - Command-Line Builds via Gradle for Android 14
 - Yet Other Alternatives 14
 - IDEs... And This Book 15
 - What We Are Not Covering 15
- [Tutorial #1 - Installing the Tools](#)
 - But First, Some Notes About Android’s Emulator 17
 - Step #1: Checking Your Hardware 18
 - Step #2: Setting Up Java and 32-Bit Linux Support 19
 - Step #3: Install Android Studio 19
 - Step #4: Install the SDKs and Add-Ons 21
 - In Our Next Episode... 31
- [Android and Projects](#)
 - Common Concepts 33
 - ***Projects and Android Studio*** ***34***
 - Starter Project Generators 41

- [Tutorial #2 - Creating a Stub Project](#)
 - About Our Tutorial Project 43
 - About the Rest of the Tutorials 44
 - About Our Tools 44
 - Step #1: Creating the Project 44
 - Step #2: Get Ready for the x86 Emulator 47
 - **Step #3: Set Up the AVD** **48**
 - Step #4: Set Up the Device 54
 - Step #5: Running the Project 59
 - In Our Next Episode... 60
- [Getting Around Android Studio](#)
 - Navigating The Project Explorer 61
 - Running Projects 64
 - Viewing Output 66
 - Accessing Android Tools 68
 - Android Studio and Release Channels 71
 - Visit the Trails! 72
- [Contents of Android Projects](#)
 - What You Get, In General 73
 - The Contents of an Android Studio Project 75
 - **The Contents of an Eclipse-Style Project** **77**
 - What You Get Out Of It 78
- [Introducing Gradle and the Manifest](#)
 - Gradle: The Big Questions 79
 - Obtaining Gradle 82
 - **Versions of Gradle and the Android Plugin for Gradle** **84**
 - Gradle Environment Variables 85
 - Examining the Gradle Files 85
 - Introducing the Manifest 88
 - Things In Common Between the Manifest and Gradle 88
 - Other Gradle Items of Note 92
 - Where's the GUI? 94
 - **The Rest of the Manifest** **94**
 - Learning More About Gradle 96
 - Visit the Trails! 97
- [Tutorial #3 - Manifest Changes](#)
 - Some Notes About Relative Paths 99
 - Step #1: Supporting Screens 99
 - Step #2: Blocking Backups 100
 - Step #3: Ignoring Lint 100
 - In Our Next Episode... 102

- [Some Words About Resources](#)
 - **String Theory** 104
 - Got the Picture? 108
 - Dimensions 115
 - The Resource That Shall Not Be Named... Yet 116
- [Tutorial #4 - Adjusting Our Resources](#)
 - Step #1: Changing the Name 117
 - **Step #2: Changing the Icon** 118
 - Step #3: Running the Result 121
 - In Our Next Episode... 122
- [The Theory of Widgets](#)
 - What Are Widgets? 123
 - Size, Margins, and Padding 125
 - What Are Containers? 125
 - The Absolute Positioning Anti-Pattern 126
 - The Theme of This Section: Themes 127
- [The Android User Interface](#)
 - **The Activity** 131
 - Dissecting the Activity 132
 - Using XML-Based Layouts 133
- [Basic Widgets](#)
 - Common Concepts 137
 - Introducing the Graphical Layout Editor 139
 - Assigning Labels 149
 - A Commanding Button 155
 - Fleeting Images 159
 - Fields of Green. Or Other Colors. 165
 - More Common Concepts 169
 - Visit the Trails! 172
- [Debugging Your App](#)
 - Get Thee To a Stack Trace 174
 - The Case of the Confounding Class Cast 175
 - Point Break 176
- [The Classic Container Classes](#)
 - Introducing the Sampler App 177
 - RTL and Your Layouts 178
 - **LinearLayout and the Box Model** 179
 - All Things Are Relative 206
 - Tabula Rasa 217
 - Hey, What About ConstraintLayout? 223
 - Turning Back to RTL 224

- [Other Common Widgets and Containers](#)
 - Just a Box to Check 227
 - Don't Like Checkboxes? How About Toggles or Switches? 232
 - Turn the Radio Up 237
 - Scrollwork 240
 - Making Progress with ProgressBars 244
 - Framing the Scene 244
 - Visit the Trails! 245
- [Tutorial #5 - Making Progress](#)
 - Step #1: Creating a New Layout Resource 247
 - **Step #2: Defining the UI** **248**
 - In Our Next Episode... 252
- [GUI Building, Continued](#)
 - Making Your Selection 253
 - Including Includes 254
 - Preview of Coming Attractions 256
- [AdapterViews and Adapters](#)
 - Adapting to the Circumstances 257
 - Lists of Naughty and Nice 259
 - Clicks versus Selections 261
 - Spin Control 265
 - Grid Your Lions (Or Something Like That...) 268
 - Fields: Now With 35% Less Typing! 272
 - Customizing the Adapter 276
 - Visit the Trails! 285
- [The WebView Widget](#)
 - Role of WebView 287
 - Daddy, Where Do WebViews Come From? 288
 - Adding the Widget 288
 - Loading Content Via a URL 289
 - Links and Redirects 291
 - Supporting JavaScript 291
 - Alternatives for Loading Content 292
 - Listening for Events 293
 - Addressing the Link/Redirect Behavior 296
 - Visit the Trails! 298
- [Defining and Using Styles](#)
 - Styles: DIY DRY 299
 - Elements of Style 301
 - Themes: Would a Style By Any Other Name... 304
 - What Happens If You Have No Theme 305

◦ Android Studio's Theme Editor	305
• JARs and Library Projects	
◦ The Dalvik VM, and a Bit of ART	312
◦ Getting the Library	312
◦ The Outer Limits	314
◦ JAR Dependency Management	315
◦ OK, So What is a Library Project?	315
◦ Using a Library Project	315
◦ Library Projects: What You Get	316
◦ The Android Support Package	317
• Tutorial #6 - Adding a Library	
◦ Step #1: Getting Rid of Existing Cruft	321
◦ Step #2: Requesting New Dependencies	322
◦ In Our Next Episode...	324
• Introducing ConstraintLayout	
◦ Why Another Container?	325
◦ Comparing with the Classics	327
◦ Getting ConstraintLayout	328
◦ Using Widgets and Containers from Libraries	329
◦ Using a ConstraintLayout	330
◦ Converting Existing Layouts	343
◦ Visit the Trails!	346
• The Action Bar	
◦ Bar Hopping	347
◦ Yet Another History Lesson	353
◦ Your Action Bar Options	354
◦ Setting the Target	356
◦ Defining the Resource	357
◦ Applying the Resource	360
◦ Responding to Events	360
◦ The Rest of the Sample Activity	361
◦ MENU Key, We Hardly Knew Ye	367
◦ Action Bars, Live in Living Color!	368
◦ Visit the Trails!	378
• Vector Drawables	
◦ Getting the Artwork	379
◦ VectorDrawableCompat	384
◦ Other VectorDrawable Backports	392
• Tutorial #7 - Setting Up the Action Bar	
◦ Step #1: Adding Some Icons	393
◦ Step #2: Defining Some Options	396

◦ Step #3: Loading and Responding to Our Options	399
◦ Step #4: Supporting Older Devices	400
◦ Step #5: Trying It Out	401
◦ In Our Next Episode...	404
• Android's Process Model	
◦ When Processes Are Created	405
◦ BACK, HOME, and Your Process	406
◦ Termination	407
◦ Foreground Means "I Love You"	408
◦ You and Your Heap	408
• Activities and Their Lifecycles	
◦ Creating Your Second (and Third and...) Activity	410
◦ Warning! Contains Explicit Intents!	414
◦ Using Implicit Intents	416
◦ Extra! Extra!	422
◦ Pondering Parcelable	423
◦ Asynchronicity and Results	425
◦ Schroedinger's Activity	425
◦ Life, Death, and Your Activity	426
◦ When Activities Die	428
◦ Walking Through the Lifecycle	429
◦ Recycling Activities	432
◦ Application: Transcending the Activity	433
◦ The Case of the Invisible Activity	434
• Tutorial #8 - Setting Up An Activity	
◦ Step #1: Creating the Stub Activity Class and Manifest Entry	437
◦ Step #2: Launching Our Activity	439
◦ In Our Next Episode...	440
• The Tactics of Fragments	
◦ The Six Questions	441
◦ Where You Get Your Fragments From	444
◦ Your First Fragment	444
◦ The Fragment Lifecycle Methods	448
◦ Your First Dynamic Fragment	449
◦ Fragments and the Action Bar	452
◦ Fragments Within Fragments: Just Say "Maybe"	454
◦ Fragments and Multiple Activities	454
◦ Support Fragments and Lifecycle Changes	455
• Tutorial #9 - Starting Our Fragments	
◦ Step #1: Create a SimpleContentFragment	457
◦ Step #2: Examining SimpleContentFragment	459

