



Beginning Android

Get started building apps for the
Android platform

—

Fifth Edition

—

Grant Allen



Apress®

Beginning Android

Fifth Edition



Grant Allen

Apress®

Beginning Android

Copyright © 2015 by Grant Allen

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed. Exempted from this legal reservation are brief excerpts in connection with reviews or scholarly analysis or material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use by the purchaser of the work. Duplication of this publication or parts thereof is permitted only under the provisions of the Copyright Law of the Publisher's location, in its current version, and permission for use must always be obtained from Springer. Permissions for use may be obtained through RightsLink at the Copyright Clearance Center. Violations are liable to prosecution under the respective Copyright Law.

ISBN-13 (pbk): 978-1-4302-4686-2

ISBN-13 (electronic): 978-1-4302-4687-9

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The images of the Android Robot (01/Android Robot) are reproduced from work created and shared by Google and used according to terms described in the Creative Commons 3.0 Attribution License. Android and all Android and Google-based marks are trademarks or registered trademarks of Google Inc. in the United States and other countries. Apress Media LLC is not affiliated with Google Inc., and this book was written without endorsement from Google Inc.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Managing Director: Welmoed Spahr

Lead Editor: Jonathan Gennick

Technical Reviewer: Jon Westfall

Editorial Board: Steve Anglin, Louise Corrigan, Jim DeWolf, Jonathan Gennick, Robert Hutchinson,

Michelle Lowman, James Markham, Susan McDermott, Matthew Moodie, Jeffrey Pepper,

Douglas Pundick, Ben Renow-Clarke, Gwenan Spearing

Coordinating Editor: Jill Balzano

Copy Editor: Rebecca Rider

Compositor: SPi Global

Indexer: SPi Global

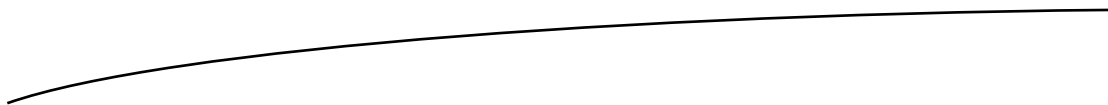
Artist: SPi Global

Distributed to the book trade worldwide by Springer Science+Business Media New York, 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springer.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a Delaware corporation.

For information on translations, please e-mail rights@apress.com, or visit www.apress.com.

Apress and friends of ED books may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Special Bulk Sales-eBook Licensing web page at www.apress.com/bulk-sales.

Any source code or other supplementary material referenced by the author in this text is available to readers at www.apress.com. For detailed information about how to locate your book's source code, go to www.apress.com/source-code/.



To all the future Android developers out there.

Contents at a Glance

About the Author	xix
About the Technical Reviewer	xxi
Acknowledgments	xxiii
Introduction	xxv
■ Part I: Get Android and Get Coding!	1
■ Chapter 1: Welcome to Android	3
■ Chapter 2: Ready, Set, Code!	7
■ Chapter 3: Inside Your First Android Project	35
■ Chapter 4: Changing and Enhancing Your Project	45
■ Part II: Activities and the User Interface	57
■ Chapter 5: Working with Android Widgets	59
■ Chapter 6: Layouts and UI Design	79
■ Chapter 7: Building Layouts with Lists and Adapters	107
■ Chapter 8: Working with Input Methods, Menus and Dialogs	127
■ Chapter 9: Adopting the Action Bar	155
■ Chapter 10: The Life of an Activity	169

- **Part III: A World of Wonderful Devices and Screens..... 179**
- **Chapter 11: Android Fragments 181**
- **Chapter 12: Intents and Receivers 197**
- **Chapter 13: Making Phone Calls 211**
- **Chapter 14: Making Noise with Audio for Android 219**
- **Chapter 15: Locations and Mapping with Android 239**

- **Part IV: Working with Resources and Services 255**
- **Chapter 16: Weaving the Web with Android 257**
- **Chapter 17: Working with Files 287**
- **Chapter 18: Managing Databases with Android 305**
- **Chapter 19: Using Preferences 327**
- **Chapter 20: Android Security and Permissions..... 343**
- **Chapter 21: Incorporating External Libraries with Android 355**
- **Chapter 22: Leveraging Android Services 371**
- **Chapter 23: Communicating with Notifications..... 385**

