DEITEL® DEVELOPER SERIES

SECOND EDITION

Android™ for Programmers

An App-Driven Approach

7 Fully Coded Android™ Apps

Volume 1

PAUL DEITEL • HARVEY DEITEL
ABBEY DEITEL

ANDROID™ FOR PROGRAMMERS
AN APP-DRIVEN APPROACH
SECOND EDITION, VOLUME 1
DEITEL® DEVELOPER SERIES

Library of Congress Cataloging-in-Publication Data

On file

ISBN-13: 978-0-13357092-2
ISBN-10: 0-13-357092-4

Text printed in the United States on recycled paper at RR Donnelley in Crawfordsville, Indiana.
First printing, December 2013

Android™ for Programmers
An App-Driven Approach
Second Edition, Volume 1
Deitel® Developer Series

Paul Deitel • Harvey Deitel • Abbey Deitel
Deitel & Associates, Inc.

PRENTICE HALL

Upper Saddle River, NJ • Boston • Indianapolis • San Francisco
New York • Toronto • Montreal • London • Munich • Paris • Madrid
Capetown • Sydney • Tokyo • Singapore • Mexico City

Trademarks

DEITEL, the double-thumbs-up bug and DIVE-INTO are registered trademarks of Deitel & Associates, Inc.

Java is a registered trademark of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Google, Android, Google Play, Google Maps, Google Wallet, Nexus, YouTube, AdSense and AdMob are trademarks of Google, Inc.

Microsoft and/or its respective suppliers make no representations about the suitability of the information contained in the documents and related graphics published as part of the services for any purpose. All such documents and related graphics are provided "as is" without warranty of any kind. Microsoft and/or its respective suppliers hereby disclaim all warranties and conditions with regard to this information, including all warranties and conditions of merchantability, whether express, implied or statutory, fitness for a particular purpose, title and non-infringement. In no event shall Microsoft and/or its respective suppliers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of information available from the services.

The documents and related graphics contained herein could include technical inaccuracies or typographical errors. Changes are periodically added to the information herein. Microsoft and/or its respective suppliers may make improvements and/or changes in the product(s) and/or the program(s) described herein at any time. Partial screen shots may be viewed in full within the software version specified.

Microsoft® and Windows® are registered trademarks of the Microsoft Corporation in the U.S.A. and other countries. Screen shots and icons reprinted with permission from the Microsoft Corporation. This book is not sponsored or endorsed by or affiliated with the Microsoft Corporation.

Throughout this book, trademarks are used. Rather than put a trademark symbol in every occurrence of a trademarked name, we state that we are using the names in an editorial fashion only and to the benefit of the trademark owner, with no intention of infringement of the trademark.

In Memory of Amar G. Bose, MIT Professor and Founder and Chairman of the Bose Corporation:

It was a privilege being your student—and members of the next generation of Deitels, who heard our dad say how your classes inspired him to do his best work.

You taught us that if we go after the really hard problems, then great things can happen.

Harvey Deitel
Paul and Abbey Deitel

This page intentionally left blank

Contents

Preface

Welcome to the dynamic world of Android smartphone and tablet app development with the Android Software Development Kit (SDK), the Java™ programming language, the Eclipse-based Android Development Tools IDE, and the new and rapidly evolving Android Studio IDE.

Android for Programmers: An App-Driven Approach, 2/e, Volume 1 presents leading-edge mobile computing technologies for professional software developers. At the heart of the book is our *app-driven approach*—we present concepts in the context of *seven complete working Android apps* rather than using code snippets. Chapters 2–8 each present one app. We begin each of these chapters with an introduction to the app, an app test-drive showing one or more sample executions and a technologies overview. Then we proceed with a detailed code walkthrough of the app's source code. All of the source code is available at `www.deitel.com/books/AndroidFP2`. We recommend that you have the source code open in the IDE as you read the book.

Sales of Android devices and app downloads have been growing exponentially. The first-generation Android phones were released in October 2008. A study by Strategy Analytics showed that by October 2013, Android had 81.3% of the global smartphone market share, compared to 13.4% for Apple, 4.1% for Microsoft and 1% for Blackberry.[1] According to an IDC report, by the end of the first quarter of 2013 Android had 56.5% of the global tablet market share, compared to 39.6% for Apple's iPad and 3.7% for Microsoft Windows tablets.[2]

There are now over one billion Android smartphones and tablets in use,[3] and more than 1.5 million Android devices are being activated daily.[4] According to IDC, Samsung is the leading Android manufacturer, accounting for nearly 40% of Android device shipments in the third quarter of 2013.

