



# Android for Absolute Beginners

Getting Started with Mobile Apps  
Development Using the Android Java SDK

---

Grant Allen

apress®

# Android for Absolute Beginners

Getting Started with Mobile Apps  
Development Using the Android Java SDK



Grant Allen

Apress®

# ***Android for Absolute Beginners: Getting Started with Mobile Apps Development Using the Android Java SDK***

Grant Allen  
LONDON, UK

ISBN-13 (pbk): 978-1-4842-6645-8  
<https://doi.org/10.1007/978-1-4842-6646-5>

ISBN-13 (electronic): 978-1-4842-6646-5

Copyright © 2021 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.

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 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, Apress Media LLC: Welmoed Spahr  
Acquisitions Editor: Steve Anglin  
Development Editor: Matthew Moodie  
Coordinating Editor: Mark Powers

Cover designed by eStudioCalamar

Cover image by Dollar Gill on Unsplash ([www.unsplash.com](http://www.unsplash.com))

Distributed to the book trade worldwide by Apress Media, LLC, 1 New York Plaza, New York, NY 10004, U.S.A. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail [orders-ny@springer-sbm.com](mailto:orders-ny@springer-sbm.com), or visit [www.springeronline.com](http://www.springeronline.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 [booktranslations@springernature.com](mailto:booktranslations@springernature.com); for reprint, paperback, or audio rights, please e-mail [bookpermissions@springernature.com](mailto:bookpermissions@springernature.com).

Apress titles 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 Print and eBook Bulk Sales web page at <http://www.apress.com/bulk-sales>.

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub via the book's product page, located at [www.apress.com/978-1-4842-6645-8](http://www.apress.com/978-1-4842-6645-8). For more detailed information, please visit <http://www.apress.com/source-code>.

Printed on acid-free paper



*To all the aspiring new Android developers in the world,  
may you have fun while you learn to build new and wonderful applications!*

---

# Table of Contents

About the Author .....xv

About the Technical Reviewer .....xvii

Acknowledgments .....xix

Introduction .....xxi

■ **Part I: Get Android, Get Started ..... 1**

■ **Chapter 1: Introducing Android ..... 3**

    Working with Android: The Best Parts ..... 3

    Working with Android: The Challenges..... 5

    Understanding Android’s Heritage and How It Affects You ..... 6

    Understanding Android’s Future ..... 9

    What to Expect for the Rest of This Book ..... 9

        Detailing More of the Remaining Chapters..... 9

■ **Chapter 2: Introducing Android Studio ..... 13**

    Understanding What Is Meant by Integrated Development Environment, or IDE..... 13

    The History and Provenance of Android Studio ..... 14

        Downloading the Android Studio Installer for Your Platform..... 14

        Working with Android Studio Versions..... 15

        Understanding How Android Studio and the Android SDK Work Together ..... 16

Preparing for Android Studio .....	16
Choosing Desktop or Laptop Hardware for Android Development.....	16
Choosing the Operating System for Your Computer .....	17
Continuing with Android Studio Setup After Installation.....	20
Working with Alternatives to Android Studio.....	30
Eclipse .....	31
IntelliJ IDEA .....	31
Tools That Target Multiple Operating Systems and Mobile Platforms.....	32
Traditional Platform-Agnostic Development Tools .....	32
■ <b>Chapter 3: Your First Android Application, Already!.....</b>	<b>33</b>
Creating Your First Android Virtual Device .....	33
Creating Your First Android Application, Already! .....	41
Getting Your First Code Written.....	46
Getting Ready to Run Your Application .....	48
Installing (Additional) SDK Packages.....	50
Running Your Application.....	52
■ <b>Chapter 4: Exploring Your First Project .....</b>	<b>55</b>
Viewing the Whole Android Project Structure.....	55
Working with Manifests.....	56
Jiving with Java.....	58
Getting Resourceful with Resources .....	58
Building Everything with Gradle Files.....	60
Explaining the Key Android Application Logical Building Blocks .....	63
Activities .....	63
Intents.....	63
Services.....	63
Content Providers .....	64
Summary .....	64

