



Contents

Preface **xix**

Before You Begin **xxxi**

I Introduction to Android **I**

1.1	Introduction	2
1.2	Android—The World’s Leading Mobile Operating System	3
1.3	Android Features	3
1.4	Android Operating System	7
1.4.1	Android 2.2 (Froyo)	7
1.4.2	Android 2.3 (Gingerbread)	8
1.4.3	Android 3.0 through 3.2 (Honeycomb)	8
1.4.4	Android 4.0 through 4.0.4 (Ice Cream Sandwich)	8
1.4.5	Android 4.1–4.3 (Jelly Bean)	9
1.4.6	Android 4.4 (KitKat)	10
1.5	Downloading Apps from Google Play	11
1.6	Packages	12
1.7	Android Software Development Kit (SDK)	13
1.8	Object-Oriented Programming: A Quick Refresher	16
1.8.1	The Automobile as an Object	17
1.8.2	Methods and Classes	17
1.8.3	Instantiation	17
1.8.4	Reuse	17
1.8.5	Messages and Method Calls	17
1.8.6	Attributes and Instance Variables	18
1.8.7	Encapsulation	18
1.8.8	Inheritance	18
1.8.9	Object-Oriented Analysis and Design (OOAD)	18
1.9	Test-Driving the Doodlz App in an Android Virtual Device (AVD)	19
1.9.1	Running the Doodlz App in the Nexus 4 Smartphone AVD	19
1.9.2	Running the Doodlz App in a Tablet AVD	28
1.9.3	Running the Doodlz App on an Android Device	30
1.10	Building Great Android Apps	30
1.11	Android Development Resources	32
1.12	Wrap-Up	35

2 Welcome App 38

Dive-Into® the Android Developer Tools: Introducing Visual GUI Design, Layouts, Accessibility and Internationalization

2.1	Introduction	39
2.2	Technologies Overview	40
2.2.1	Android Developer Tools IDE	40
2.2.2	TextViews and ImageViews	40
2.2.3	App Resources	40
2.2.4	Accessibility	40
2.2.5	Internationalization	40
2.3	Creating an App	41
2.3.1	Launching the Android Developer Tools IDE	41
2.3.2	Creating a New Project	41
2.3.3	New Android Application Dialog	42
2.3.4	Configure Project Step	43
2.3.5	Configure Launcher Icon Step	43
2.3.6	Create Activity Step	45
2.3.7	Blank Activity Step	46
2.4	Android Developer Tools Window	47
2.4.1	Package Explorer Window	48
2.4.2	Editor Windows	48
2.4.3	Outline Window	48
2.4.4	App Resource Files	48
2.4.5	Graphical Layout Editor	49
2.4.6	The Default GUI	49
2.5	Building the App's GUI with the Graphical Layout Editor	51
2.5.1	Adding Images to the Project	51
2.5.2	Changing the Id Property of the RelativeLayout and the TextView	52
2.5.3	Configuring the TextView	53
2.5.4	Adding ImageViews to Display the Images	57
2.6	Running the Welcome App	59
2.7	Making Your App Accessible	60
2.8	Internationalizing Your App	62
2.9	Wrap-Up	66

3 Tip Calculator App 69

Introducing GridLayout, LinearLayout, EditText, SeekBar, Event Handling, NumberFormat and Defining App Functionality with Java

3.1	Introduction	70
3.2	Test-Driving the Tip Calculator App	71
3.3	Technologies Overview	72
3.3.1	Class Activity	72
3.3.2	Activity Lifecycle Methods	72
3.3.3	Arranging Views with LinearLayout and GridLayout	73