Billions of apps have been downloaded from Google Play™—Google's marketplace for Android apps. The opportunities for Android app developers are enormous.

Fierce competition among popular mobile platforms and carriers is leading to rapid innovation and falling prices. Competition among the dozens of Android device manufacturers is driving hardware and software innovation within the Android community.

Copyright Notice and Code License

1. `http://blogs.strategyanalytics.com/WSS/post/2013/10/31/Android-Captures-Record-81-Percent-Share-of-Global-Smartphone-Shipments-in-Q3-2013.aspx`.
2. `http://www.idc.com/getdoc.jsp?containerId=prUS24093213`.
3. `http://www.android.com/kitkat`.
4. `http://www.technobuffalo.com/2013/04/16/google-daily-android-activations-1-5-million`.

3.0 Unported License (`http://creativecommons.org/licenses/by/3.0`), *with the exception that they may not be reused in any way in educational tutorials and textbooks, whether in print or digital format. Additionally, the authors and publisher make no warranty of any kind, expressed or implied, with regard to these programs or to the documentation contained in this book. The authors and publisher shall not be liable in any event for incidental or consequential damages in connection with, or arising out of, the furnishing, performance, or use of these programs. You're welcome to use the apps in the book as shells for your own apps, building on their existing functionality. If you have any questions, contact us at* `deitel@deitel.com`.

Intended Audience

We assume that you're a Java programmer with object-oriented programming experience. Because of the improved Android development tools, we were able to eliminate almost all XML markup in this edition. There are still two small, easy-to-understand XML files you'll need to manipulate. We use only complete, working apps, so if you don't know Java but have object-oriented programming experience in languages like C#/.NET, Objective-C/Cocoa or C++ (with class libraries), you should be able to master the material quickly, learning a good amount of Java and Java-style object-oriented programming along the way.

This book is *not* a Java tutorial, but it presents a significant amount of Java in the context of Android app development. If you're interested in learning Java, check out our publications:

- *Java for Programmers, 2/e* (`www.deitel.com/books/javafp2`)
- *Java Fundamentals: Parts I and II* LiveLessons videos (`www.deitel.com/books/LiveLessons`).
- *Java How to Program, 10/e* (`www.deitel.com/books/jhtp10`)

If you're not familiar with XML, see these online tutorials:

- `http://www.ibm.com/developerworks/xml/newto`
- `http://www.w3schools.com/xml/xml_whatis.asp`
- `http://www.deitel.com/articles/xml_tutorials/20060401/XMLBasics`
- `http://www.deitel.com/articles/xml_tutorials/20060401/XMLStructuringData`

Key Features

Here are some of this book's key features:

App-Driven Approach. Chapters 2–8 each present one completely coded app—we discuss what the app does, show screen shots of the app in action, test-drive it and overview the technologies and architecture we'll use to build it. Then we build the app's GUI and resource files, present the complete code and do a detailed code walkthrough. We discuss the programming concepts and demonstrate the functionality of the Android APIs used in the app.

Android SDK 4.3 and 4.4. We cover various new Android Software Development Kit (SDK) 4.3 and 4.4 features.

Fragments. Starting with Chapter 5, we use `Fragments` to create and manage portions of each app's GUI. You can combine several fragments to create user interfaces that take ad-

vantage of tablet screen sizes. You also can easily interchange fragments to make your GUIs more dynamic, as you'll do in Chapter 8.

Support for multiple screen sizes and resolutions. Throughout the app chapters we demonstrate how to use Android's mechanisms for automatically choosing resources (layouts, images, etc.) based on a device's size and orientation.

Eclipse-Based Android Development Tools (ADT) IDE coverage in the print book. The free Android Development Tools (ADT) integrated development environment (IDE)—which includes Eclipse and the ADT plugin—combined with the free Java Development Kit (JDK) provide all the software you'll need to create, run and debug Android apps, export them for distribution (e.g., upload them to Google Play™) and more.