---

■ <b>Chapter 5: Android Studio In-Depth.....</b>	<b>65</b>
Starting with the Project Explorer .....	65
Getting Comfortable Switching Project Explorer Views .....	66
Using the Project Explorer Context Menu .....	68
Running and Debugging with Android Studio.....	68
Running with Run Configs: A Recap .....	68
Running Farther with AVDs and Connected Devices .....	70
Debugging Instead of Running Your Code .....	77
Setting and Clearing Breakpoints .....	77
Starting Applications for Debugging.....	78
Stepping Through Code While Debugging .....	80
Attaching the Debugger.....	80
Viewing What You’ve Run .....	81
Understanding Your Build .....	81
Understanding Events in the Event Log.....	82
Understanding Logcat.....	83
Revisiting the SDK Manager.....	84
Highlighting the Other Major Features of Android Studio.....	86
Summary .....	86
■ <b>Chapter 6: Mastering Your Entire Developer Ecosystem .....</b>	<b>87</b>
Choosing Dedicated Developer Hardware .....	87
Determining Which CPU Is Right for You.....	88
Too Much Memory Is Never Enough! .....	88
Storing It All! .....	89
Viewing It All!.....	90
Connecting It All! .....	93
Test Handsets and Tablets.....	93
Option 1: All-Virtual Testing.....	94
Option 2: Start Virtual and Augment with Top Devices.....	94
Option 3: Buy, Buy, and Buy Some More .....	94
Option 4: Other People’s Hardware.....	95

Building with Build Tools .....	96
Update Your gradle.properties File .....	96
Use the Gradle Command Line .....	97
Visit gradle.org to Learn More .....	98
Managing Your Code with Source Control .....	98
Completing Your Software Repertoire .....	99
Summary .....	101
<b>■ Part II: Get the Android Development Know-how .....</b>	<b>103</b>
<b>■ Chapter 7: Introducing Java for Android Development .....</b>	<b>105</b>
Java, Java, Everywhere .....	106
Android's Java Time Warp .....	107
Using a JDK Installation for Learning Java .....	107
Your First Java Concepts in Code .....	109
The Key Java Building Blocks for Android Development .....	112
Structure of Code .....	113
Understanding Object-Oriented Design, Classes, and Objects .....	114
Working with Class Methods and Data Members .....	114
Summary .....	116
<b>■ Chapter 8: Introducing XML for Android Development .....</b>	<b>117</b>
Getting Started with XML .....	117
XML Version and Encoding .....	118
XML Elements .....	118
XML Attributes .....	120
XML Values .....	121
XML Namespaces .....	122
Other Parts of the XML World .....	123
Using XML in Your Android Applications .....	124
Reviewing the Android Manifest XML File .....	124



---

Using XML for Resource Definitions .....	125
Defining Activity Layouts with XML .....	128
Summary .....	130
■ <b>Chapter 9: Exploring Android Concepts: Core UI Widgets .....</b>	<b>131</b>
Everything Starts with View .....	131
Key Methods Derived from View .....	132
Key Attributes and Properties Derived from View .....	132
Introducing the Core UI Widgets in Android .....	132
Labeling Things with TextView .....	133
Buttoning the Perfect UI .....	138
Getting the Picture with ImageView and ImageButton .....	142
Editing and Inputting Text with EditText .....	144
Checking Out the CheckBox .....	147
Switching It Up with a Switch .....	150
Choosing Things with Radio Buttons .....	153
Learning Even More UI Widgets .....	155
Summary .....	155
■ <b>Chapter 10: Exploring Android Concepts: Layouts and More .....</b>	<b>157</b>
What Is Android Jetpack .....	157
Using Popular Layout Designs .....	159
Revisiting the Constraint Layout .....	159
Using Relative Layouts .....	164
Lining Up with Linear Layouts .....	170
Even More Layout Options .....	176
Table Layouts .....	176
Grid Layouts .....	177
Mastering XML-Based Layouts with Java Logic: The Best of Both Worlds! .....	177
Connecting XML Layout Definition in Java Code .....	178
Revisiting MyFirstApp via ButtonAgain .....	179
Summary .....	181

■ <b>Chapter 11: Understanding Activities</b> .....	<b>183</b>
Delving into the Android Activity Lifecycle .....	183
Understanding Activity Lifecycle Callback Methods .....	185
Appreciating the Objectives of the Activity Lifecycle.....	189
Managing Activity Configuration Changes.....	190
Working with the Default Approach to Configuration Change .....	190
Summary .....	191
■ <b>Chapter 12: Introducing Fragments</b> .....	<b>193</b>
Starting with the Fragment Class.....	193
Using Fragment-Based Design for Your Applications .....	194
Working with the Fragment Lifecycle.....	195
Reviewing the Fragment Lifecycle Callback Methods.....	196
Starting Simple with Fragment Lifecycle Events.....	198
Creating a Fragment-Based Application .....	198
Selecting Which Layout to Inflate .....	200
Coding for Fragments.....	201
ColorFragmentExample in Action .....	203
Summary .....	206
■ <b>Part III: Get Sophisticated</b> .....	<b>207</b>
■ <b>Chapter 13: Working with Sound, Audio, and Music for Android</b> .....	<b>209</b>
Playing Back Audio .....	209
Choosing Your Audio Approach .....	209
Working with the Media Package.....	210
Creating an Audio Application .....	211
Choosing Between Assets and Resources.....	211
Writing the Code for Audio Playback .....	212
Playing Back Audio the Simple Way .....	213
Coding the Java Logic for AudioPlayExample.....	214