- Index..... 403**

Contents

About the Author	xix
About the Technical Reviewer	xxi
Acknowledgments	xxiii
Introduction	xxv
■ Part I: Get Android and Get Coding!	1
■ Chapter 1: Welcome to Android	3
The Fast Path to Learning Android Development	3
Thinking in a Smartphone Mindset	4
Computing on the Go!	4
Size Matters	4
The Interface Is More Than the Screen	4
You Are Not Alone	4
Translating Developer Dreams into Android Action	5
A Look Inside Android	5
A Look Inside This Book	6

- **Chapter 2: Ready, Set, Code!** 7
 - Getting the Prerequisites for Android 7
 - Determining the Right Java for You 7
 - Knowing (Enough) Java for Android..... 9
 - Choosing a Development Environment 9
 - Option 1: Choosing Android Studio 10
 - Option 2: Choosing Eclipse the Quick Way..... 12
 - Options 3: Choosing Eclipse the Do-It-Yourself Way..... 13
 - Completing Your Development Environment Setup 18
 - Creating Your First Android Virtual Device 20
 - Creating Your First Android Application 25
 - Writing Your First Android Code..... 31
- **Chapter 3: Inside Your First Android Project** 35
 - Looking at Android Project Structure 35
 - Starting at the Source 37
 - Benefitting from Generated Components 38
 - Using Libraries and JARs 38
 - Counting Your Assets..... 39
 - Putting Things in the Bin 39
 - Using Resources..... 39
 - Picturing What Drawables Can Do For You 39
 - User Interface Layouts..... 40
 - What’s on the Menu?..... 40
 - Valuable Values..... 41
 - Working with Other Key Files in Your Project 41
 - Controlling Applications with Android Manifests 41
 - Permutating Projects with Properties..... 43
 - “R” You Forgetting Something?..... 43
 - Solving the Appcompat Mystery 44

■ Chapter 4: Changing and Enhancing Your Project	45
Extending Your Application with the Android Manifest.....	45
Editing the Android Manifest	45
Specifying Target Screen Sizes	46
Controlling Support for Different Android Versions.....	48
Writing Actual Android Code.....	51
Introducing the Activity	52
Looking Inside Your First Activity.....	53
Building and Running Your Enhanced Application.....	54
■ Part II: Activities and the User Interface	57
■ Chapter 5: Working with Android Widgets	59
Understanding Activity Fundamentals.....	59
Naming with Labels.....	60
Pressing Buttons (redux).....	61
Handling Images with ImageView and ImageButton.....	62
Fielding Text with EditText.....	63
CheckBox? Got It.	65
Switching to the Switch	68
Working with Radio Buttons.....	71
Timing All Manner of Things with Clocks	73
Easing Input with SeekBar	75
Views: The Base of All UI Widgets	76
Useful Common Inherited Properties.....	76
Useful Common Inherited Methods	76
Even More UI Widgets	77
■ Chapter 6: Layouts and UI Design	79
Working with Relative Layouts	80
Positioning Relative to the Activity Container	80
Identifying Properties for Relative Layout.....	81
Relative Positioning Properties.....	81

- Determining the Order of Layout Relationships..... 82
- A RelativeLayout Example 82
- Overlapping Widgets in Relative Layouts..... 85
- Working with Linear Layouts..... 86**
 - Controlling the Five Key Qualifiers for LinearLayout..... 87
 - An Example LinearLayout 88
- Working with Table Layouts..... 93**
 - Understanding Widget Behavior within TableLayout..... 93
 - An Example TableLayout..... 95
- Working with Grid Layouts 97**
- Layout Manipulation with the Layout Editor 99**
 - Recap of the Layout Editor UI 100
 - Even More Reasons for XML Layouts 101
 - Converting to XML-based Layouts with Java Logic..... 101
 - Attaching Layout Definitions to Java 103
- Completing Your Revised App..... 103**
- Chapter 7: Building Layouts with Lists and Adapters..... 107**
 - Using Adapters to Populate Options 108
 - Why Adapters?..... 108
 - Starting with ArrayAdapter 108
 - Making Lists 110
 - Making More Sophisticated Lists 112
 - Adapting to the GridView..... 115
 - Taking Options for a Spin 118
 - Automatic Field Population 122
 - Summary..... 126
- Chapter 8: Working with Input Methods, Menus and Dialogs 127**
 - Using The Input Method Framework 127
 - Real and Virtual Keyboards..... 127
 - Customizing IMEs 129