Android Studio IDE. This is the preferred IDE for the future of Android app development. Because it's new and evolving rapidly, we put our discussions of it online at:

```
http://www.deitel.com/books/AndroidFP2
```

We'll show how to import existing projects so you can test-drive our apps. We'll also demonstrate how to create new apps, build GUIs, modify resource files and test your apps. If you have any questions, contact us at `deitel@deitel.com`.

Immersive Mode. The status bar at the top of the screen and the menu buttons at the bottom can be hidden, allowing your apps to fill more of the screen. Users can access the status bar by swiping down from the top of the screen, and the system bar (with the back button, home button and recent apps button) by swiping up from the bottom.

Printing Framework. Android 4.4 KitKat allows you to add printing functionality to your apps, such as locating available printers over Wi-Fi or the cloud, selecting the paper size and specifying which pages to print.

Testing on Android Smartphones, Tablets and the Android Emulator. For the best app-development experience, you should test your apps on actual Android smartphones and tablets. You can still have a meaningful experience using just the Android emulator (see the Before You Begin section), however it's processor-intensive and can be slow, particularly with games that have a lot of moving parts. In Chapter 1, we mention some Android features that are not supported on the emulator.

Multimedia. The apps use a broad range of Android multimedia capabilities, including graphics, images, frame-by-frame animation and audio.

Uploading Apps to Google Play. Chapter 9, Google Play and App Business Issues, walks you through the registration process for Google Play and setting up a merchant account so you can sell your apps. You'll learn how to prepare apps for submission to Google Play, find tips for pricing your apps, and resources for monetizing them with in-app advertising and in-app sales of virtual goods. You'll also find resources for marketing your apps. Chapter 9 can be read after Chapter 1.

Features

Syntax Coloring. For readability, we syntax color the code, similar to Eclipse's and Android Studio's use of syntax coloring. Our syntax-coloring conventions are as follows:

```
comments appear in green
keywords appear in dark blue
constants and literal values appear in light blue
all other code appears in non-bold black
```

Code Highlighting. We emphasize the key code segments in each program by enclosing them in yellow rectangles.

Using Fonts for Emphasis. We use various font conventions:

- The defining occurrences of key terms appear in **bold maroon** for easy reference.
- On-screen IDE components appear in **bold Helvetica** (e.g., the **File** menu).
- Program source code appears in `Lucida` (e.g., `int x = 5;`).

In this book you'll create GUIs using a combination of visual programming (point and click, drag and drop) and writing code.

We use different fonts when we refer to GUI elements in program code versus GUI elements displayed in the IDE:

- When we refer to a GUI component that we create in a program, we place its class name and object name in a `Lucida` font—e.g., "`Button saveContactButton`."
- When we refer to a GUI component that's part of the IDE, we place the component's text in a **bold Helvetica** font and use a plain text font for the component's type—e.g., "the **File** menu" or "the **Run** button."

Using the > Character. We use the **>** character to indicate selecting a menu item from a menu. For example, we use the notation **File > New** to indicate that you should select the **New** menu item from the **File** menu.

Source Code. All of the book's source code is available for download from:

```
www.deitel.com/books/AndroidFP2
www.informit.com/title/0133570924
```

Documentation. All the Android and Java documentation you'll need to develop Android apps is available free at `http://developer.android.com` and `http://www.oracle.com/technetwork/java/javase/downloads/index.html`. The documentation for Eclipse is available at `www.eclipse.org/documentation`. The documentation for Android Studio is available at `http://developer.android.com/sdk/installing/studio.html`.

Chapter Objectives. Each chapter begins with a list of learning objectives.

Figures. Hundreds of tables, source code listings and Android screen shots are included.

Software Engineering. We stress program clarity and performance, and concentrate on building well-engineered, object-oriented software.

Index. We include an extensive index for reference. The page number of the defining occurrence of each key term in the book is highlighted in the index in **bold maroon**.