◦ In Our Next Episode...	460
• Swiping with ViewPager	
◦ Pieces of a Pager	461
◦ Paging Fragments	462
◦ Paging Other Stuff	467
◦ Indicators	468
◦ Revisiting the Containers Sampler	471
• Tutorial #10 - Rigging Up a ViewPager	
◦ Step #1: Add a ViewPager to the Layout	477
◦ Step #2: Creating a ContentsAdapter	478
◦ Step #3: Setting Up the ViewPager	479
◦ In Our Next Episode...	482
• Resource Sets and Configurations	
◦ What's a Configuration? And How Do They Change?	483
◦ Configurations and Resource Sets	484
◦ Screen Size and Orientation	485
◦ Coping with Complexity	488
◦ Choosing The Right Resource	489
◦ API-Versioned Resources	493
◦ Default Change Behavior	495
◦ State Saving Scenarios	497
◦ Your Options for Configuration Changes	498
◦ Blocking Rotations	512
◦ And Now, a Word From the Android Project View	513
◦ Configuration Challenges	514
• Material Design Basics	
◦ Your App, in Technicolor!	519
• Dealing with Threads	
◦ The Main Application Thread	525
◦ Getting to the Background	527
◦ Asyncing Feeling	527
◦ Alternatives to AsyncTask	536
◦ And Now, The Caveats	538
◦ Event Buses	539
◦ Visit the Trails!	549
• Requesting Permissions	
◦ Frequently-Asked Questions About Permissions	552
◦ Characteristics of Permissions	557
◦ New Permissions in Old Applications	560
◦ Android 6.0+ Runtime Permission System	561
◦ A Simple Runtime Permission Abstraction	573

- [Tutorial: Runtime Permission Support](#)
 - Step #0: Install the Android 6.0 SDK 578
 - Step #1: Import and Review the Starter Project 579
 - Step #2: Update Gradle for Android 6.0+ 585
 - Step #3: Review the Planned UX 587
 - Step #4: Detect the First Run 588
 - Step #5: On First Run, Ask For Permissions 589
 - Step #6: Check for Permissions Before Taking a Picture 593
 - Step #7: Detect If We Should Show Some Rationale 595
 - Step #8: Add a Rationale UI and Re-Request Permissions 596
 - Step #9: Check for Permissions Before Recording a Video 603
 - Step #10: Detect If We Should Show Some Rationale (Again) 604
 - Step #11: Support Configuration Changes 607
- [Assets, Files, and Data Parsing](#)
 - Packaging Files with Your App 617
 - Files and Android 619
 - Working with Internal Storage 620
 - **Working with External Storage** **629**
 - Multiple User Accounts 636
 - Linux Filesystems: You Sync, You Win 637
 - StrictMode: Avoiding Janky Code 639
 - Files, and Your Development Machine 641
 - XML Parsing Options 644
 - JSON Parsing Options 645
 - Using Files with Implicit Intents 645
 - Visit the Trails! 646
- [Tutorial #11 - Adding Simple Content](#)
 - Step #1: Adding Some Content 647
 - Step #2: Using SimpleContentFragment 648
 - Step #3: Launching Our Activities, For Real This Time 649
 - Step #4: Getting a Bit More Material 650
 - Step #5: Seeing the Results 652
 - In Our Next Episode... 654
- [Tutorial #12 - Displaying the Book](#)
 - Step #1: Adding a Book 655
 - Step #2: Creating a ModelFragment 656
 - Step #3: Defining Our Model 657
 - Step #4: Examining Our Model 659
 - Step #5: Defining Our Event 659
 - Step #6: Loading Our Model 660
 - Step #7: Registering for Events 663

◦ Step #8: Adapting the Content	664
◦ Step #9: Showing the Content When Loaded	665
◦ Step #10: Attaching our ModelFragment	666
◦ Step #11: Showing the Content After a Configuration Change	667
◦ Step #12: Setting Up StrictMode	668
◦ In Our Next Episode...	669
• Using Preferences	
◦ Getting What You Want	671
◦ Stating Your Preference	672
◦ Collecting Preferences with PreferenceFragment	673
◦ Android Studio's Preferences Editor	680
◦ Types of Preferences	681
• Tutorial #13 - Using Some Preferences	
◦ Step #1: Defining the Preference XML Files	687
◦ Step #2: Creating Our Preference Activity	691
◦ Step #3: Adding To Our Action Bar	692
◦ Step #4: Launching the Preference Activity	693
◦ Step #5: Loading the Preferences	695
◦ Step #6: Saving the Last-Read Position	699
◦ Step #7: Restoring the Last-Read Position	700
◦ Step #8: Keeping the Screen On	701
◦ In Our Next Episode...	703
• SQLite Databases	
◦ Introducing SQLite	705
◦ Thinking About Schemas	706
◦ Start with a Helper	706
◦ Getting Data Out	711
◦ The Rest of the CRUD	718
◦ Hey, What About Hibernate?	724
◦ Visit the Trails!	724
• Tutorial #14 - Saving Notes	
◦ Step #1: Adding a DatabaseHelper	725
◦ Step #2: Examining DatabaseHelper	726
◦ Step #3: Creating a NoteFragment	727
◦ Step #4: Examining NoteFragment	728
◦ Step #5: Creating the NoteActivity	728
◦ Step #6: Examining NoteActivity	729
◦ Step #7: Add Notes to the Action Bar	730
◦ Step #8: Defining a NoteLoadedEvent	732
◦ Step #9: Loading a Note from the Database	733
◦ Step #10: Loading the Note Into the Fragment	734

◦ Step #11: Updating the Database	735
◦ Step #12: Saving the Note	736
◦ Step #13: Adding a Delete Action Bar Item	738
◦ Step #14: Closing the NoteFragment When Deleted	741
◦ In Our Next Episode...	746
• Internet Access	
◦ <i>DIY HTTP</i>	747
◦ What About HttpClient?	757
◦ HTTP via DownloadManager	760
◦ Using Third-Party JARs	760
◦ SSL	761
◦ <i>Using HTTP Client Libraries</i>	761
◦ Visit the Trails	787
• Intents, Intent Filters	
◦ What's Your Intent?	789
◦ Stating Your Intent(ions)	791
◦ Responding to Implicit Intents	791
◦ Requesting Implicit Intents	794
◦ ShareActionProvider	798
◦ Practice Safe Content Resolution	801
• Broadcasts and Broadcast Receivers	
◦ The Stopped State	805
◦ <i>Example System Broadcasts</i>	806
◦ The Order of Things	815
◦ Keeping It Local	816
◦ Visit the Trails!	816
• Tutorial #15 - Sharing Your Notes	
◦ Step #1: Adding a ShareActionProvider	817
◦ Step #2: Sharing the Note	818
◦ Step #3: Testing the Result	820
◦ In Our Next Episode...	823
• Services and the Command Pattern	
◦ Why Services?	825
◦ Setting Up a Service	826
◦ Communicating To Services	828
◦ Scenario: The Music Player	830
◦ Communicating From Services	834
◦ Scenario: The Downloader	836
◦ Services and Configuration Changes	842
• Tutorial #16 - Updating the Book	
◦ Step #1: Adding a Stub DownloadCheckService	845