3.3.4	Creating and Customizing the GUI with the Graphical Layout Editor and the Outline and Properties Windows	73
3.3.5	Formatting Numbers as Locale-Specific Currency and Percentage Strings	74
3.3.6	Implementing Interface <code>TextWatcher</code> for Handling <code>EditText</code> Text Changes	74
3.3.7	Implementing Interface <code>OnSeekBarChangeListener</code> for Handling <code>SeekBar</code> Thumb Position Changes	74
3.3.8	<code>AndroidManifest.xml</code>	75
3.4	Building the App's GUI	75
3.4.1	<code>GridLayout</code> Introduction	75
3.4.2	Creating the <code>TipCalculator</code> Project	77
3.4.3	Changing to a <code>GridLayout</code>	77
3.4.4	Adding the <code>TextViews</code> , <code>EditText</code> , <code>SeekBar</code> and <code>LinearLayouts</code>	78
3.4.5	Customizing the Views to Complete the Design	80
3.5	Adding Functionality to the App	84
3.6	<code>AndroidManifest.xml</code>	92
3.7	Wrap-Up	93

4 Twitter® Searches App 97

SharedPreferences, Collections, ImageButton, ListView, ListActivity, ArrayAdapter, Implicit Intents and AlertDialogs

4.1	Introduction	98
4.2	Test-Driving the App	99
4.2.1	Importing the App and Running It	99
4.2.2	Adding a Favorite Search	100
4.2.3	Viewing Twitter Search Results	101
4.2.4	Editing a Search	102
4.2.5	Sharing a Search	104
4.2.6	Deleting a Search	104
4.2.7	Scrolling Through Saved Searches	105
4.3	Technologies Overview	105
4.3.1	<code>ListView</code>	105
4.3.2	<code>ListActivity</code>	106
4.3.3	Customizing a <code>ListActivity</code> 's Layout	106
4.3.4	<code>ImageButton</code>	106
4.3.5	<code>SharedPreferences</code>	106
4.3.6	Intents for Launching Other Activities	107
4.3.7	<code>AlertDialog</code>	107
4.3.8	<code>AndroidManifest.xml</code>	108
4.4	Building the App's GUI	108
4.4.1	Creating the Project	108
4.4.2	<code>activity_main.xml</code> Overview	109
4.4.3	Adding the <code>GridLayout</code> and Components	110

x Contents

4.4.4	Graphical Layout Editor Toolbar	115
4.4.5	ListView Item's Layout: list_item.xml	116
4.5	Building the MainActivity Class	117
4.5.1	package and import Statements	117
4.5.2	Extending ListActivity	119
4.5.3	Fields of Class MainActivity	119
4.5.4	Overriding Activity Method onCreate	120
4.5.5	Anonymous Inner Class That Implements the saveButton's onClickListener to Save a New or Updated Search	122
4.5.6	addTaggedSearch Method	124
4.5.7	Anonymous Inner Class That Implements the ListView's onItemClickListener to Display Search Results	125
4.5.8	Anonymous Inner Class That Implements the ListView's onItemLongClickListener to Share, Edit or Delete a Search	127
4.5.9	shareSearch Method	129
4.5.10	deleteSearch Method	130
4.6	AndroidManifest.xml	132
4.7	Wrap-Up	132

5 Flag Quiz App 136

Fragments, Menus, Preferences, AssetManager, Tweened Animations, Handler, Toasts, Explicit Intents, Layouts for Multiple Device Orientations

5.1	Introduction	137
5.2	Test-Driving the Flag Quiz App	139
5.2.1	Importing the App and Running It	139
5.2.2	Configuring the Quiz	139
5.2.3	Taking the Quiz	141
5.3	Technologies Overview	143
5.3.1	Menus	143
5.3.2	Fragments	143
5.3.3	Fragment Lifecycle Methods	144
5.3.4	Managing Fragments	144
5.3.5	Preferences	144
5.3.6	assets Folder	144
5.3.7	Resource Folders	145
5.3.8	Supporting Different Screen Sizes and Resolutions	145
5.3.9	Determining the Screen Size	146
5.3.10	Toasts for Displaying Messages	146
5.3.11	Using a Handler to Execute a Runnable in the Future	146
5.3.12	Applying an Animation to a View	146
5.3.13	Logging Exception Messages	147
5.3.14	Using an Explicit Intent to Launch Another Activity in the Same App	147
5.3.15	Java Data Structures	147
5.4	Building the GUI and Resource Files	147