Accessorizing with Soft Keyboards	133
Scrolling in to View	135
Forcing the IME out of Existence	138
Working with Menus	138
Creating Menus for your Application	139
Working With Context Menus.....	140
Menus In Action	141
Inflating Menus from XML	145
Interacting with Dialogs	148
Creating Toast Notifications.....	149
Generating Traditional Dialog Alerts.....	149
Pardon The Interruption.....	150
Summary.....	153
■ Chapter 9: Adopting the Action Bar	155
Recognizing the Action Bar	155
Managing Android Versions for Action Bar	156
Creating Action Bar Applications	157
Enabling the Action Bar for your Application	157
Moving Menu Functionality to the Action Bar	159
Using Java to Manage Action Bars	160
Working with a Standard Action Bar in Java	161
Choosing a Tab Layout for the Action Bar	163
Using Other Layouts for the Action Bar.....	167
Summary.....	167
■ Chapter 10: The Life of an Activity	169
Understanding the Activity Lifecycle	169
Working with Activity Callback Methods	171
Understanding the Goals of Android Activity Behavior	175
Working with Activity Configuration Changes	176
Understanding the Default Configuration Change Approach	176
Saving Custom State	177

- **Part III: A World of Wonderful Devices and Screens** 179
- **Chapter 11: Android Fragments** 181
 - Introducing the Fragment Class 181
 - Backwards Compatibility with Fragments 182
 - Designing with Fragments in Mind 182
 - Introducing the Fragment Lifecycle 184
 - Understanding Fragment Lifecycle Callback Methods 184
 - Keeping Sane with Fragment Lifecycles 187
 - Implementing Your Own Fragment-Based Application 187
 - Creating Layouts for the Fragments Examples 187
 - Controlling Which Layout Is Chosen 189
 - Coding Differences for Fragments 190
 - Seeing Differing Fragment Behavior in Action 191
 - Summary 195
- **Chapter 12: Intents and Receivers** 197
 - Introducing Intents 197
 - The Anatomy of an Intent 198
 - Intent Actions for all Kinds of Purposes 199
 - Understanding Intent Routing 200
 - Adding Intents to Your Manifest 200
 - Seeing Intent-launched Activities in Action 202
 - Deciding on Activity Dependency 202
 - Creating an Intent 203
 - Starting the Intent-Invoked Activity 204
 - Introducing Receivers 208
 - Using Receivers When No UI is Needed 209
 - Navigating Receiver Limitations 209
- **Chapter 13: Making Phone Calls** 211
 - Checking for Phones and “Phonies” 211
 - Mandating Telephony Support 211
 - Optional Telephony Support 212

Making Outbound Calls	212
Using ACTION_DIAL.....	217
Using ACTION_CALL.....	217
Working with Incoming Calls.....	217
■ Chapter 14: Making Noise with Audio for Android	219
Introducing the Media package.....	219
Building Your First Audio Application.....	219
Using Resources or Assets	220
Coding for Audio Playback Using the Media Framework.....	221
Starting With A Simple Audio Player Layout	222
Coding the AudioExample Behavior	223
Building Your Own Android Music Streaming Application	226
Synchronous versus Asynchronous Playback	230
Playing at Last	230
Alternative Audio Playback Approaches	231
Using SoundPool for Simultaneous Playback	231
Going Fully-Asynchronous with AsyncPlayer.....	232
Recording Audio with Android	232
Making Recordings with MediaRecorder.....	232
■ Chapter 15: Locations and Mapping with Android	239
Choosing Map Providers for Your Application.....	239
Google Maps	240
The OpenStreetMap Project.....	241
HERE/Ovi Maps	241
Which Technology to Choose?	241
Preparing for Google Maps Development.....	242
Creating Your API Project	242
Enabling a Maps API on Your Google API Project	242
Getting the API Key	243
Retrieving Your API Key.....	243