◦ Step #2: Tying the Service Into the Action Bar	846
◦ Step #3: Defining Our Event	848
◦ Step #4: Defining Our JSON	848
◦ Step #5: Defining Our Retrofit Interface	849
◦ Step #6: Retrieving Our JSON Via Retrofit	850
◦ Step #7: Downloading the Update	852
◦ Step #8: Unpacking the Update	853
◦ Step #9: Using the Update	857
◦ In Our Next Episode...	862
• Large-Screen Strategies and Tactics	
◦ Objective: Maximum Gain, Minimum Pain	863
◦ The Fragment Strategy	863
◦ Fragment Example: The List-and-Detail Pattern	872
◦ Other Master-Detail Strategies	886
◦ Showing More Pages	899
◦ Fragment FAQs	905
◦ Screen Size and Density Tactics	906
◦ Other Considerations	909
• Tutorial #17 - Supporting Large Screens	
◦ Step #1: Creating Our Layouts	913
◦ Step #2: Loading Our Sidebar Widgets	917
◦ Step #3: Opening the Sidebar	918
◦ Step #4: Loading Content Into the Sidebar	919
◦ Step #5: Removing Content From the Sidebar	922
• Backwards Compatibility Strategies and Tactics	
◦ Think Forwards, Not Backwards	929
◦ Aim Where You Are Going	931
◦ A Target-Rich Environment	931
◦ Lint: It's Not Just For Belly Buttons	932
◦ A Little Help From Your Friends	933
◦ Avoid the New on the Old	933
◦ Testing	937
◦ Keeping Track of Changes	937
• System Services	
◦ What is a System Service?	939
◦ What System Services Are There?	940
• Google Play Services	
◦ What Is Google Play Services?	943
◦ What Is In the Play Services SDK?	944
◦ Adding Play Services to Your Project	951
• Getting Help	

◦ Questions. Sometimes, With Answers.	963
◦ Heading to the Source	964
◦ Getting Your News Fix	965
• Working with Library Projects	
◦ Prerequisites	967
◦ Creating a Library Project	967
◦ Using a Library Project, Part II	970
◦ Library Projects and the Manifest	970
◦ Limitations of Library Projects	970
• Gradle and Legacy Projects	
◦ Prerequisites and Warnings	973
◦ “Legacy”?	973
◦ Creating Your Gradle Build File	974
◦ Examining the Gradle File	979
• Gradle and Tasks	
◦ Key Build-Related Tasks	981
◦ Results	984
• Gradle and the New Project Structure	
◦ Prerequisites and Warnings	985
◦ Objectives of the New Project Structure	985
◦ Terminology	986
◦ Creating a Project in the New Structure	989
◦ What the New Project Structure Looks Like	990
◦ Configuring the Stock Build Types	993
◦ Adding Build Types	998
◦ Adding Product Flavors and Getting Build Variants	1000
◦ Doing the Splits	1004
◦ Revisiting the Legacy Gradle File	1007
◦ Working with the New Project Structure in Android Studio	1008
◦ Flavors, Build Types, and the Project Structure Dialog	1011
• Gradle and Dependencies	
◦ Prerequisites and Warnings	1013
◦ “Dependencies”?	1013
◦ A Tale of Two Dependencies Closures	1014
◦ Depending Upon a JAR	1014
◦ Depending Upon NDK Binaries	1016
◦ Depending Upon an Android Library Project	1016
◦ Depending Upon Sub-Projects	1018
◦ Depending Upon Artifacts	1019
◦ Creating Android JARs from Gradle	1027
◦ A Property of Transitive (Dependencies)	1028

◦ Dependencies By Build Type	1028
◦ Dependencies By Flavor	1029
◦ <i>Examining Some CWAC Builds</i>	1030
◦ Dependencies and the Project Structure Dialog	1034
• Manifest Merger Rules	
◦ Prerequisites	1035
◦ Manifest Scenarios	1035
◦ Pieces of Manifest Generation	1037
◦ Examining the Merger Results	1039
◦ Viewing Merged Manifests in Android Studio	1039
◦ Merging Elements and Attributes	1040
◦ Employing Placeholders	1050
• Signing Your App	
◦ Prerequisites	1055
◦ Role of Code Signing	1055
◦ What Happens In Debug Mode	1056
◦ Production Signing Keys	1057
• Distribution	
◦ Prerequisites	1065
◦ Get Ready To Go To Market	1065
• Writing a Gradle Plugin	
◦ Prerequisites	1071
◦ Customizing a Gradle Build	1071
◦ Some Use Cases for a Custom Plugin	1072
◦ Writing a Plugin	1073
◦ Distributing the Plugin	1080
◦ Using the Plugin	1082
◦ Creating a Real Plugin	1086
• Code Generation	
◦ Prerequisites	1091
◦ What Drives the Custom Code?	1091
◦ Java as Poetry	1093
◦ Writing a Code Generation Plugin	1094
◦ Using the Generated Code	1105
• Advanced Gradle for Android Tips	
◦ Prerequisites	1107
◦ Gradle, DRY	1107
◦ Automating APK Version Information	1113
◦ Adding to BuildConfig	1116
◦ Down and Dirty with the DSL	1117
• Testing with JUnit4	

◦ Prerequisites	1119
◦ Instrumentation Tests and Unit Tests	1119
◦ Writing JUnit4 Test Cases	1121
◦ Configuring Gradle	1127
◦ Running Your Instrumentation Tests	1128
◦ Testing Android Library Projects	1133
• Testing with Espresso	
◦ Prerequisites	1135
◦ Adding a Shot of Espresso	1136
◦ Writing Tests in Espresso	1138
◦ Stronger Espresso	1144
◦ Opting Out of Analytics	1158
◦ Waiting for the World to Change	1159
• Testing with UI Automator	
◦ Prerequisites	1165
◦ What Is UI Automator?	1165
◦ Why Choose UI Automator Over Alternatives?	1166
◦ Gradle and Android Studio Settings	1166
◦ Creating a Test Case	1167
◦ Running Your Tests	1175
◦ Finding Your Widgets	1175
◦ Using the UI Automator Viewer	1176
• Measuring Test Coverage	
◦ Prerequisites	1179
◦ Who Tests the Testers?	1179
◦ Some Types of Test Coverage	1180
◦ Coverage and Your Instrumentation Tests	1182
• Unit Testing	
◦ Prerequisites	1187
◦ I Thought We Were Already Unit Testing?	1188
◦ Scenario: Clean Architecture	1188
◦ Setting Up Unit Testing	1189
◦ Writing POJO Unit Tests	1192
◦ Running Unit Tests	1196
◦ Mocking Android	1201
• MonkeyRunner and the Test Monkey	
◦ Prerequisites	1223
◦ MonkeyRunner	1223
◦ Monkeying Around	1225
• Java 8 Lambda Expressions	
◦ Prerequisites	1229

◦ The Basic Idea	1230
◦ You Don't (Yet) Know Jack	1231
◦ Using Lambda Expressions	1232
◦ Alternative: Method References	1236
• Rx Basics	
◦ Prerequisites	1239
◦ Life is But a Stream	1239
◦ Action and Reaction	1241
◦ A Rx For What Ails You	1242
◦ Rx and Lambdas	1243
◦ A Simple Stream	1243
◦ Switching to Lambdas	1246
◦ Be Your Own Stream	1246
◦ Removing the AsyncTask	1248
◦ Lambdas and Lifetimes	1253
◦ Streaming from a Resource	1254
◦ Error Handling	1256
◦ Transmogrification	1257
◦ Rx-Enabled Libraries	1258
◦ Further Reading	1260
• Notifications	
◦ Prerequisites	1263
◦ What's a Notification?	1263
◦ Showing a Simple Notification	1266
◦ A Tale of Two NotificationComps	1270
◦ The Activity-Or-Notification Scenario	1271
◦ Big (and Rich) Notifications	1272
◦ Foreground Services	1278
◦ Disabled Notifications	1281
• Advanced Notifications	
◦ Prerequisites	1285
◦ Being a Good Citizen	1285
◦ Wear? There!	1286
◦ Stacking Notifications	1291
◦ Avoiding Wear	1297
◦ Other Wear-Specific Notification Options	1298
◦ Remote Input, On-Device	1312
◦ Notification Groups	1317
◦ Lockscreen Notifications	1318
◦ Priority, and Heads-Up Notifications	1327
◦ Full-Screen Notifications	1329