5.4.1	Creating the Project	147
5.4.2	strings.xml and Formatted String Resources	148
5.4.3	arrays.xml	149
5.4.4	colors.xml	150
5.4.5	dimens.xml	150
5.4.6	activity_settings.xml Layout	151
5.4.7	activity_main.xml Layout for Phone and Tablet Portrait Orientation	151
5.4.8	fragment_quiz.xml Layout	151
5.4.9	activity_main.xml Layout for Tablet Landscape Orientation	154
5.4.10	preferences.xml for Specifying the App's Settings	155
5.4.11	Creating the Flag Shake Animation	156
5.5	MainActivity Class	158
5.5.1	package Statement, import Statements and Fields	158
5.5.2	Overridden Activity Method onCreate	159
5.5.3	Overridden Activity Method onStart	161
5.5.4	Overridden Activity Method onCreateOptionsMenu	161
5.5.5	Overridden Activity Method onOptionsItemSelected	162
5.5.6	Anonymous Inner Class That Implements OnSharedPreferencesChangeListener	163
5.6	QuizFragment Class	164
5.6.1	package Statement and import Statements	164
5.6.2	Fields	165
5.6.3	Overridden Fragment Method onCreateView	166
5.6.4	Method updateGuessRows	168
5.6.5	Method updateRegions	169
5.6.6	Method resetQuiz	169
5.6.7	Method loadNextFlag	171
5.6.8	Method getCountryName	173
5.6.9	Anonymous Inner Class That Implements OnClickListener	173
5.6.10	Method disableButtons	176
5.7	SettingsFragment Class	176
5.8	SettingsActivity Class	177
5.9	AndroidManifest.xml	177
5.10	Wrap-Up	178

6 Cannon Game App 182

Listening for Touches, Manual Frame-By-Frame Animation, Graphics, Sound, Threading, SurfaceView and SurfaceHolder

6.1	Introduction	183
6.2	Test-Driving the Cannon Game App	185
6.3	Technologies Overview	185
6.3.1	Attaching a Custom View to a Layout	185
6.3.2	Using the Resource Folder raw	185
6.3.3	Activity and Fragment Lifecycle Methods	185

6.3.4	Overriding View Method onTouchEvent	186
6.3.5	Adding Sound with SoundPool and AudioManager	186
6.3.6	Frame-by-Frame Animation with Threads, SurfaceView and SurfaceHolder	186
6.3.7	Simple Collision Detection	187
6.3.8	Drawing Graphics Using Paint and Canvas	187
6.4	Building the App's GUI and Resource Files	187
6.4.1	Creating the Project	187
6.4.2	strings.xml	188
6.4.3	fragment_game.xml	188
6.4.4	activity_main.xml	189
6.4.5	Adding the Sounds to the App	189
6.5	Class Line Maintains a Line's Endpoints	189
6.6	MainActivity Subclass of Activity	190
6.7	CannonGameFragment Subclass of Fragment	190
6.8	CannonView Subclass of View	192
6.8.1	package and import Statements	192
6.8.2	Instance Variables and Constants	193
6.8.3	Constructor	194
6.8.4	Overriding View Method onSizeChanged	196
6.8.5	Method newGame	197
6.8.6	Method updatePositions	198
6.8.7	Method fireCannonball	201
6.8.8	Method alignCannon	202
6.8.9	Method drawGameElements	203
6.8.10	Method showGameOverDialog	205
6.8.11	Methods stopGame and releaseResources	206
6.8.12	Implementing the SurfaceHolder.Callback Methods	207
6.8.13	Overriding View Method onTouchEvent	208
6.8.14	CannonThread: Using a Thread to Create a Game Loop	209
6.9	Wrap-Up	210

7 Doodlz App 215

Two-Dimensional Graphics, Canvas, Bitmap, Accelerometer, SensorManager, Multitouch Events, MediaStore, Printing, Immersive Mode

7.1	Introduction	216
7.2	Technologies Overview	218
7.2.1	Using SensorManager to Listen for Accelerometer Events	218
7.2.2	Custom DialogFragments	218
7.2.3	Drawing with Canvas and Bitmap	219
7.2.4	Processing Multiple Touch Events and Storing Lines in Paths	219
7.2.5	Android 4.4 Immersive Mode	219
7.2.6	GestureDetector and SimpleOnGestureListener	219
7.2.7	Saving the Drawing to the Device's Gallery	219