Building Your First Maps Application	244
Using MapActivity or FragmentActivity with MapView	244
Permissions and Manifest Settings for Maps	245
Building the Layout for Your Map Application	246
Adding Code for a Basic Map	246
Adding Flair and Features to Your Maps.....	250
Zoomin'	250
True North.....	250
Toolbars and the My Location Button	250
Shakin' It!	251
Changing Map Type	251
Listeners for Every Conceivable Type of Map Interaction	252
Even More Map Options.....	252
Summary	254
■ Part IV: Working with Resources and Services	255
■ Chapter 16: Weaving the Web with Android	257
Working with the WebView Widget.....	257
Creating a Simple Application with a WebView	258
Simple and Not-So-Simple WebView Security	259
Building HTML Layouts with WebView	262
UI-less Internet Interaction with Android.....	265
Why Multiple Approaches to Web Interaction?	265
Building a JSON-based Application with HttpURLConnection	267
Troubleshooting Issues with Internet Connections.....	272
Mastering Downloads with DownloadManager.....	274
Introducing the DownloadManager System Service.....	274
Selecting Permissions for DownloadManager Use.....	275
A Simple Downloader Example.....	276
Coding DownloadManager Behavior.....	277
Dealing with Download Problems.....	284

■ Chapter 17: Working with Files	287
Using Raw and Asset-based Files	287
Populating Lists from Resource Files	288
Working with Files from the File System.....	292
Android's File System Model	292
Permissions for Reading and Writing Files.....	293
Examining External Files in Action.....	295
Seeing File Save and Read Behavior in Action	299
Ensuring External Storage Is Available When Needed	301
Other Considerations with Android File Systems	302
Android Filesystems Throughout History.....	302
Controlling UI Delays with File I/O	303
■ Chapter 18: Managing Databases with Android	305
SQLite: A Seriously Popular Database	305
A Crash Course in SQLite.....	306
Creating SQLite Databases for Your Application	307
Introducing the SQLiteExample Application.....	308
Building a Database-Driven Activity	312
Choosing Query Approaches for SQLite and Android.....	315
Managing Data Results with Cursors	317
Modifying Data with Your Android Application.....	319
Inserting Data	319
Updating Data.....	320
Deleting Data.....	321
Tracking SQLite Versions and Android Versions	321
Packaging and Managing SQLite Databases for Android	323
Managing Performance and Android Storage.....	323
Packaging Your Own SQLite Database with Your Application	324
Preparing SQLite Databases for Packaging.....	324

- Chapter 19: Using Preferences 327**
 - Everything Old Is New Again 327
 - Starting with Preferences 328
 - Recording Preferences 329
 - Working with Preference-Specific Activities and Fragments 329
 - Seeing the Big Picture with Preferences 330
 - Using Preference and Preference Header Resources 331
 - Filling in the Blanks with PreferenceFragment 336
 - Bringing the Preference Puzzle Pieces Together 338
 - Other Preference Considerations 340
 - Customizing Preferences 340
 - Nesting Preferences and Display Quirks 341
 - Using the Old-School Approach to Preferences 341
 - Losing Headers and Fragments 341
 - Adapting PreferenceActivity for Older Behavior 342

- Chapter 20: Android Security and Permissions 343**
 - Requesting Permissions for Your Application 343
 - Dealing with Debugging Permissions 346
 - Installing Applications and Accepting Permissions 346
 - Pre-Marshmallow Behavior 346
 - Marshmallow and Beyond 347
 - Trading In Permissions 348
 - Creating Custom Permissions 349
 - Declaring Custom Permissions 349
 - Enforcing Custom Permissions 350
 - Securing Applications for Publication and Execution 351
 - Securing Your Application with Certificate Signing 351
 - Signing Is (Almost) Forever, So Take Care 353
 - Protecting Android and Applications at Runtime 353

■ Chapter 21: Incorporating External Libraries with Android	355
The ART of Android	355
Choosing Library Sources or JARs	356
Direct Source Inclusion.....	356
Incorporating JARs	357
When Is Java not Java?.....	358
Absent JVM Features.....	358
Considering Java Performance for Android	358
Adapting an Application to Use an External Library	360
Sourcing Apache Commons IOUtils	360
Adding the JAR to Your Project's Libraries	361
Referencing (or Refactoring for) Your External Libraries	365
Using Languages Beyond Java.....	368
Scripting Layer for Android Overview	368
Choosing Your Approach to SL4A.....	368
Testing SL4A on Your Emulator.....	369
■ Chapter 22: Leveraging Android Services	371
Services Background	371
Defining Your Own Service	372
Implementing Your Own Service Class	372
Controlling Service Lifecycle via Callbacks	373
Adding Manifest Entries for Your Service	373
Service Communication	374
Client-to-Service Communication	374
Service-to-Client Communication	377
Services in Action.....	380
Choosing the Service Design.....	380
Implementing the Java Logic for Our Service.....	380
Creating an Example Client for the Service	382
Testing the Service in Action	384