Working with Open-Source Apps

There are numerous free, open-source Android apps available online which are excellent resources for learning Android app development. We encourage you to download open-

source apps and read their source code to understand how they work. **Caution: The terms of open-source licenses vary considerably.** Some allow you to use the app's source code freely for any purpose, while others stipulate that the code is available for personal use only—not for creating for-sale or publicly available apps. **Be sure to read the licensing agreements carefully. If you wish to create a commercial app based on an open-source app, you should consider having an intellectual property attorney read the license; be aware that these attorneys charge significant fees.**

Android for Programmers: An App-Driven Approach, Second Edition, Volume 2

Volume 2, which will be published in 2014, contains additional app-development chapters that introduce property animation, Google Play game services, video, speech synthesis and recognition, GPS, the Maps API, the compass, object serialization, web services, audio recording and playback, Bluetooth®, HTML5 mobile apps and more. **For the status of Volume 2 and for continuing book updates, visit**

```
http://www.deitel.com/books/AndroidFP2
```

Android Fundamentals, Second Edition LiveLessons Video Training Products

Our *Android Fundamentals, Second Edition* LiveLessons videos show you what you need to know to start building robust, powerful Android apps with the Android Software Development Kit (SDK) 4.3 and 4.4, the Java™ programming language and the Eclipse™ and Android Studio integrated development environments (IDEs). It will include approximately 20 hours of expert training synchronized with *Android for Programmers, Second Edition* (Volumes 1 and 2). The videos for Volume 1 will be available spring 2014. For additional information about Deitel LiveLessons video products, visit

```
www.deitel.com/livelessons
```

or contact us at `deitel@deitel.com`. You can also access our LiveLessons videos if you have a subscription to Safari Books Online (`www.safaribooksonline.com`).

Join the Deitel & Associates, Inc. Social Networking Communities

To receive updates on this and our other publications, new and updated apps, online Resource Centers, instructor-led onsite training courses, partner offers and more, join the Deitel social networking communities on Facebook® (`http://www.deitel.com/deitelfan`), Twitter® (`@deitel`), LinkedIn® (`http://bit.ly/DeitelLinkedIn`) Google+™ (`http://google.com/+DeitelFan`), and YouTube® (`http://youtube.com/user/DeitelTV`) and subscribe to the *Deitel® Buzz Online* newsletter (`http://www.deitel.com/newsletter/subscribe.html`).

Contacting the Authors

We'd sincerely appreciate your comments, criticisms, corrections and suggestions for improvement. Please address all questions and other correspondence to:

```
deitel@deitel.com
```

We'll respond promptly, and post corrections and clarifications on:

```
www.deitel.com/books/AndroidFP2
```

and on Facebook, Twitter, Google+, LinkedIn and the *Deitel® Buzz Online*.

Visit www.deitel.com to:

- Download code examples
- Check out the growing list of programming Resource Centers
- Receive updates for this e-book, subscribe to the free *Deitel® Buzz Online* e-mail newsletter at `www.deitel.com/newsletter/subscribe.html`
- Receive information on our *Dive Into® Series* instructor-led programming language training courses offered at customer sites worldwide

Acknowledgments

Thanks to Barbara Deitel for long hours devoted to this project—she created all of our Android Resource Centers, and patiently researched hundreds of technical details.

This book was a cooperative effort between professional and academic divisions of Pearson. We appreciate the efforts and 18-year mentorship of our friend and professional colleague Mark L. Taub, Editor-in-Chief of the Pearson Technology Group. Mark and his team handle all of our professional books and LiveLessons video products. Kim Boedigheimer recruited distinguished members of the Android community and managed the review team for the Android content. We selected the cover art and Chuti Prasertsith and Sandra Schroeder designed the cover. John Fuller manages the production of all of our Deitel Developer Series books.

We also appreciate the guidance, wisdom and energy of Tracy Johnson, Executive Editor, Computer Science. Tracy and her team handle all of our academic textbooks. Carole Snyder recruited the book's academic reviewers and managed the review process. Bob Engelhardt manages the production of our academic publications.

We'd like to thank Michael Morgano, a former colleague of ours at Deitel & Associates, Inc., now an Android developer at Imerj™, who co-authored the first editions of this book and our book, *iPhone for Programmers: An App-Driven Approach*. Michael is an extraordinarily talented software developer.