---

Using Streaming for Audio Playback .....	217
Playing the Music Stream.....	220
Exploring Other Playback Options .....	220
Recording Audio .....	221
Recording Audio with MediaRecorder .....	221
Expanding Your Developer Audio Toolset.....	227
Understanding Key Audio Aspects.....	228
Choosing Further Audio Tools .....	229
Summary .....	231
■ <b>Chapter 14: Working with Video and Movies for Android</b> .....	<b>233</b>
Playing Back Video .....	233
Designing a VideoView-Based Layout .....	234
Controlling Video Playback in Your Code .....	236
Understanding Key Video Concepts.....	239
Bitrate .....	239
Frame Rate .....	240
Codecs.....	240
Understanding the Complex World of Video Containers and Child Codecs.....	241
Expanding Your Developer Video Toolset.....	243
Reviewing the Range of Video Editing Tools Available to You .....	243
Popular Open Source Video Editing Suites .....	244
Popular Commercial Video Editing Suites.....	244
Introducing HandBrake for Video Editing.....	245
Downloading and Installing HandBrake.....	245
Running and Using HandBrake.....	245
Going Further with Video for Android.....	249
Summary.....	249

■ <b>Chapter 15: Introducing Notifications</b>	<b>251</b>
Configuring Notifications	251
Customizing Notifications with the Notification Object	252
Adding Icons for Notifications	255
Introducing Notification Channels in API Level 26	256
Notifications in Action	257
Supporting Logic to Create the Notifications	257
Seeing Notifications from the User's Perspective	260
Summary	264
■ <b>Chapter 16: Exploring Device Capabilities with Calls</b>	<b>265</b>
Calling the Shots	265
Specifying Telephony Support	265
Making Telephony Support Optional	266
Making Calls	266
Laying Out the CallExample Application	267
Working Logic for the CallExample Application	270
Making an ACTION_DIAL Call	271
Making an ACTION_CALL Call	272
Handling Incoming Calls	273
■ <b>Part IV: Get Together</b>	<b>275</b>
■ <b>Chapter 17: Understanding Intents, Events, and Receivers</b>	<b>277</b>
Introducing Android Intents	277
Understanding Intent Behavior	278
Understanding Intent Routing	280
Including Intents in Your Manifest	280
Using Android's Verified Links	282
Seeing Intent-Launched Activities in Action	282
Deciding on Activity Dependency	282
Creating an Intent	283
Starting the Intent-Invoked Activity	284

---

Introducing Receivers .....	288
Using Receivers When No UI Is Needed.....	289
Working with Receiver Limitations .....	289
Summary .....	290
■ <b>Chapter 18: Introducing Android Services .....</b>	<b>291</b>
Services Background .....	291
Using WorkManager as an Alternative to Services .....	292
Starting with Your Own Service.....	292
Implementing Your Service Class .....	292
Managing the Service Lifecycle via Callbacks.....	293
Providing Manifest Entries for Your Service .....	294
Service Communication.....	294
Service-to-Client Communication .....	297
Using bindService() Methods for All Communication .....	298
Intents and Broadcast Receivers.....	298
Use PendingIntent Objects.....	298
Use Messenger and Message Objects.....	299
Use Independent Messaging .....	299
Create Callbacks and Listeners .....	299
Use Notifications.....	300
Services in Action.....	300
Choosing the Service Design.....	300
Creating the Java Logic for the Service.....	300
Creating an Example Client to Use the Service .....	302
Testing the Service in Action .....	304
Summary .....	304
■ <b>Chapter 19: Working with Files in Android.....</b>	<b>305</b>
Using Assets and Raw Files .....	305
Populating Lists from Resource Files .....	306
Working with Files from the File System.....	310