- Chapter 23: Communicating with Notifications..... 385**
 - Configuring Notifications..... 385
 - Customizing the Notification Object 386
 - Adding Icons for Notifications 388
 - Notifications in Action 389
 - Coding Notification Behavior 390
 - Notifications as the User Experiences Them 392
 - Using the Builder Approach to Notifications..... 396
 - Other Uses and Extensions to Notifications..... 397
 - Notifications and Services..... 397
 - More Advanced Notification Features 398
 - Timeline Notifications 398
 - Bundled Notifications 398
 - Expanding Notifications..... 400
 - More Embedded Actions in Notifications..... 400
 - Notifications Tailored for Wear..... 401
- Index..... 403**

About the Author

Grant Allen has worked in the information technology field for over 20 years as a chief technology officer, entrepreneur, enterprise architect, and data management expert. Grant's roles have taken him around the world, specializing in global-scale systems design, together with mentoring and coaching startups and hi-tech companies like Google on building great teams and great technology. He is a frequent speaker on topics such as big data, mobile ecosystems, Android, wearables, the Internet of Things, disruptive innovation, and more. Grant has a PhD in building innovative high-technology environments and is the author of six books on mobile development with Android and data management. You can learn more about all his work at www.grantxallen.com.

About the Technical Reviewer



Jon Westfall, PhD, is an assistant professor of psychology, researcher, programmer, and speaker. He has worked as a consultant since 1997, founding his own firm, Bug Jr. Systems. As a consultant he has developed custom software solutions (including native Windows 32 applications, Windows .NET applications, Windows Phone 7 and Android mobile applications, as well as ASP, ASP.NET, and PHP web applications). He has also served as a senior network and systems architect and administrator (on both Windows and Unix networks, and hybrids) and has also been recognized as a Microsoft Most Valuable Professional (MVP) 2008–2012. He has authored several books and presented at academic as well as technology conferences and gatherings. He can be found writing on his blog at JonWestfall.com, and on twitter [@jonwestfall](https://twitter.com/jonwestfall).

Acknowledgments

I'd like to thank all the great folks at Apress for helping get this newest edition of *Beginning Android* into the hands of you, the reader. In particular, Jonathan Gennick and Jill Balzano have been an amazing support team, encouraging, cajoling, and pushing me along the way.

I will also take this opportunity to thank all those Android developers who have encouraged me at countless conferences and events around the world. It is great to hear how books like *Beginning Android* help budding developers get started.

Introduction

Welcome to *Beginning Android, Fifth Edition*. In this fifth edition, I have completely rewritten the book and have taken stock of all the great changes that have happened to the Android platform with the releases of version 5.0 Lollipop and version 6.0 Marshmallow.

Tackling the latest and greatest aspects of Android will become second nature once you explore this book, but more importantly, you will also have a rock-solid foundation into the fundamentals of Android that span all its versions. You will explore and learn about activities, fragments, all of the user interface and user experience elements, services, data management, web services, and many, many more building blocks for your Android applications.

The best judge of this book's value is the great applications you can create from the knowledge with which it provides you. I look forward to trying all of the great applications that will flow from the knowledge you take from this book. You can explore more about the book itself at the website, www.beginningandroid.org (that's .org, not .com), and contact me via that site or my personal site at www.grantxallen.com.

All the best with your Android development!

—Grant Allen
November 2015

Part **I**

Get Android and Get Coding!

Chapter 1

Welcome to Android

I would like to welcome you to this book, *Beginning Android, 5th Edition*. This is a new edition of the book, but I'm sure this is not the first time you have encountered Android—either by using one of the billion or more Android devices currently activated throughout the world, or by hearing about it from friends, colleagues, or the media. It is hard to escape Android; not only is it the world's most prevalent mobile device operating system, but it keeps growing. Google reports that it sees more than one million Android devices activated every day.