◦ Progress Notifications	1332
◦ Custom Views	1340
◦ Life After Delete	1344
◦ The Mysterious Case of the Missing Number	1344
◦ Notifications and MessagingStyle	1345
◦ Changes in API Level 23	1350
◦ Sounds and Android 7.0	1351
• Multi-Window Support	
◦ Prerequisites	1353
◦ A History of Windows	1353
◦ What The User Sees	1354
◦ What Your Code Sees	1356
◦ Opting Out	1357
◦ Opting In	1358
◦ Configuring the Layout	1359
◦ Avoiding Stutter	1360
◦ Managing the Background	1362
◦ How Low Can You Go?	1362
◦ Parallel Processing	1363
◦ Split-Screen, HOME, and Your Activity	1365
◦ Split-Screen and Orientations	1366
◦ Forcing Your App Into Multi-Window/Multi-Instance	1366
◦ Supporting Legacy Proprietary Multi-Window	1374
◦ Freeform Multi-Window Mode	1377
• Advanced ConstraintLayout	
◦ Prerequisites	1385
◦ Disclosing Your Bias	1385
◦ Centering Yourself	1389
◦ Keeping Things Proportional	1393
◦ Constraining the ConstraintLayout Size	1394
◦ What If We're GONE?	1395
• Introducing GridLayout	
◦ Prerequisites	1397
◦ Issues with the Classic Containers	1397
◦ The New Contender: GridLayout	1399
◦ GridLayout and the Android Support Package	1399
◦ Our Test App	1400
◦ Replacing the Classics	1403
◦ Implicit Rows and Columns	1409
◦ Row and Column Spans	1410
• The Percent Support Library	

◦ Prerequisites	1417
◦ What Percent Gives Us	1418
◦ Using Percent	1419
◦ About Those Performance Gains	1425
◦ Maintaining Aspect Ratio	1438
◦ Other Problems	1438
• Dialogs and DialogFragments	
◦ Prerequisites	1439
◦ DatePickerDialog and TimePickerDialog	1439
◦ AlertDialog	1445
◦ DialogFragments	1446
◦ DialogFragment: The Other Flavor	1451
◦ Dialogs: Modal, Not Blocking	1451
• Advanced ListView	
◦ Prerequisites	1453
◦ Multiple Row Types, and Self Inflation	1453
◦ Choice Modes and the Activated Style	1460
◦ Custom Mutable Row Contents	1461
◦ From Head To Toe	1466
◦ Enter RecyclerView	1470
• Action Modes	
◦ Prerequisites	1472
◦ A Matter of Context	1472
◦ Manual Action Modes	1473
◦ Multiple-Choice-Modal Action Modes	1478
◦ Long-Click To Initiate an Action Mode	1482
• Other Advanced Action Bar Techniques	
◦ Prerequisites	1489
◦ Action Layouts	1489
◦ Action Views and Action Providers	1490
◦ Searching with SearchView	1491
◦ Floating Action Bars	1497
• Toolbar	
◦ Prerequisites	1501
◦ Basic Toolbar Mechanics	1501
◦ Use Case #1: Split Action Bar	1502
◦ Use Case #2: Contextual Actions	1508
◦ Use Case #3: Replacement Action Bar	1518
• AppCompat: The Official Action Bar Backport	
◦ Prerequisites	1521
◦ Ummmm... Why?	1521

◦ The Basics of Using AppCompatActivity	1524
◦ Other AppCompatActivity Effects	1531
◦ Toolbar and AppCompatActivity	1537
◦ To Material, or Not to Material	1541
• The Android Design Support Library	
◦ Prerequisites	1544
◦ GUIs and the Support Package	1544
◦ <i>Adding the Library... and What Comes With It</i>	1545
◦ <i>Introducing CWAC-CrossPort</i>	1545
◦ <i>Snackbars: Sweeter than Toasts</i>	1546
◦ <i>Absolutely FABulous</i>	1554
◦ <i>Material Tabs with TabLayout</i>	1572
◦ <i>Floating Labels</i>	1584
• Implementing a Navigation Drawer	
◦ Prerequisites	1595
◦ What is a Navigation Drawer?	1595
◦ A Simple Navigation Drawer	1597
◦ Alternative Row Layouts	1603
◦ Additional Considerations	1605
◦ What Should Not Be in the Drawer	1614
◦ Independent Implementations	1615
• RecyclerView	
◦ Prerequisites	1618
◦ AdapterView and its Discontents	1618
◦ Enter RecyclerView	1618
◦ A Trivial List	1619
◦ Divider Options	1627
◦ Handling Click Events	1634
◦ What About Cursors?	1641
◦ Grids	1648
◦ Varying the Items	1652
◦ Mutable Row Contents	1665
◦ Changing the Contents	1693
◦ The Order of Things	1699
◦ <i>Other Bits of Goodness</i>	1707
◦ Animating the Deltas Using DiffUtil	1708
◦ The March of the Libraries	1716
• Advanced RecyclerView	
◦ Prerequisites	1725
◦ <i>RecyclerView as Pager</i>	1725
◦ <i>Declaring a LayoutManager in the Layout</i>	1748

- **Transcript Mode** 1748
- [Advanced Uses of WebView](#)
 - Prerequisites 1753
 - **Friends with Benefits** 1753
 - Navigating the Waters 1768
 - Settings, Preferences, and Options (Oh, My!) 1768
 - Security and Your WebView 1769
 - Chrome Custom Tabs 1772
- [The Input Method Framework](#)
 - Prerequisites 1775
 - Keyboards, Hard and Soft 1775
 - Tailored To Your Needs 1776
 - Tell Android Where It Can Go 1781
 - Fitting In 1783
 - Jane, Stop This Crazy Thing! 1786
- [Fonts](#)
 - Prerequisites 1787
 - Love The One You're With 1787
 - Yeah, But Do We Really Have To Do This in Java? 1791
 - Here a Glyph, There a Glyph 1792
- [Rich Text](#)
 - Prerequisites 1795
 - The Span Concept 1795
 - Loading Rich Text 1797
 - Editing Rich Text 1800
 - Saving Rich Text 1805
 - Manipulating Rich Text 1805
- [Animators](#)
 - Prerequisites 1807
 - ViewPropertyAnimator 1807
 - The Foundation: Value and Object Animators 1812
 - Animating Custom Types 1815
 - Hardware Acceleration 1816
 - The Three-Fragment Problem 1817
- [Legacy Animations](#)
 - Prerequisites 1829
 - It's Not Just For Toons Anymore 1829
 - A Quirky Translation 1830
 - Fading To Black. Or Some Other Color. 1834
 - When It's All Said And Done 1836
 - Loose Fill 1837

◦ Hit The Accelerator	1838
◦ Animate. Set. Match.	1838
◦ Active Animations	1840
• Custom Drawables	
◦ Prerequisites	1841
◦ Where Do These Things Go?	1842
◦ ColorDrawable	1843
◦ AnimationDrawable	1843
◦ StateListDrawable	1847
◦ ColorStateList	1848
◦ LayerDrawable	1850
◦ TransitionDrawable	1851
◦ LevelListDrawable	1852
◦ ScaleDrawable and ClipDrawable	1854
◦ InsetDrawable	1863
◦ ShapeDrawable	1864
◦ BitmapDrawable	1875
◦ Composite Drawables	1882
◦ A Stitch In Time Saves Nine	1886
• Mapping with Maps V2	
◦ Prerequisites	1895
◦ A Brief History of Mapping on Android	1896
◦ Where You Can Use Maps V2	1896
◦ Licensing Terms for Maps V2	1897
◦ What You Need to Start	1897
◦ The Book Samples... And You!	1901
◦ Setting Up a Basic Map	1901
◦ Playing with the Map	1907
◦ Map Tiles	1910
◦ Placing Simple Markers	1910
◦ Seeing All the Markers	1914
◦ Flattening and Rotating Markers	1916
◦ Sprucing Up Your “Info Windows”	1919
◦ Images and Your Info Window	1925
◦ Setting the Marker Icon	1931
◦ Responding to Taps	1933
◦ Dragging Markers	1934
◦ The “Final” Limitations	1937
◦ A Bit More About IPC	1940
◦ Finding the User	1941
◦ Dealing with Runtime Permissions	1945