Android's File System Model .....	310
Permissions for Reading and Writing Files.....	311
Examining External Files in Action .....	312
What Goes into Saving and Reading Files .....	315
Saving and Reading Files in Action.....	317
Ensuring External Storage Is Available When Needed.....	319
Other Considerations with Android File Systems.....	320
Summary.....	322
■ <b>Chapter 20: Working with Databases in Android.....</b>	<b>323</b>
Working with SQLite: The World's Most Popular Database! .....	323
Quickly Learning SQLite for Android Development.....	324
Creating SQLite Databases for Your Applications .....	325
Introducing the SQLiteExample Application.....	326
Creating a Database-Driven Activity .....	330
Choosing Query Approaches for SQLite and Android.....	333
Managing Results from Queries with Cursors .....	335
Modifying Data with Your Android Application.....	337
Inserting Data .....	337
Updating Data .....	339
Deleting Data .....	339
Using the Room Persistence Library .....	340
Packaging and Managing SQLite Databases for Android .....	340
Managing Android Storage for Database Performance .....	340
Packaging SQLite Databases with Your Applications.....	341
Choosing SQLite Management Tools to Prepare Databases for Packaging.....	342
Working with Built-In Tools.....	342
Working with Third-Party Database Tools.....	342
Summary.....	343
■ <b>Index.....</b>	<b>345</b>



# About the Author

**Grant Allen, PhD**, has worked in the technology field for over 20 years, as a CTO, entrepreneur, and emerging technology expert. After successful startup exits and a decade at Google, Grant's focus is now mentoring and coaching startups and hi-tech companies on building great teams and great technology. He is a frequent speaker on topics such as big data, mobile ecosystems, Android, wearables, IoT, disruptive innovation, and more. Grant has a PhD in computer science based on research he performed while at Google and an MBA specializing in technology management, and he is the author of seven books on various mobile platform and data technology topics.



# About the Technical Reviewer

**Val Okafor** is a software architect with expertise in Android development and resides in sunny San Diego of California, USA. He has over 14 years of industry experience and has worked for corporations such as Sony Electronics, The Home Depot, San Diego County, and American Council on Exercise. Val earned his BSc in IT from National University, San Diego, and master's in software engineering from Regis University, Colorado. He is the creator and principal engineer of the Pronto line of mobile apps including Pronto Journal, Pronto Invoice, and Okason Contractor App. His passion for software development goes beyond his skill and training; he also enjoys sharing his knowledge with other developers. He has taught Android development to over 5,000 students through Udemy.





# Acknowledgments

Writing a book during a global pandemic is a strange experience and only made possible with the help of all the folks at Apress and the technical reviewer. I'd like to thank Steve Anglin, Val Okafor, Mark Powers, Matthew Moodie, and all the other people at Apress who made this book a reality.

I would also like to thank Google for continuing to make Android a fun and dynamic platform on which to learn to develop applications.



# Introduction

A book for beginners with Android development should provide enough subject matter to spur the imagination and lay a foundation for future learning and experimentation. The trick is to make sure the content doesn't overload you, the reader, or leave you at your wit's end as to how to get started.

In this book, I have tried to balance giving you a firm foundation in the core concepts and techniques you will need to get started, without packing in every possible topic that might be of use to you eventually. That would have resulted in thousands and thousands more pages of content, which gets to the point of being counterproductive.

You will see and create a wide range of applications based on the topics covered in the 20 chapters presented here, and I've also provided a range of pointers to more online content at Apress, Google, and other websites that take some of the topics beyond the beginner's realm. This book is a beginning, and where you go from here is entirely up to you!



Part I

# Get Android, Get Started

---

# Chapter 1

## Introducing Android

Welcome to the start of your developer journey with Android. Perhaps you already own and use an Android phone, tablet, or other devices. You would be in great company, with over a billion devices around the world used every day based on the Android operating system. Even if you think you don't already use Android, you may be surprised to learn that it has made its way into an astonishing range of devices and products, many of which you might have used without even realizing the power of Android was helping you out.

Android now powers an amazing range of devices beyond phones and tablets, including smart watches, fitness devices, car entertainment and navigation systems, game consoles, toys, kitchen appliances, garden watering systems, plumbing and heating controls, and even barbecues! Yes, barbecues! This expansion in devices that use Android shows no signs of slowing down, but the most popular and likely place in which to find Android is still on phones and tablets. As a budding developer for Android, this means that you'll learn many aspects that apply to all – or at least many – of the types of devices mentioned here, but your initial focus is likely to be best applied learning to build applications for Android phones and tablets.