7.2.8	Android 4.4 Printing and the Android Support Library's PrintHelper Class	220
7.3	Building the App's GUI and Resource Files	220
7.3.1	Creating the Project	220
7.3.2	strings.xml	220
7.3.3	dimens.xml	221
7.3.4	Menu for the DoodleFragment	222
7.3.5	activity_main.xml Layout for MainActivity	223
7.3.6	fragment_doodle.xml Layout for DoodleFragment	223
7.3.7	fragment_color.xml Layout for ColorDialogFragment	224
7.3.8	fragment_line_width.xml Layout for LineWidthDialogFragment	226
7.3.9	Adding Class EraseImageDialogFragment	227
7.4	MainActivity Class	228
7.5	DoodleFragment Class	229
7.6	DoodleView Class	236
7.7	ColorDialogFragment Class	248
7.8	LineWidthDialogFragment Class	251
7.9	EraseImageDialogFragment Class	255
7.10	Wrap-Up	256

8 Address Book App 260

ListFragment, FragmentTransactions and the Fragment Back Stack, Threading and AsyncTasks, CursorAdapter, SQLite and GUI Styles

8.1	Introduction	261
8.2	Test-Driving the Address Book App	264
8.3	Technologies Overview	264
8.3.1	Displaying Fragments with FragmentTransactions	265
8.3.2	Communicating Data Between a Fragment and a Host Activity	265
8.3.3	Method onSaveInstanceState	265
8.3.4	Defining Styles and Applying Them to GUI Components	265
8.3.5	Specifying a Background for a TextView	265
8.3.6	Extending Class ListFragment to Create a Fragment That Contains a ListView	266
8.3.7	Manipulating a SQLite Database	266
8.3.8	Performing Database Operations Outside the GUI Thread with AsyncTasks	266
8.4	Building the GUI and Resource Files	266
8.4.1	Creating the Project	266
8.4.2	Creating the App's Classes	267
8.4.3	strings.xml	267
8.4.4	styles.xml	268
8.4.5	textview_border.xml	269
8.4.6	MainActivity's Layout: activity_main.xml	270
8.4.7	DetailsFragment's Layout: fragment_details.xml	270
8.4.8	AddEditFragment's Layout: fragment_add_edit.xml	272
8.4.9	Defining the Fragments' Menus	273

8.5	MainActivity Class	274
8.6	ContactListFragment Class	280
8.7	AddEditFragment Class	287
8.8	DetailsFragment Class	293
8.9	DatabaseConnector Utility Class	301
8.10	Wrap-Up	306

9 Google Play and App Business Issues **310**

9.1	Introduction	311
9.2	Preparing Your Apps for Publication	311
9.2.1	Testing Your App	312
9.2.2	End User License Agreement	312
9.2.3	Icons and Labels	312
9.2.4	Versioning Your App	313
9.2.5	Licensing to Control Access to Paid Apps	313
9.2.6	Obfuscating Your Code	313
9.2.7	Getting a Private Key for Digitally Signing Your App	314
9.2.8	Screenshots	314
9.2.9	Promotional App Video	315
9.3	Pricing Your App: Free or Fee	316
9.3.1	Paid Apps	317
9.3.2	Free Apps	317
9.4	Monetizing Apps with In-App Advertising	318
9.5	Monetizing Apps: Using In-App Billing to Sell Virtual Goods	319
9.6	Registering at Google Play	320
9.7	Setting Up a Google Wallet Merchant Account	321
9.8	Uploading Your Apps to Google Play	322
9.9	Launching the Play Store from Within Your App	323
9.10	Managing Your Apps in Google Play	324
9.11	Other Android App Marketplaces	324
9.12	Other Popular Mobile App Platforms	324
9.13	Marketing Your Apps	325
9.14	Wrap-Up	329

A Introduction to Java Applications **332**

A.1	Introduction	333
A.2	Your First Program in Java: Printing a Line of Text	333
A.3	Modifying Your First Java Program	337
A.4	Displaying Text with printf	339
A.5	Another Application: Adding Integers	339
A.6	Memory Concepts	343
A.7	Arithmetic	344
A.8	Decision Making: Equality and Relational Operators	347
A.9	Wrap-Up	351