◦ Drawing Lines and Areas	1949
◦ Gestures and Controls	1952
◦ Tracking Camera Changes	1952
◦ Maps in Fragments and Pagers	1956
◦ Animating Marker Movement	1961
◦ Maps, of the Indoor Variety	1970
◦ Taking a Snapshot of a Map	1970
◦ MapFragment vs. MapView	1971
◦ About That AbstractMapActivity Class...	1971
◦ Helper Libraries for Maps V2	1976
◦ Problems with Maps V2 at Runtime	1980
◦ Problems with Maps V2 Deployment	1980
◦ What Non-Compliant Devices Show	1980
◦ Mapping Alternatives	1981
◦ News and Getting Help	1981
• Crafting Your Own Views	
◦ Prerequisites	1983
◦ Pick Your Poison	1983
◦ Colors, Mixed How You Like Them	1984
◦ ReverseChronometer: Simply a Custom Subclass	1995
◦ AspectLockedFrameLayout: A Custom Container	2001
◦ Mirror and MirroringFrameLayout: Draw It Yourself	2004
• Advanced Preferences	
◦ Prerequisites	2017
◦ Introducing PreferenceActivity	2017
◦ Intents for Headers or Preferences	2022
◦ Conditional Headers	2023
◦ Dependent Preferences	2028
◦ Nested Screens	2031
◦ Listening to Preference Changes	2034
◦ Defaults, and Defaults	2037
◦ Listening to Preference Value Changes	2038
◦ Dynamic ListPreference Contents	2038
◦ Dealing with External Changes to Preferences	2043
◦ Preferences in Device Settings App	2045
• Custom Dialogs and Preferences	
◦ Prerequisites	2049
◦ Your Dialog, Chocolate-Covered	2049
◦ Preferring Your Own Preferences, Preferably	2053
• Progress Indicators	
◦ Prerequisites	2061

◦ Progress Bars	2061
◦ ProgressBar and Threads	2065
◦ Tailoring Progress Bars	2068
◦ Progress Dialogs	2078
◦ Title Bar and Action Bar Progress Indicators	2080
◦ Direct Progress Indication	2082
• More Fun with Pagers	
◦ Prerequisites	2085
◦ Hosting ViewPager in a Fragment	2085
◦ Pages and the Action Bar	2087
◦ ViewPagers and Scrollable Contents	2090
◦ Columns for Large, Pages for Small	2090
◦ Introducing ArrayAdapter	2096
◦ Columns for Large Landscape, Pages for the Rest	2099
◦ Adding, Removing, and Moving Pages	2104
◦ Inside ArrayAdapter	2108
• Focus Management and Accessibility	
◦ Prerequisites	2123
◦ Prepping for Testing	2124
◦ Controlling the Focus	2124
◦ Accessibility and Focus	2134
◦ Accessibility Beyond Focus	2135
◦ Accessibility Beyond Impairment	2145
• Miscellaneous UI Tricks	
◦ Prerequisites	2147
◦ Full-Screen and Lights-Out Modes	2147
◦ Offering a Delayed Timeout	2158
• Event Bus Alternatives	
◦ Prerequisites	2163
◦ A Brief Note About the Sample Apps	2163
◦ Standard Intents as Event Bus	2163
◦ LocalBroadcastManager as Event Bus	2164
◦ greenrobot's EventBus 3.x	2175
◦ greenrobot's EventBus 2.x	2181
◦ Hey, What About Otto?	2182
• Tasks	
◦ Prerequisites	2183
◦ First, Some Terminology	2183
◦ And Now, a Bit About Task Killers	2189
◦ A Canary for the Task's Coal Mine	2196
◦ The Default User Experience	2199

◦ Explaining the Default Behavior	2203
◦ Basic Scenarios for Changing the Behavior	2206
◦ Dealing with the Persistent Tasks	2221
◦ Documents As Tasks	2223
◦ Other Task-Related Activity Properties	2226
◦ Other Task-Related Activity Methods	2235
• The Assist API (“Now On Tap”)	
◦ Prerequisites	2238
◦ What Data Gets Disclosed	2238
◦ Adding to the Data	2240
◦ Removing from the Data	2243
◦ Blocking Assist as a User	2246
◦ Implementing Your Own Assistant	2249
• The Data Binding Framework	
◦ Prerequisites	2263
◦ The What, Now?	2264
◦ The Basic Steps	2265
◦ The Extended Layout Resource	2271
◦ The Binding Expression Language	2275
◦ Observables and Updating the Binding	2278
◦ Two-Way Binding	2291
◦ Other Features of Note	2294
• Drag and Drop	
◦ Prerequisites	2323
◦ The Scope of Drag and Drop	2323
◦ The Pieces of Drag-and-Drop	2325
◦ <i>Drag-and-Drop, within an Activity</i>	2329
◦ Drag-and-Drop, Between Apps	2338
◦ Detecting Cross-App Drag Events	2350
◦ Intra-App Cross-Window Drag-and-Drop	2351
◦ Pondering Legacy Multi-Window	2354
◦ Dragging and Dropping Simple Stuff	2354
◦ Multi-Action Drag-and-Drop	2355
◦ <i>Nested Drop Targets</i>	2361
◦ Pondering Standards	2370
◦ Pondering Accessibility	2370
• Keyboard and Mouse Input	
◦ Prerequisites	2371
◦ Offering Keyboard Shortcuts	2371
◦ Custom Copy-and-Paste	2381
◦ Physical Keyboards and Focusing	2384

◦ Offering Mouse Context Menus	2389
◦ Offering Tooltips	2394
• Viewing PDFs	
◦ Prerequisites	2401
◦ The Criteria	2402
◦ The Classic Solution: ACTION_VIEW	2404
◦ The Really Bad Idea: Google Docs	2404
◦ The Built-In Option: PdfRenderer	2405
◦ The Thunder Lizard Choice: PDF.js	2414
◦ The Native Approach: Pdfium	2418
◦ What To Choose?	2422
• Home Screen App Widgets	
◦ Prerequisites	2425
◦ App Widgets and Security	2425
◦ The Big Picture for a Small App Widget	2426
◦ Crafting App Widgets	2427
◦ Another and Another	2434
◦ App Widgets: Their Life and Times	2435
◦ Controlling Your (App Widget's) Destiny	2435
◦ One Size May Not Fit All	2436
◦ Lockscreen Widgets	2442
◦ Preview Images	2448
◦ Being a Good Host	2450
• Adapter-Based App Widgets	
◦ Prerequisites	2451
◦ AdapterViews for App Widgets	2451
◦ Building Adapter-Based App Widgets	2452
• Content Provider Theory	
◦ Prerequisites	2467
◦ Using a Content Provider	2467
◦ Building Content Providers	2475
◦ Issues with Content Providers	2483
• Content Provider Implementation Patterns	
◦ Prerequisites	2485
◦ The Single-Table Database-Backed Content Provider	2485
◦ The Local-File Content Provider	2493
◦ The Protected Provider	2502
◦ The Stream Provider	2504
◦ FileProvider	2508
◦ StreamProvider	2513
• The Loader Framework	

◦ Prerequisites	2521
◦ Cursors: Issues with Management	2522
◦ Introducing the Loader Framework	2522
◦ Choosing an Implementation	2524
◦ Using CursorLoader	2524
◦ What Else Is Missing?	2527
◦ What Happens When...?	2527
◦ Writing a Custom Loader	2528
• The ContactsContract and CallLog Providers	
◦ Prerequisites	2537
◦ Introducing You to Your Contacts	2538
◦ <i>Pick a Peck of Pickled People</i>	2539
◦ Spin Through Your Contacts	2541
◦ Makin' Contacts	2547
◦ Looking at the CallLog	2552
• The CalendarContract Provider	
◦ Prerequisites	2558
◦ You Can't Be a Faker	2558
◦ Do You Have Room on Your Calendar?	2558
◦ Penciling In an Event	2564
• The MediaStore Provider	
◦ Prerequisites	2567
◦ What Is the MediaStore?	2568
◦ MediaStore and "Other" External Storage	2569
◦ How Does My Content Get Indexed?	2570
◦ How Do I Retrieve Video from the MediaStore?	2570
• Consuming Documents	
◦ Prerequisites	2579
◦ The Storage Access... What?	2579
◦ The Storage Access Framework Participants	2581
◦ Picking How to Pick (a Peck of Pickled Pepper Photos)	2581
◦ Opening a Document	2582
◦ Why We Want Things To Be Openable	2585
◦ The Rest of the CRUD	2586
◦ The DocumentFile Helper	2587
◦ Getting Durable Access	2588
◦ Extended Example: A Tiny Text Editor	2597
◦ Document Trees	2628
◦ Getting a Tree: Example	2630
◦ <i>Scoped Directory Access Bug</i>	2647
• Providing Documents	