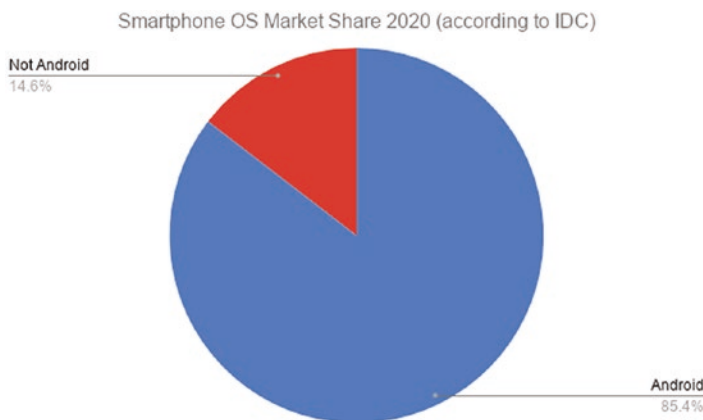
### Working with Android: The Best Parts

Mobile and smartphone development has been one of the most exciting technology stories of the last decade or more. You might not personally have experienced every twist and turn, but you are likely to have been exposed to devices across one or more of the recent eras of smartphones, and you have likely heard names like Nokia, Apple, Microsoft, Google, and Android mentioned in many combinations.

Today, Android presents you with the benefits learned from all of the great technology battles between these companies and platforms over the last 10 years and lets you ride that wave, serving up fantastic tools, capabilities, and help to get you on the path to Android application development. That is perhaps Android's most compelling appeal right now,

as this book is written. It is neck and neck at the forefront of all smartphone and mobile technology (along with iOS from Apple), and by choosing Android, you immediately gain access to the benefits it provides, which include

- Developer tools and platforms: We'll cover many of these items in this very book, from Android Studio, the Android SDK, Google Play services, and the Google Play online store, to name just a few.
- A massive existing market: I'm not joking when I say a billion-plus users are waiting for your application, with many more potential future users on the horizon. As Figure 1-1 shows, the global market penetration for Android is breathtaking.
- A global community of like-minded developers: You are not alone in pursuing your Android application development dreams. There are tens of thousands, hundreds of thousands, or possibly even millions of other developers who have experiences to share.
- A rock-solid technology foundation: Android started life using the Java programming language as the predominant technology with which to build applications. Today it also supports Kotlin and C++. We will stick to Java for the examples and techniques explored in this book, as it is one of the most widely used, widely respected, and mature development technologies in existence – the vast majority of Android applications are built with Java.



**Figure 1-1.** Android's global market share among smartphone users

I could go on with more examples of the breadth and depth of the Android development world and its advantages, but instead I'll let the rest of this book help demonstrate many more of them.

---

## Working with Android: The Challenges

I wouldn't be honest if I didn't flag that Android is blessed in many ways, but also has challenges. The good news is that many of these challenges are both well known and easily accommodated. Here are some of the areas that a new developer is likely to encounter before too long:

- You are developing for small devices. Whether we are talking Android phones, Android tablets, in-car dashboards, or a range of other form factors, one common constraint you will need to deal with is the size of the screen. We will explore this topic in depth when we discuss emulating Android devices with virtual devices, but always remember that your experience sitting at a desktop or laptop computer, building for Android, is different from the experience your users will have on the phones and tablets.
- Think beyond the screen. Don't let small screens be your only avenue for providing great features and cool capabilities in your applications! As you will see, later chapters will explore things like sound and audio, on-board sensors, vibration, and more. Don't become a captive to the screen – you have many more tools at your disposal.
- Alternatives are a single swipe away. Everyone with an Android device – or indeed any smartphone – has first-hand experience of many applications. Users have choice, and users will also be running a range of other applications on their phones. This means your application might be sharing resources and user attention, and users can do all kinds of things you never expected!
- Android isn't always Android. We will be covering the latest versions of Android in this book, including Android version 11, which at the time of writing is nearly one year old, and previewing Android version 12, which will be released later this year as part of Google's typical annual release cycle for Android updates. However, out in the marketplace, the billion or more users I mentioned earlier in this chapter are using earlier Android versions. And not just Android 10 or Android 9, but versions all the way back to Android 4 and even earlier! Android has a complicated history when it comes to encouraging (or even allowing) users to upgrade, and this is something to keep in mind when deciding how much of that billion-user market to target.

Whatever you do, do not get disheartened by these points. Think of them more as the first set of lessons to learn as you embark on developing Android applications.