If the growth doesn't excite you, then the diversity of Android devices should. We all know Android was developed for phones, and later, for tablets. But did you know that it now powers everything from inflight entertainment systems, to TVs, headsets, glasses, watches, shoes, games consoles, and even a brand of Japanese toilet? As Android grows, so does people's need for great applications they use on their devices. And that is where this book helps. As a new Android developer, it will be natural for you to use this book in conjunction with online searches, and I encourage you to do that. Be aware, however, that because Android has rapidly changed over the last eight years, the material you find online can be of varying quality and applicability. Sites like Stack Overflow are excellent resources, but it's prudent to check for what version of Android (or in what year) their content was written.

The Fast Path to Learning Android Development

Given the plethora of devices, you might be left wondering where to start. Too many choices can lead to indecision, and I want to get you, the reader, working with Android as quickly as possible. To help you make a rapid start, this book largely focuses on the original and most common device type on which Android runs: the smartphone. I extend that to also include tablets, since they have matured to be mainstream devices and the line is blurred between phones and tablets in any case—thus the current “Phablet” trend.

This focus on phones helps you in lots of ways right from the start. You will be learning skills and approaches for the largest device market in history, so your applications will have the biggest possible audience. And you will also be learning the largest, richest, and most

mature parts of Android development. Don't worry, though! I also cover the leading edge of Android later in the book, so you can also start mastering topics like Google Glass and Android Wear.

Thinking in a Smartphone Mindset

As a reader of this book, you might well be new to Android development, but your background as a developer in other areas, your technology expertise, and your know-how is harder to predict. Before we delve into your first look at Android code and the building blocks of an Android device, it is useful to start with some understanding of the fundamentals common to all Android development on smartphones and tablets.

Computing on the Go!

The first, and most obvious, point to make about thinking like an Android developer is that Android devices may have their roots in mobile phones, but today, actually making and receiving phone calls is only a minor part of their popularity and pervasiveness. You should think of the billion or more Android devices in circulation as little engines of mobile computing that just happen to be able to make phone calls as one of their many capabilities. They are literally mobile computers!

Size Matters

In developing for Android, you are dealing with screens of many sizes, but one thing in common for the vast, vast majority of mobile Android devices is that you are working with small screens or displays. Although you may develop for Android with the comfort of a full-sized desktop on a modern notebook computer, your target audience, at the extreme, will have only a square inch or so of display real estate on which to enjoy your development efforts.

The Interface Is More Than the Screen

Modern smartphones are more than just a screen for viewing text and other content. Phones make calls, play music, vibrate and provide haptic feedback for games, sense the temperature, report direction and location, and much more. When you think of the way in which your users might interact with your applications, think beyond the simple visual screen. A phone interacts with multiple senses, and you should design with all of them in mind.

You Are Not Alone

Yours will not be the only application running on someone's Android device. In fact, they may not just have other applications, but other applications that can do similar—or even the same—work as your own. Be mindful that you cannot hog all of the device and all of its features. Also keep in mind that for your users, it is two simple taps of a finger for them to switch to another application.

In later chapters, I also cover how resources and capabilities are shared and managed across an Android device, and how being aware that your application can be interrupted at any time is one of the key use cases you need to consider when developing Android applications.

Translating Developer Dreams into Android Action

Now that you know how to think about developing for Android, you might also be thinking that it sounds like a lot of compromises, and is that all there is to it? The good news is that by taking care of a bunch of heavy lifting for you and by providing you with an amazing array of support, tools, and the foundation on which to build applications of all sorts, Android lets you focus on the essentials of mobile development.

First, Android’s dominant programming language is Java. That’s good, because it is one of the most widely used languages on the planet. When you are developing, you get to use the vast majority of standard Java, as well as additional libraries the community has added over the years since Android’s public release.

Second, you have access to a range of development tools that are totally free! Yes, free. Google provides not one, but two, free integrated development environments (IDEs) from which to choose in which you can let your developer imagination go wild. We discuss both of these in more detail in Chapter 2, but for now, you can rest easy knowing you can work with either or both—Android Developer Tools (ADT) in Eclipse, or Android Studio, which is an IntelliJ IDEA-based IDE.