B	Introduction to Classes, Objects, Methods and Strings	356
B.1	Introduction	357
B.2	Declaring a Class with a Method and Instantiating an Object of a Class	357
B.3	Declaring a Method with a Parameter	360
B.4	Instance Variables, <i>set</i> Methods and <i>get</i> Methods	363
B.5	Primitive Types vs. Reference Types	367
B.6	Initializing Objects with Constructors	368
B.7	Floating-Point Numbers and Type <code>double</code>	370
B.8	Wrap-Up	374
C	Control Statements	378
C.1	Introduction	379
C.2	Algorithms	379
C.3	Pseudocode	380
C.4	Control Structures	380
C.5	<code>if</code> Single-Selection Statement	381
C.6	<code>if...else</code> Double-Selection Statement	381
C.7	<code>while</code> Repetition Statement	384
C.8	Case Study: Counter-Controlled Repetition	384
C.9	Case Study: Sentinel-Controlled Repetition	388
C.10	Case Study: Nested Control Statements	393
C.11	Compound Assignment Operators	396
C.12	Increment and Decrement Operators	396
C.13	Primitive Types	398
C.14	Essentials of Counter-Controlled Repetition	399
C.15	<code>for</code> Repetition Statement	400
C.16	Examples Using the <code>for</code> Statement	402
C.17	<code>do...while</code> Repetition Statement	404
C.18	<code>switch</code> Multiple-Selection Statement	405
C.19	<code>break</code> and <code>continue</code> Statements	412
C.20	Logical Operators	412
C.21	Wrap-Up	415
D	Methods: A Deeper Look	423
D.1	Introduction	424
D.2	Program Modules in Java	424
D.3	<code>static</code> Methods, <code>static</code> Fields and Class <code>Math</code>	425
D.4	Declaring Methods with Multiple Parameters	427
D.5	Notes on Declaring and Using Methods	430
D.6	Method-Call Stack and Activation Records	431
D.7	Argument Promotion and Casting	431
D.8	Java API Packages	432

D.9	Introduction to Random-Number Generation	433
	D.9.1 Scaling and Shifting of Random Numbers	434
	D.9.2 Random-Number Repeatability for Testing and Debugging	435
D.10	Case Study: A Game of Chance; Introducing Enumerations	436
D.11	Scope of Declarations	440
D.12	Method Overloading	442
D.13	Wrap-Up	445

E Arrays and ArrayLists 452

E.1	Introduction	453
E.2	Arrays	453
E.3	Declaring and Creating Arrays	454
E.4	Examples Using Arrays	455
E.5	Case Study: Card Shuffling and Dealing Simulation	464
E.6	Enhanced for Statement	468
E.7	Passing Arrays to Methods	469
E.8	Case Study: Class GradeBook Using an Array to Store Grades	473
E.9	Multidimensional Arrays	478
E.10	Case Study: Class GradeBook Using a Two-Dimensional Array	482
E.11	Class Arrays	488
E.12	Introduction to Collections and Class ArrayList	490
E.13	Wrap-Up	493

F Classes and Objects: A Deeper Look 498

F.1	Introduction	499
F.2	Time Class Case Study	499
F.3	Controlling Access to Members	503
F.4	Referring to the Current Object's Members with the this Reference	504
F.5	Time Class Case Study: Overloaded Constructors	506
F.6	Default and No-Argument Constructors	512
F.7	Composition	513
F.8	Enumerations	516
F.9	Garbage Collection	518
F.10	static Class Members	519
F.11	final Instance Variables	523
F.12	Packages	523
F.13	Package Access	524
F.14	Wrap-Up	524

G Object-Oriented Programming: Inheritance and Polymorphism 527

G.1	Introduction to Inheritance	528
G.2	Superclasses and Subclasses	529
G.3	protected Members	530