## Understanding Android's Heritage and How It Affects You

Android dates back to 2003, when in October of that year a group of developers in Palo Alto got together and used the Linux kernel as the basis for their dream of a new age of devices with much more capable interfaces. After struggling in its formative years, Google bought the Android business in 2005, and its pre-Google cofounder Andy Rubin joined Google to continue the effort developing an operating system for smartphones and other gadgets.

In 2008, after a few false starts, Google and its partners HTC and T-Mobile released the first handset, known as the “Dream” or “G1,” depending on which country you were in. My very own G1 still runs to this day, but is no longer my daily go-to phone. Figure 1-2 shows my G1 at the original lock screen.



**Figure 1-2.** The author's original G1 Android phone, from 2008

After this first release, momentum slowly ramped up for Android, and updates to the operating system started appearing with the names of sweet treats as code names – Cupcake, Donut, Eclair, and so on, representing versions 1.5, 2.0, and 2.1.

---

Around the time of first public release, Google also partnered with telcos, chip companies, and handset manufacturers to create the Open Handset Alliance, which was aimed at building a broad coalition to support Android into the future. Google also moved to establish the “Android Open Source Project” as the custodian for the base open source code for Android.

There is much more to the story of Android, with many new versions of software and many new manufacturers joining in the fun. But the preceding points were the unheralded start of several of the issues that Android developers must contend with as they build applications.

With many companies moving to build Android-powered phones, Google did not directly control how Android was maintained on each manufacturer’s devices nor how telcos in major markets managed the consumer lifecycle of phones as part of contracts, sales, and so forth. What emerged over the coming years after the initial releases of Android was a market with many different devices, running many different versions of Android, with very patchy upgrade guarantees. This is known in the Android world as the fragmentation problem.

For developers, this means that building applications for Android requires some extra effort in thinking about the population of devices in the wild and what versions of Android they run. At the time of writing this book, Google has made a range of efforts to make this less of a concern and encourage manufacturers to upgrade devices. This is having some impact. Google has also added a range of developer features to ease the effort required of you, the developer, to handle this varied market, and we will discuss several of these features in later chapters.

In closing this topic, Figure 1-3 shows a view of the current distribution of Android versions in use throughout the world in mid-2020.



ANDROID PLATFORM VERSION	API LEVEL	CUMULATIVE DISTRIBUTION
4.0 Ice Cream Sandwich	15	
4.1 Jelly Bean	16	99.8%
4.2 Jelly Bean	17	99.2%
4.3 Jelly Bean	18	98.4%
4.4 KitKat	19	98.1%
5.0 Lollipop	21	94.1%
5.1 Lollipop	22	92.3%
6.0 Marshmallow	23	84.9%
7.0 Nougat	24	73.7%
7.1 Nougat	25	66.2%
8.0 Oreo	26	60.8%
8.1 Oreo	27	53.5%
9.0 Pie	28	39.5%
10. Android 10	29	8.2%

**Figure 1-3.** Android versions and API levels in use in 2020

Note the somewhat self-flattering presentation of these numbers. Instead of reporting the absolute fraction of devices in use for each version of Android, you are instead shown the “cumulative distribution” for devices running a particular version or later. It is easy enough to perform the arithmetic to work out the real absolute percentages. For example, 11.2% of devices run Android 6.0 “Marshmallow,” being the difference between its 84.9% cumulative distribution and Android 7.0’s 73.7% cumulative distribution. At the time of writing, Android 11 had been released for several months, but was not yet showing in the statistics. Android 12 is slated for release later in this current year, with only minor tweaks to Android 11. Rest assured both Android 11 and then Android 12 will rocket up the charts and be a good basis for learning Android development.

The takeaway from this discussion is that to target a meaningful percentage of Android devices, you will want to embrace Android capabilities from several versions in the past, rather than only cater to the bleeding edge.

## Understanding Android's Future

A system like Android, with billions of users already, can already be considered a roaring success. But there's plenty of growth and opportunity left for Android, in a wealth of new areas. For instance, a great deal of attention is now centered on how Android will be part of "hybrid" computing in the future. This hybrid model is where some devices use Android for part of the operation, serving up applications just as your smartphone does today, but switch to using a second operating system like Chrome OS for other tasks.

Android will also follow a regular path of updates itself, as will many of the great online services like Google Play and other cloud offerings, which will further push the boundaries of what's possible with Android devices and Android applications.