Third, many additional tools—such as emulators, performance monitors, templates, design tools, runtime environments, and more—have been added to the developer’s world. And that’s not all: you can write an entire book on this! Joking aside, Android does ask you to adopt a different development mindset to the one you need to design traditional desktop applications, but then it makes an enormous effort to support you every step of the way.

A Look Inside Android

To finish your introduction to thinking like an Android developer, let me introduce the fundamental building blocks of Android applications. I cover each of these areas in much more detail in the coming chapters.

Activities are the fundamental “screens” or UI elements with which users interact. Because of the nature of Android and the many display sizes with which you might work, a user might see one or more activities as part of the user interface at any one time. Activities are easy for you to design and develop, and they are easy for users to launch and use. Your target Android systems love activities, so feel free to be prolific in your creation and use of them!

Intents are messages passed between applications and the Android environment and are akin to events-driven messaging in modern desktop development environments. You can write your applications so they listen out for events and respond appropriately—whether that’s answering a phone call, reacting to information from onboard sensors, or responding to changes in application state. Even better, you can create and pass your own intents, expanding the capabilities of your applications *and others* on the same device.

Services are background applications that provide features, abilities, or actions on which your applications (and their activities) can rely. Much like services or daemons in other operating systems, Android services are long-lived and generally run without a visible user interface (UI). Some services do provide visual elements, and you'll learn more about services in Chapter [22](#).

Content providers are a convenient abstraction to the many types and sources of data you might want to use in your application. In traditional development environments, you have to worry about all the tedious logistics of working with files, databases, network sockets, protocols, and the like. Android vastly simplifies your life as a developer by supporting the content provider metaphor for any data that needs to be shared on the device (and even for off-device data). When building your own applications, you are also able to define and develop your own content providers to enable data sharing with other applications, without needing to work with low-level storage issues, proprietary query languages, odd protocol considerations, or any other baggage.

A Look Inside This Book

With your mind now surging with thoughts of Android development, here's the roadmap of how we'll explore all of the topics already mentioned, and more.

- The next three chapters complete Part I of the book; you'll code your first Android application in the very next chapter! No delays, no excuses. After you write your first Android application, we'll then pull it apart to ensure you are familiar with the real-world code that represents the concepts I've already described here. We'll also tinker with and explore what Android is doing for you, and what you need to be doing yourself as a nascent Android developer.
- Part II helps you master the world of activities and the possibilities for user interface magic with Android. I cover all the essential components at your disposal for interface design and behavior and how mobile-specific concepts like device rotation work.
- Part III expands on your new knowledge of interface design for Android and introduces you to the power and flexibility of *fragments*. Fragments allow you to work with devices of any size, from 1 inch to 100 inches! As well as displays, we also explore input devices such as the camera, the microphone, and other onboard capabilities.
- Part IV delves behind the scenes, building your knowledge of services, resource handling, and other powerful components of Android that you will want to use to make the best possible applications.
- Part V, the final part of the book, takes you the frontiers of Android and looks at a range of topics, such as Google Play services, cloud-based facilities you can incorporate into your application, and the explosive growth in wearable Android with Google Glass and Android Wear.

Now that you have had a proper introduction to Android development and know where you are going, it's time to get started actually developing your first Android application!

Ready, Set, Code!

There is no time like the present to get started with your very first Android application. That's right, you are going to start coding right now! This chapter takes you right through setting up the tools to create Android applications, and you'll have your first working Android application before this chapter is done. Ready? Let's code!

Getting the Prerequisites for Android

There are two main prerequisites to quickly consider for your forthcoming Android development journey: will you need to separately install Java, and do you know what I call “minimum viable Java” for the development you might like to do?

Determining the Right Java for You

In a moment, I discuss the two most popular ways to set up a development environment for Android applications: preconfigured Eclipse and Android Developer Tools (ADT), and Android Studio. Both of these options give you everything you need bundled together, including Java, so you can start coding straight away. But if you are like me, you want more than two options; you might also like to learn about how the tools are set up and how to use them to create Android apps.

If you want to take full control of setting up the Android developer components and not use one of the all-in-one bundles, or if you have an existing development environment in mind and don't want to use one of the options provided by Google, then you need to ensure you have Java set up and available on your machine.