◦ Prerequisites	2649
◦ Have Your Content, and Provide it Too	2649
◦ Key Provider Concepts	2651
◦ <i>Pieces of a Provider</i>	2652
◦ Optional Provider Capabilities	2667
• Encrypted Storage	
◦ Prerequisites	2674
◦ Scenarios for Encryption	2674
◦ Obtaining SQLCipher	2674
◦ Using SQLCipher	2675
◦ SQLCipher Limitations	2676
◦ Passwords and Sessions	2677
◦ About Those Passphrases...	2678
◦ Encrypted Preferences	2685
◦ IOCipher	2687
• Packaging and Distributing Data	
◦ Prerequisites	2689
◦ Packing a Database To Go	2689
• Advanced Database Techniques	
◦ Prerequisites	2693
◦ Full-Text Indexing	2693
• Data Backup	
◦ Prerequisites	2711
◦ First, Some Terminology	2712
◦ Differing Definitions of “Backup”	2712
◦ Implementing IT-Style Backup	2714
◦ The Google Backup Bootstrap	2739
◦ Boosting Backup Security	2748
◦ Alternative Approaches	2749
• SSL	
◦ Prerequisites	2753
◦ Basic SSL Operation	2753
◦ Problems in Paradise	2754
◦ Introducing Network Security Configuration	2757
◦ SSL Problems and Network Security Configuration	2760
◦ Other SSL Strengthening Techniques	2770
◦ Advanced Uses of CWAC-NetSecurity	2773
◦ NetCipher	2775
• NetCipher	
◦ Prerequisites	2777
◦ Network Security’s Got Onions	2777

◦ The NetCipher HTTP Integration APIs	2779
◦ The Rest of the Builder API	2786
• Embedding a Web Server	
◦ Prerequisites	2797
◦ Why a Web Server?	2797
◦ Introducing AsyncHttpServer	2799
◦ Embedding a Simple Server	2799
◦ Template-Driven Responses, with Handlebars	2816
◦ Supporting WebSockets	2821
◦ Securing the Web Server	2827
◦ Towards a Reusable Web Server Service	2834
• Miscellaneous Network Topics	
◦ Prerequisites	2845
◦ Downloading Files	2845
◦ Data Saver	2858
• Audio Playback	
◦ Prerequisites	2861
◦ Get Your Media On	2861
◦ MediaPlayer for Audio	2862
◦ Other Ways to Make Noise	2867
• Audio Recording	
◦ Prerequisites	2871
◦ Recording by Intent	2871
◦ Recording to Files	2874
◦ Recording to Streams	2879
◦ Raw Audio Input	2882
◦ Requesting the Microphone	2882
• Video Playback	
◦ Prerequisites	2883
◦ Moving Pictures	2883
• Using the Camera via 3rd-Party Apps	
◦ Prerequisites	2889
◦ Being Specific About Features	2889
◦ Still Photos: Letting the Camera App Do It	2890
◦ A Matter of Orientation	2898
◦ Scanning with ZXing	2905
◦ Videos: Letting the Camera App Do It	2906
◦ CWAC-Cam2: A CameraActivity Of Your Own	2909
◦ Directly Working with the Camera	2917
• Working Directly with the Camera	
◦ Prerequisites	2919

◦	Notes About the Code Snippets	2919
◦	A Tale of Two APIs	2920
◦	Performing Basic Camera Operations	2921
◦	Configuring the Still Camera	2944
◦	And Now, The Problems	2953
•	Media Routes	
◦	Prerequisites	2959
◦	Terminology	2959
◦	A Tale of Two MediaRouters	2960
◦	Attaching to MediaRouter	2961
◦	User Route Selection with MediaRouteActionProvider	2963
◦	Using Live Video Routes	2978
◦	Using Remote Playback Routes	2978
•	Supporting External Displays	
◦	Prerequisites	2997
◦	A History of External Displays	2997
◦	What is a Presentation?	2998
◦	Playing with External Displays	2999
◦	Detecting Displays	3005
◦	A Simple Presentation	3006
◦	A Simpler Presentation	3012
◦	Presentations and Configuration Changes	3018
◦	Presentations as Fragments	3019
◦	Another Sample Project: Slides	3030
◦	Device Support for Presentation	3038
◦	Presentations from a Service	3039
◦	Hey, What About Chromecast?	3042
•	Google Cast and Chromecast	
◦	Prerequisites	3045
◦	Here a Cast, There a Cast	3045
◦	Common Chromecast Development Notes	3047
◦	Your API Choices	3047
◦	Senders and Receivers	3048
◦	Supported Media Types	3049
◦	Cast SDK Dependencies	3050
◦	Developing Google Cast Apps	3052
•	The “Ten-Foot UI”	
◦	Prerequisites	3053
◦	What is the “Ten-Foot UI”?	3054
◦	Overscan	3054
◦	Navigation	3055

◦ Stylistic Considerations	3056
◦ The Leanback UI	3058
◦ Testing Your Theories	3073
• Putting the TVs All Together: Decktastic	
◦ Prerequisites	3076
◦ Introducing Decktastic	3076
◦ Implementing Decktastic	3080
• Creating a MediaRouteProvider	
◦ Prerequisites	3115
◦ Terminology	3115
◦ DIY Chromecast	3116
◦ Creating the MediaRouteProvider	3118
◦ Consuming the MediaRouteProvider	3129
◦ Implementing This “For Realz”	3133
• The Media Projection APIs	
◦ Prerequisites	3135
◦ Requesting Screenshots	3135
◦ Recording the Screen	3150
◦ Yet Another Sample: andshooter	3170
• Advanced Permissions	
◦ Prerequisites	3177
◦ Securing Yourself	3177
◦ Signature Permissions	3180
◦ The Custom Permission Vulnerability	3182
◦ Finding the Available Permissions	3192
• Restricted Profiles and UserManager	
◦ Prerequisites	3205
◦ Android Tablets and Multiple User Accounts	3205
◦ Determining What the User Can Do	3211
◦ Impacts of Device-Level Restrictions	3214
◦ Enabling Custom Restrictions	3214
◦ Implicit Intents May Go “Boom”	3225
• Miscellaneous Security Techniques	
◦ Prerequisites	3227
◦ Public Key Validation	3227
◦ Choosing Your Signing Keysize	3239
◦ Avoiding Accidental APIs	3240
◦ Other Ways to Expose Data	3245
◦ Jacking Attacks	3246
◦ Using FLAG_SECURE	3254
• AlarmManager and the Scheduled Service Pattern	

◦ Scenarios	3257
◦ Options	3258
◦ A Simple Example	3260
◦ The Five set...() Varieties	3262
◦ The Four Types of Alarms	3264
◦ When to Schedule Alarms	3264
◦ Archetype: Scheduled Service Polling	3266
◦ Staying Awake at Work	3270
◦ Warning: Not All Android Devices Play Nice	3275
◦ Debugging Alarms	3275
◦ WakefulBroadcastReceiver	3278
◦ Android 6.0 and the War on Background Processing	3282
◦ Android 7.0 and OnAlarmListener	3282
• PowerManager and WakeLocks	
◦ Prerequisites	3285
◦ Keeping the Screen On, UI-Style	3285
◦ The Role of the WakeLock	3286
◦ What WakefulIntentService Does	3287
• JobScheduler	
◦ Prerequisites	3289
◦ The Limitations of AlarmManager	3289
◦ Enter the JobScheduler	3290
◦ Employing JobScheduler	3290
◦ Pondering Backoff Criteria	3304
◦ Other JobScheduler Features	3306
◦ JobScheduler Period Limits	3306
◦ GcmNetworkManager	3306
◦ Periodic Work, Across Device Versions	3307
◦ Android 6.0 and “the War on Background Processing”	3311
◦ Scheduling Content Monitoring	3327
• Accessing Location-Based Services	
◦ Prerequisites	3333
◦ Location Providers: They Know Where You’re Hiding	3334
◦ Finding Yourself	3334
◦ On the Move	3336
◦ Getting Locations via PendingIntent	3346
◦ Are We There Yet? Are We There Yet? Are We There Yet?	3346
◦ Testing... Testing...	3347
◦ Alternative Flavors of Updates	3348
◦ The Fused Option	3349
◦ Locations and Features	3350