## What to Expect for the Rest of This Book

*Android for Absolute Beginners* is written to take you, the reader, on your first journey writing software, learning the practices, techniques, and approaches that will arm you for your future application development goals, as well as many of the fundamentals upon which computer programs are built. We will of course be using Android as both the target environment for your learning, as well as the inspiration for what's possible, and the basis for covering many topics that you'll encounter in other areas of software development as well.

I have laid out the book in four broad parts, each targeting your growing understanding of how to write apps for Android, and Android itself, as you progress through the content and examples. The chapters in each part cover the following topics.

## Detailing More of the Remaining Chapters

**Chapter 1, Introducing Android:** You're reading it! You have already covered much of the introduction to Android and in just a few pages will embark on your first Android application.

**Chapter 2, Introducing Android Studio:** We will look at the most popular toolset with which to build Android applications – Android Studio – and how to go about getting this free software. We will also briefly examine some alternatives to Android Studio and how to stay informed about new releases, new features, and other changes in how Android applications can be built. Lastly, we'll introduce the idea of an "emulated" Android phone – an Android Virtual Device (AVD) – which is a way of having your computer simulate an Android handset and provides a playground on which to test your applications without needing to tinker with your actual phone.

**Chapter 3, Your First Android Application, Already!:** Yes, that's right. You'll get straight into creating your very first Android application. No need to wait until you've finished the book! Our first example will be very simple, but it will lay the foundation from which we will expand in all of the remaining chapters.

**Chapter 4, Exploring Your First Project:** In this chapter, we will take a virtual magnifying glass to the example you created in Chapter 3 and walk through each of the pieces of your first application to begin understanding where they came from, what they do, and why they are where they are.

**Chapter 5, Android Studio In-Depth:** If you are going to use an integrated set of tools for your future development, then diving deep into their capabilities will be a must. In this chapter, we will explore all of the key aspects of Android Studio, including code editing features, debuggers, profiling tools, and more.

**Chapter 6, Mastering Your Entire Developer Ecosystem:** This rounds out your understanding of all of the tools you can, and will, use in your development journey. This chapter will explore the tools that sit outside Android Studio's integrated environment, but nevertheless are vital to it, including the Java Developer Kit (JDK), Gradle, source control systems for your code and applications, management of Android Virtual Devices, and other key parts of your environment. We'll also look at aspects of your developer hardware that can influence your path to Android development.

**Chapter 7, Introducing Java for Android Development:** Ready to "level up" with Java? Regardless of your current knowledge level, this chapter will highlight the key areas of Java coding that you need for Android development, along with further resources to expand your Java expertise.

**Chapter 8, Introducing XML for Android Development:** Many aspects of Android application behavior are controlled by XML (Extensible Markup Language) data. This chapter walks you through the fundamentals of XML and how to apply XML for various aspects of your Android applications, including an application's manifest, its user interface, and more.

**Chapter 9, Exploring Android Concepts: Core UI Widgets:** Building on Chapter 8, we explore how to lay out Android user interfaces with stock components such as menus, on-screen widgets like fields, lists, images, and other visual items. This chapter will also introduce the key concept of an activity – the fundamental building block of Android user interfaces. Lastly, we will give an overview of Android Jetpack, which is a modern library offering contemporary layout approaches while offering backward compatibility.

**Chapter 10, Exploring Android Concepts: Layouts and More:** This chapter expands your repertoire and breadth of understanding for all of Android's user interface components and builds further on the work from Chapters 8 and 9.

**Chapter 11, Understanding Activities:** Armed with the UI concepts from earlier chapters, you will explore the full power of activities as the basic building block of all Android applications.

**Chapter 12, Introducing Fragments:** You will learn the wider concept of fragments that power development for many different screen sizes and layout options.

**Chapter 13, Working with Sound, Audio, and Music for Android:** In this chapter, we will explore all aspects of audio for your applications, including playback of audio and use of sounds in applications, recording audio, and even creating audio on Android devices.

**Chapter 14, Working with Video and Movies for Android:** If you are a budding Steven Spielberg, Sophia Coppola, or even just YouTube star, this chapter is for you. We will cover Android's video capabilities for both capture and playback and how to work these capabilities into your applications.

**Chapter 15, Introducing Notifications:** Branch out beyond the boundaries of your application by using the event frameworks and notification systems that Android provides.

**Chapter 16, Exploring Device Capabilities with Calls:** There is more to your Android world of possibilities than just what can be shown on-screen. We will look at call capabilities, accessing sensors, and other information.