Getting the JDK

The Java Development Kit (JDK) has always been readily accessible, first through Sun and now through Oracle. If you point your browser to java.oracle.com, Oracle redirects you to the latest-and-greatest landing page for getting the JDK. But be warned! When it comes to getting Java for Android development, you need to make sure you choose a supported

version of Java. For now, this means avoiding any Java 8 release. Regardless of platform—Windows, Linux, Mac—you want to choose either Java 6 or Java 7 (also known as Java 1.6 and Java 1.7—confusing, eh?).

Before you dive straight in to picking a JDK version, it is worth noting that, over the years, Java has been a very large source of security vulnerabilities and has been notoriously difficult to keep patched. This is thanks to repeated changes of ownership, changes in approach to managing Java on operating systems of all kinds, and the tendency of various software companies to bundle versions of Java with products that they then never update. You can't escape the need to install the JDK because the Java language is the chosen development language for Android, but you should at least be aware of the added security implications of having it on your development machine.

Android has supported Java 6 since its relatively early versions and has officially supported Java 7 since the KitKat version 4.2 release (with a few very minor no-go areas, such as `try-with-resources`). If you are setting up an entirely new environment, I recommend that you choose Java 7 for your platform because doing so will future-proof your Android development environment to the greatest degree possible.

Note When Android was first released, Java 5 was supported, but that version is best left to the history books.

Because a few of the tools bundled with the Android development environments from Google still only come in 32-bit form, you should choose the 32-bit distribution of Java 7 for your system if you want to get maximum benefit. For Mac users, however, only a 64-bit JDK is available, which means a handful of tools are not available to you, though none of those feature in this book. For the purposes of this book, you only need the standard edition (SE) version of the JDK. At the time of writing, the latest security patch release for Java 7 was Java SE 7u71/72.

To download your version of the JDK, follow the links on the Oracle download page, and choose the correct JDK distribution for your platform. For Linux users, this is the file `jdk-7u71-linux-i586.rpm`; for Windows, it is `jdk-7u71-windows-i586.exe`; and for Mac, it is `jdk-7u71-macosx-x64.dmg`. Make very sure it is the JDK you are downloading, and not myriad other pieces of software that are mixed in to the Oracle download pages. You do not want just the Java Runtime Environment (JRE), nor are you after OpenOffice, Oracle database software, or the other pieces you see on the quite-confusing download page.

ALTERNATIVES TO THE OFFICIAL JAVA RELEASE

Astute Linux fans will note that Oracle only provides an RPM distribution for Java. If you are a user of Debian, Ubuntu, Mint, or some other flavor of Linux that doesn't use RPM for package management, you do have options. Although it is not officially supported by Oracle, the OpenJDK distribution of Java works well on Debian and its derivatives. Using this distribution helps you avoid any complications you'll encounter if you try to convert the official RPM using a tool like `alien`.

Once you have downloaded the relevant JDK bundle, you are ready to install.

Installing the JDK

Follow the instructions included in your Java bundle to install the JDK appropriately. On some systems, you might need administrative privileges to make your Java installation available system wide.

Knowing (Enough) Java for Android

This book assumes you have a passing knowledge of coding in Java, as do most other books and resources on Android development. If you are new to Android and new to Java, then I recommend a crash course in the basics of Java—such as that offered by the book *Learn Java for Android Development* by my fellow Apress author, Jeff Friesen (Apress, 2010). Alternatively, countless online Java tutorials can get you up to speed quickly. The secret is knowing on which topics you need to focus to be productive with Android development and which you can ignore for now.

To help focus your learning, or refresh your memory if it has been a while since you last wrote Java code, the following are the key topics that will get you (re)started with Java.

General coding knowledge:

- Structure and layout of code
- Object-orientation, classes, and objects
- Methods, data members

Java-specific coding knowledge:

- Interfaces and implementations
- Threading and concurrency
- Garbage collection
- Exception handling
- File handling
- Generics
- Collections

There is plenty more to Java than these topics alone, but they provide the foundation you need to get started quickly with Android development.

Choosing a Development Environment

In the years since Android's initial release, the tools and technology available to help build applications have exploded in variety and capability. Google provides several options, which I explore shortly, and many other companies and groups also offer development tools. I cover a few of those more exotic options much later in this book. For now, I stick to the most common environments you are likely to use: Android Studio and Eclipse.