- [The Fused Location Provider](#)
 - Prerequisites 3351
 - Why Use the Fused Location Provider? 3351
 - Why Not Use the Fused Location Provider? 3352
 - Finding Our Location, Once 3352
 - Requesting Location Updates 3358
 - I Can Haz Location? 3360
- [Working with the Clipboard](#)
 - Prerequisites 3369
 - Working with the Clipboard 3369
 - ClipData and Drag-and-Drop 3374
 - Monitoring the Clipboard 3375
 - The Android 4.3 Clipboard Bug 3376
- [Telephony](#)
 - Prerequisites 3379
 - Report To The Manager 3380
 - You Make the Call! 3380
 - No, Really, You Make the Call! 3383
- [Working With SMS](#)
 - Prerequisites 3385
 - Sending Out an SOS, Give or Take a Letter 3386
 - Monitoring and Receiving SMS 3394
 - The SMS Inbox 3400
 - Asking to Change the Default 3401
 - SMS and the Emulator 3402
- [NFC](#)
 - Prerequisites 3403
 - What Is NFC? 3403
 - To NDEF, Or Not to NDEF 3405
 - NDEF Modalities 3405
 - NDEF Structure and Android's Translation 3406
 - The Reality of NDEF 3407
 - Sources of Tags 3409
 - Writing to a Tag 3409
 - Responding to a Tag 3418
 - Expected Pattern: Bootstrap 3419
 - Mobile Devices are Mobile 3420
 - Enabled and Disabled 3420
 - Android Beam 3420
 - Beaming Files 3428
 - Another Sample: SecretAgentMan 3429

◦ Additional Resources	3439
• Device Administration	
◦ Prerequisites	3441
◦ Objectives and Scope	3441
◦ Defining and Registering an Admin Component	3442
◦ Going Into Lockdown	3448
◦ Passwords and Device Administration	3455
◦ Getting Along with Others	3460
• Basic Use of Sensors	
◦ Prerequisites	3461
◦ The Sensor Abstraction Model	3461
◦ Considering Rates	3462
◦ Reading Sensors	3463
◦ Batching Sensor Readings	3474
• Printing and Document Generation	
◦ Prerequisites	3476
◦ The Android Print System	3476
◦ About the Sample App	3477
◦ Printing a Bitmap	3478
◦ Printing an HTML Document	3480
◦ Printing a PDF File	3485
◦ Printing Using a Canvas	3493
◦ Print Jobs	3496
◦ Printing, Threads, and Services	3496
◦ Printing Prior to Android 4.4	3499
◦ HTML Generation	3499
◦ PDF Generation Options	3503
• Dealing with Different Hardware	
◦ Prerequisites	3505
◦ Filtering Out Devices	3505
◦ Runtime Capability Detection	3508
◦ Dealing with Device Bugs	3509
• Writing and Using Parcelables	
◦ Prerequisites	3511
◦ The Role of Parcelable	3511
◦ Writing a Parcelable	3512
◦ The Limitations of Parcelable	3518
◦ Beware the PendingIntent	3522
• Responding to URLs	
◦ Prerequisites	3529
◦ Manifest Modifications	3529

- Creating a Custom URL 3531
- Reacting to the Link 3532
- App Links 3534
- [App Shortcuts](#)
 - Prerequisites 3545
 - Enabling Deep Dives 3546
 - App Shortcuts, from the User’s POV 3546
 - Offering Manifest App Shortcuts 3548
 - Offering Dynamic App Shortcuts 3554
 - Privacy, Security, and App Shortcuts 3566
- [Plugin Patterns](#)
 - Prerequisites 3567
 - Definitions, Scenarios, and Scope 3567
 - The Keys to Any Plugin System 3568
 - Case Study: DashClock 3576
 - Other Plugin Examples 3579
- [PackageManager Tricks](#)
 - Prerequisites 3599
 - Asking Around 3599
 - Preferred Activities 3603
 - Middle Management 3609
- [Remote Services and the Binding Pattern](#)
 - Prerequisites 3613
 - The Binding Pattern 3614
 - When IPC Attacks! 3620
 - Service From Afar 3623
 - Tightening Up the Security 3629
 - Servicing the Service 3634
 - Thinking About Security 3638
 - The “Everlasting Service” Anti-Pattern 3638
- [Advanced Manifest Tips](#)
 - Prerequisites 3641
 - Just Looking For Some Elbow Room 3641
 - Using an Alias 3651
 - Getting Meta (Data) 3652
- [Miscellaneous Integration Tips](#)
 - Prerequisites 3657
 - Direct Share 3657
 - Take the Shortcut 3666
 - Homing Beacons for Intents 3673
 - Integrating with Text Selection 3673

◦ Quick Settings and TileService	3685
• Reusable Components	
◦ Prerequisites	3695
◦ Where Do I Find Them?	3695
◦ How Are They Packaged?	3696
◦ How Do I Create Them?	3697
◦ Other Considerations for Publishing Reusable Code	3701
• Replacing App Code Dynamically	
◦ Prerequisites	3703
◦ Typical Objectives	3704
◦ The Challenges	3706
◦ The Scripting Solution	3708
◦ The Hybrid Solution	3709
◦ The Patch Solution	3710
◦ A DIY Solution	3711
◦ Is Any Of This a Good Idea?	3734
• Android Studio Editors and Dialogs	
◦ Prerequisites	3735
◦ Project Structure	3735
◦ Translations Editor	3745
• Advanced Emulator Capabilities	
◦ Prerequisites	3751
◦ Other Notable Configuration Options	3751
◦ The Emulator Sidebar	3757
◦ Emulator Window Operations	3768
◦ Headless Operation	3768
• Lint and the Support Annotations	
◦ Prerequisites	3771
◦ What It Is	3772
◦ When It Runs	3772
◦ What to Fix	3773
◦ What to Configure	3774
◦ Support Annotations	3778
• Inspecting Layouts	
◦ Prerequisites	3791
◦ The Layout Inspector	3791
◦ Hierarchy View	3793
• Screenshots and Screencasts	
◦ Prerequisites	3799
◦ Collecting from Android Studio	3799
◦ Screencasts	3803

◦ Collecting from the Command Line	3803
◦ Collecting from Another App	3805
◦ Tips and Tricks	3806
• ADB Tips and Tricks	
◦ Prerequisites	3807
◦ This is the Droid That You Are Looking For	3807
◦ Installing and Uninstalling Apps	3808
◦ Playing with Permissions	3808
◦ Starting and Stopping Components	3809
◦ Killing Processes and Clearing Data	3810
◦ Changing Display Metrics	3810
• Issues with Speed	
◦ Prerequisites	3815
◦ Getting Things Done	3815
◦ Your UI Seems... Janky	3816
◦ Not Far Enough in the Background	3816
◦ Playing with Speed	3817
• Finding CPU Bottlenecks	
◦ Prerequisites	3819
◦ Android Studio Monitors	3820
◦ Method Tracing	3821
◦ Other General CPU Measurement Techniques	3829
◦ UI “Jank” Measurement	3830
• Focus On: NDK	
◦ Prerequisites	3847
◦ The Role of the NDK	3848
◦ NDK Installation and Project Setup	3851
◦ Writing Your Makefile(s)	3855
◦ Building Your Library	3856
◦ Using Your Library Via JNI	3857
◦ Building and Deploying Your Project	3863
◦ Gradle and the NDK	3864
• Improving CPU Performance in Java	
◦ Prerequisites	3873
◦ Reduce CPU Utilization	3873
◦ Reduce Time on the Main Application Thread	3878
◦ Improve Throughput and Responsiveness	3886
• Finding and Eliminating Jank	
◦ Prerequisites	3889
◦ The Case: ThreePaneDemoBC	3889
◦ Are We Janky?	3890

- Finding the Source of the Jank 3890
- Where Things Went Wrong 3901
- Removing the Jank 3901
- Frame Metrics API 3902
- [Issues with Bandwidth](#)
 - Prerequisites 3909
 - You're Using Too Much of the Slow Stuff 3910
 - You're Using Too Much of the Expensive Stuff 3910
 - You're Using Too Much of Somebody Else's Stuff 3911
 - You're Using Too Much... And There Is None 3912
- [Focus On: TrafficStats](#)
 - Prerequisites 3913
 - TrafficStats Basics 3913
 - Example: TrafficMonitor 3915
 - Other Ways to Employ TrafficStats 3924
- [Measuring Bandwidth Consumption](#)
 - Prerequisites 3927
 - On-Device Measurement 3927
 - Off-Device Measurement 3931
 - Android Studio Network Monitor 3933
- [Being Smarter About Bandwidth](#)
 - Prerequisites 3935
 - Bandwidth Savings 3935
 - Bandwidth Shaping 3939
 - Avoiding Metered Connections 3943
- [Issues with Application Heap](#)
 - Prerequisites 3945
 - You Are in a Heap of Trouble 3946
 - Determining Your Heap Size At Runtime 3947
 - Fragments of Memory 3947
 - Getting a Trim 3948
 - Warning: Contains Graphic Images 3949
 - Releasing SQLite Memory 3961
 - Cheating 3961
 - *The 1MB IPC Transaction Limit* 3962
- [Finding Memory Leaks](#)
 - Prerequisites 3965
 - Android Studio Realtime Monitor 3966
 - Getting Heap Dumps 3968
 - Analyzing Heap Dumps in Android Studio 3969
 - Common Leak Scenarios 3976