**Chapter 17, Understanding Intents, Events, and Receivers:** Behind the scenes of every Android application, a wealth of background features keep things moving. This chapter covers the core concepts central to the Android platform and shows how they shape and influence your applications.

**Chapter 18, Introducing Android Services:** In this chapter, we'll explore working with other code and other applications to enrich your own application and what it can do for your users.

**Chapter 19, Working with Files in Android:** Android enables you to work with many kinds of data, configuration, and other files. This chapter will start your journey understanding where and how Android applications can leverage traditional files for powering your user experiences.

**Chapter 20, Working with Databases in Android:** Data drives every application, and knowing how to store, manage, and use data for applications is key to making them great. This chapter will cover the many ways Android provides for working with data.

With that foretaste of what's to come complete, there's no time like the present to get started. Chapter 2 awaits on the next page!

---

# Chapter 2

## Introducing Android Studio

In this chapter, we will introduce Android Studio, the primary tool you will use to write the software for your Android applications. While there are a range of other pieces of software you will rely on, the Android Studio Integrated Development Environment (IDE) will be central to bringing your ideas to life. You can think of it as the software you use to write further software.

If you have done any kind of software development in the past, you might find some of the concepts familiar. If that is the case, feel free to skip ahead to the sections later in this chapter that dive straight into the mechanics of installing Android Studio for your chosen platform. You can of course keep reading, to see if there's anything new or interesting on the topic.

Let's now delve into the most important toolset you will use as an Android apps developer – Android Studio!

### Understanding What Is Meant by Integrated Development Environment, or IDE

Before we dive in, let's define what is meant by IDE for those who haven't encountered the term or are completely new to programming. The term Integrated Development Environment is almost self-explanatory, but not quite. If we pull it apart, the "Development Environment" bit is simply referring to the fact that it is the environment (in the software sense) where developers develop. That's akin to saying Microsoft Word, or Google Docs, is an author's "Writing Environment" – the place where they do their writing.

The term "Integrated" is also straightforward, but knowing what is being integrated is key to an overall understanding of an IDE. To extend our writing analogy a little further, as a software developer you will also be writing, but in your case, it will be human-readable programming code in Java and potentially several other languages. To that end, every IDE includes (integrates) a code editor, where you write the actual raw code. So far, so good.

Just as Microsoft Word provides some extra bundled tools for an author, like spelling check and online dictionary definition lookup, so too an IDE pulls in other tools and makes them available in an integrated way to a software developer. Alongside the code editor, there are usually language reference tools so you can look up how a software library works, build tools that take your written code and compile it into a working piece of software, debugging tools to help you identify and understand problems and errors, and a host of other tools like performance profilers, code formatters, syntax highlighters, live inspection tools, network monitors, and more.

The crux of the IDE is that all of these tools are integrated, and more or less work well together, without needing the developer to manually pass things around between the tools. Prior to the emergence of IDEs, this was the less glamorous life of a developer. With an IDE like Android Studio in place to handle mundane, convoluted, and repetitive actions, you are left free to focus on the creative problem-solving aspects of writing applications, worrying less about the low-level plumbing.

## **The History and Provenance of Android Studio**

When Google first released developer tools for Android, it targeted one of the most popular open source development environments of the day, Eclipse. It released a set of tools that plugged into Eclipse, known as the Android Developer Tools, or ADT. This combination – Eclipse and ADT – satisfied many Android developers for many years. Fast-forward over a decade, and while Eclipse is still a hugely popular IDE, a range of other IDEs have risen and fallen in prominence, and Google is nothing if not keen to be at the leading edge of these sorts of changes.

In 2013, Google announced that it was cooperating with JetBrains, a company that had developed a modern IDE known as IntelliJ IDEA that focused primarily on first-class developer experiences for those building Java-based applications, and the new collaborative product would be based on IntelliJ IDEA and known as Android Studio and released as a completely free IDE for Android development. Android Studio version 1.0 was released later in 2014. In subsequent years, Google announced that Android Studio would be the premier IDE for developing Android applications and it would cease focusing on and investing in other tools. This hasn't stopped Eclipse fans from continuing to use Eclipse, the ADT, and a range of other tools – we'll touch on such alternatives at the end of this chapter.

## **Downloading the Android Studio Installer for Your Platform**

You now have enough context to understand why Android Studio is important and the role it will play for you in your future Android application development. It's time to get Android Studio and get coding!

The number one location for downloading the installation packages for Android Studio is the official Android web page itself. You can start at the home page, at [www.android.com/](http://www.android.com/), or dive straight into the developer download page at <https://developer.android.com/studio>.