G.4	Relationship between Superclasses and Subclasses	531
G.4.1	Creating and Using a <code>CommissionEmployee</code> Class	531
G.4.2	Creating and Using a <code>BasePlusCommissionEmployee</code> Class	536
G.4.3	Creating a <code>CommissionEmployee–BasePlusCommissionEmployee</code> Inheritance Hierarchy	541
G.4.4	<code>CommissionEmployee–BasePlusCommissionEmployee</code> Inheritance Hierarchy Using <code>protected</code> Instance Variables	544
G.4.5	<code>CommissionEmployee–BasePlusCommissionEmployee</code> Inheritance Hierarchy Using <code>private</code> Instance Variables	547
G.5	Class Object	552
G.6	Introduction to Polymorphism	553
G.7	Polymorphism: An Example	554
G.8	Demonstrating Polymorphic Behavior	555
G.9	Abstract Classes and Methods	558
G.10	Case Study: Payroll System Using Polymorphism	559
G.10.1	Abstract Superclass <code>Employee</code>	560
G.10.2	Concrete Subclass <code>SalariedEmployee</code>	563
G.10.3	Concrete Subclass <code>HourlyEmployee</code>	565
G.10.4	Concrete Subclass <code>CommissionEmployee</code>	566
G.10.5	Indirect Concrete Subclass <code>BasePlusCommissionEmployee</code>	568
G.10.6	Polymorphic Processing, Operator <code>instanceof</code> and Downcasting	569
G.10.7	Summary of the Allowed Assignments Between Superclass and Subclass Variables	574
G.11	<code>final</code> Methods and Classes	575
G.12	Case Study: Creating and Using Interfaces	576
G.12.1	Developing a Payable Hierarchy	577
G.12.2	Interface <code>Payable</code>	578
G.12.3	Class <code>Invoice</code>	579
G.12.4	Modifying Class <code>Employee</code> to Implement Interface <code>Payable</code>	581
G.12.5	Modifying Class <code>SalariedEmployee</code> for Use in the <code>Payable</code> Hierarchy	583
G.12.6	Using Interface <code>Payable</code> to Process Invoices and Employees Polymorphically	585
G.13	Common Interfaces of the Java API	586
G.14	Wrap-Up	587

H Exception Handling: A Deeper Look **591**

H.1	Introduction	592
H.2	Example: Divide by Zero without Exception Handling	592
H.3	Example: Handling <code>ArithmeticExceptions</code> and <code>InputMismatchExceptions</code>	594
H.4	When to Use Exception Handling	599
H.5	Java Exception Hierarchy	599
H.6	<code>finally</code> Block	602
H.7	Stack Unwinding and Obtaining Information from an Exception Object	606
H.8	Wrap-Up	609

I	GUI Components and Event Handling	612
I.1	Introduction	613
I.2	Nimbus Look-and-Feel	613
I.3	Text Fields and an Introduction to Event Handling with Nested Classes	614
I.4	Common GUI Event Types and Listener Interfaces	620
I.5	How Event Handling Works	621
I.6	JButton	623
I.7	JComboBox; Using an Anonymous Inner Class for Event Handling	627
I.8	Adapter Classes	630
I.9	Wrap-Up	631
J	Other Java Topics	632
J.1	Introduction	633
J.2	Collections Overview	633
J.3	Type-Wrapper Classes for Primitive Types	634
J.4	Interface Collection and Class Collections	634
J.5	Lists	635
	J.5.1 ArrayList and Iterator	635
	J.5.2 LinkedList	637
	J.5.3 Views into Collections and Arrays Method asList	640
J.6	Collections Methods	642
	J.6.1 Method sort	642
	J.6.2 Method shuffle	644
J.7	Interface Queue	645
J.8	Sets	645
J.9	Maps	646
J.10	Introduction to Files and Streams	649
J.11	Class File	650
J.12	Introduction to Object Serialization	651
J.13	Introduction to Multithreading	652
J.14	Creating and Executing Threads with the Executor Framework	653
J.15	Overview of Thread Synchronization	657
J.16	Concurrent Collections Overview	658
J.17	Multithreading with GUI	658
J.18	Wrap-Up	665
K	Operator Precedence Chart	668
L	Primitive Types	670
	Index	671