- A Canary in a Leaky Coal Mine 3984
- [Issues with System RAM](#)
 - Prerequisites 3995
 - Can't We All Just Get Along? 3995
 - Contributors to System RAM Consumption 3996
 - Measuring System RAM Consumption: Tools 3997
 - Measuring System RAM Consumption: Runtime 4013
 - Learn To Let Go (Of Your Heap) 4014
- [Issues with Battery Life](#)
 - Prerequisites 4017
 - You're Getting Blamed 4018
 - Not All Batteries Are Created Equal 4019
 - Stretching Out the Last mWh 4019
- [Power Measurement Options](#)
 - Prerequisites 4021
 - batterystats and the Battery Historian 4022
 - The Qualcomm Tool (That Must Not Be Named) 4033
 - PowerTutor 4034
 - Battery Screen in Settings Application 4038
 - BatteryInfo Dump 4040
- [Sources of Power Drain](#)
 - Prerequisites 4043
 - Screen 4044
 - Disk I/O 4045
 - WiFi and Mobile Data 4046
 - GPS 4049
 - Camera 4050
 - Additional Sources 4050
- [Addressing Application Size Issues](#)
 - Prerequisites 4053
 - The APK Analyzer 4053
 - Java Code, and the 64K Method Limit 4056
 - **Native Code** **4061**
 - Images 4062
 - APK Expansion Files 4064
- [Crash Reporting Using ACRA](#)
 - Prerequisites 4065
 - What Happens When Things Go "Boom"? 4065
 - Introducing ACRA 4066
 - Where ACRA Reports Crashes 4067
 - ACRA Integration Basics 4069

◦	What the User Sees	4074
◦	What You See	4081
◦	Customizing Where Reports Go	4088
◦	Adding Additional Data	4089
◦	Removing Data	4092
◦	End-User Configuration	4093
◦	ACRA and Processes	4093
•	JVM Scripting Languages	
◦	Prerequisites	4095
◦	Languages on Languages	4095
◦	A Brief History of JVM Scripting	4096
◦	Limitations	4097
◦	SL4A and JVM Languages	4098
◦	Embedding JVM Languages	4098
◦	Other JVM Scripting Languages	4112
•	In-App Diagnostics	
◦	Prerequisites	4115
◦	The Diagnostic Activity	4116
◦	The Diagnostic Web App	4127
◦	The Diagnostic Overlay	4135
•	Anti-Patterns	
◦	Prerequisites	4151
◦	Leak Threads... Or Things Attached to Threads	4151
◦	Use Large Heap Unnecessarily	4153
◦	Misuse the MENU Button	4155
◦	Interfere with Navigation	4156
◦	Use android:sharedUserId	4158
◦	Implement a “Quit” Button	4159
◦	Terminate Your Process	4161
◦	Try to Hide from the User	4162
◦	Use Multiple Processes	4163
◦	Hog System Resources	4165
•	Widget Catalog: AdapterViewFlipper	
◦	Key Usage Tips	4167
◦	A Sample Usage	4168
◦	Visual Representation	4168
•	Widget Catalog: CalendarView	
◦	Key Usage Tips	4169
◦	A Sample Usage	4170
◦	Visual Representation	4171
•	Widget Catalog: DatePicker	

◦ Key Usage Tips	4175
◦ A Sample Usage	4176
◦ Visual Representation	4178
• Widget Catalog: <code>ExpandableListView</code>	
◦ Key Usage Tips	4183
◦ A Sample Usage	4184
◦ Visual Representation	4190
• Widget Catalog: <code>SeekBar</code>	
◦ Key Usage Tips	4193
◦ A Sample Usage	4193
◦ Visual Representation	4196
• Widget Catalog: <code>SlidingDrawer</code>	
◦ Key Usage Tips	4199
◦ A Sample Usage	4200
◦ Visual Representation	4201
• Widget Catalog: <code>StackView</code>	
◦ Key Usage Tips	4205
◦ A Sample Usage	4206
◦ Visual Representation	4207
• Widget Catalog: <code>TabHost</code> and <code>TabWidget</code>	
◦ Deprecation Notes	4209
◦ Key Usage Tips	4209
◦ A Sample Usage	4210
◦ Visual Representation	4212
• Widget Catalog: <code>TimePicker</code>	
◦ Key Usage Tips	4215
◦ A Sample Usage	4215
◦ Visual Representation	4217
• Widget Catalog: <code>ViewFlipper</code>	
◦ Key Usage Tips	4221
◦ A Sample Usage	4222
◦ Visual Representation	4223
• Chrome and Chrome OS	
◦ Prerequisites	4225
◦ How This Works	4225
◦ Testing Your App on Chrome OS	4227
◦ Be Prepared To Be Wiped Out	4235
◦ Enabling Your App for Chrome OS	4235
◦ Your App on Chrome OS	4236
◦ Distribution Options	4245
◦ Apps Sans Role	4245

◦ Getting Help	4246
• Android Things Basics	
◦ Prerequisites	4247
◦ <i>Um, What’s a Thing?</i>	4248
◦ <i>Headless, But No Horseman</i>	4249
◦ <i>Setting Up a Raspberry-Flavored Thing</i>	4249
◦ <i>An App For Your Thing</i>	4253
◦ <i>Control Interfaces</i>	4257
◦ <i>What’s Different?</i>	4260
◦ <i>Environment Details</i>	4263
◦ <i>Considerations</i>	4265
• Device Catalog: Kindle Fire	
◦ Prerequisites	4269
◦ Introducing the Kindle Fire series	4269
◦ What Features and Configurations Does It Use?	4270
◦ What Is Really Different?	4272
◦ Getting Your Development Environment Established	4278
◦ How Does Distribution Work?	4280
◦ Amazon Equivalents of Google Services	4281
◦ Getting Help with the Kindle Fire	4282
• Device Catalog: BlackBerry	
◦ I Thought BlackBerry Had Their Own OS?	4284
◦ What Else Is Different?	4284
◦ What Are We Making?	4287
◦ Getting Your Development Environment Established	4287
◦ How Does Distribution Work?	4290
• Device Catalog: Android TV	
◦ Prerequisites	4293
◦ Hey, Wait a Minute... I Thought the Name Was “Google TV”?	4293
◦ Some Android TV Hardware	4294
◦ What Features and Configurations Does It Use?	4296
◦ What Is Really Different?	4297
◦ Getting Your Development Environment Established	4299
◦ How Does Distribution Work?	4301
• Device Catalog: Amazon Fire TV and Fire TV Stick	
◦ Prerequisites	4305
◦ Introducing the Fire TV Devices	4305
◦ What Features and Configurations Do They Use?	4311
◦ What Is Really Different?	4312
◦ Casting and Fire TV	4313
◦ Getting Your Development Environment Established	4314

- Working with the Remote and Controller 4316
- How Does Distribution Work? 4318
- Getting Help 4318
- [Appendix A: CWAC Libraries](#)
 - cwac-adapter 4321
 - cwac-cam2 4321
 - cwac-colormixer 4322
 - ***cwac-crossport*** **4322**
 - cwac-layouts 4322
 - cwac-merge 4322
 - cwac-netsecurity 4323
 - cwac-pager 4323
 - cwac-presentation 4323
 - cwac-provider 4323
 - cwac-richedit 4324
 - cwac-sacklist 4324
 - cwac-security 4324
 - cwac-strictmodeex 4324
 - cwac-wakeful 4324
- [Appendix B: Community Theater and the Appinars](#)
 - Viewing the Appinar Roster 4325
 - Managing Appinars 4328
 - Viewing an Appinar 4331