Reviewers of the Content from Android for Programmers: An App-Driven Approach *and* Android How to Program *Recent Editions*

We wish to acknowledge the efforts of our first and second edition reviewers. They scrutinized the text and the code and provided countless suggestions for improving the presentation: Paul Beusterien (Principal, Mobile Developer Solutions), Eric J. Bowden, COO (Safe Driving Systems, LLC), Tony Cantrell (Georgia Northwestern Technical College), Ian G. Clifton (Independent Contractor and Android App Developer, Daniel Galpin (Android Advocate and author of *Intro to Android Application Development*), Jim Hathaway (Application Developer, Kellogg Company), Douglas Jones (Senior Software Engineer, Fullpower Technologies), Charles Lasky (Nagautuck Community College), Enrique Lopez-Manas (Lead Android Architect, Sixt, and Computer Science Teacher at the Univer-

sity of Alcalá in Madrid), Sebastian Nykopp (Chief Architect, Reaktor), Michael Pardo (Android Developer, Mobiata), Ronan "Zero" Schwarz (CIO, OpenIntents), Arijit Sengupta (Wright State University), Donald Smith (Columbia College), Jesus Ubaldo Quevedo-Torrero (University of Wisconsin, Parkside), Dawn Wick (Southwestern Community College) and Frank Xu (Gannon University).

Well, there you have it! *Android for Programmers: An App-Driven Approach, Second Edition, Volume 1* will quickly get you developing Android apps. We hope you enjoy reading the book as much as we enjoyed writing it!

Paul Deitel
Harvey Deitel
Abbey Deitel

About the Authors

Paul Deitel, CEO and Chief Technical Officer of Deitel & Associates, Inc., is a graduate of MIT, where he studied Information Technology. He holds the Java Certified Programmer and Java Certified Developer certifications, and is an Oracle Java Champion. Through Deitel & Associates, Inc., he has delivered hundreds of programming courses worldwide to clients, including Cisco, IBM, Siemens, Sun Microsystems, Dell, Fidelity, NASA at the Kennedy Space Center, the National Severe Storm Laboratory, White Sands Missile Range, Rogue Wave Software, Boeing, SunGard Higher Education, Nortel Networks, Puma, iRobot, Invensys and many more. He and his co-author, Dr. Harvey M. Deitel, are the world's best-selling programming-language textbook/professional book/video authors.

Dr. Harvey Deitel, Chairman and Chief Strategy Officer of Deitel & Associates, Inc., has more than 50 years of experience in computing. Dr. Deitel earned B.S. and M.S. degrees in Electrical Engineering from MIT and a Ph.D. in Mathematics from Boston University. In the 1960s, through Advanced Computer Techniques and Computer Usage Corporation, he worked on the teams building various IBM operating systems. In the 1970s, he built commercial software systems. He has extensive college teaching experience, including earning tenure and serving as the Chairman of the Computer Science Department at Boston College before founding Deitel & Associates, Inc., in 1991 with his son, Paul Deitel. The Deitels' publications have earned international recognition, with translations published in Simplified Chinese, Traditional Chinese, Korean, Japanese, German, Russian, Spanish, French, Polish, Italian, Portuguese, Greek, Urdu and Turkish. Dr. Deitel has delivered hundreds of programming courses to corporate, academic, government and military clients.

Abbey Deitel, President of Deitel & Associates, Inc., is a graduate of Carnegie Mellon University's Tepper School of Management where she received a B.S. in Industrial Management. Abbey has been managing the business operations of Deitel & Associates, Inc. for 16 years. She has contributed to numerous Deitel & Associates publications and, together with Paul and Harvey, is the co-author of *Android for Programmers: An App-Driven Approach, 2/e*, *iPhone for Programmers: An App-Driven Approach*, *Internet & World Wide Web How to Program, 5/e*, *Visual Basic 2012 How to Program, 6/e* and *Simply Visual Basic 2010, 5/e*.

Deitel® Dive-Into® Series Corporate Training

Deitel & Associates, Inc., founded by Paul Deitel and Harvey Deitel, is an internationally recognized authoring and corporate training organization, specializing in Android and iOS app development, computer programming languages, object technology and Internet and web software technology. The company's clients include many of the world's largest corporations, government agencies, branches of the military, and academic institutions. The company offers instructor-led training courses delivered at client sites worldwide on major programming languages and platforms, including Android app development, Objective-C and iOS app development, Java™, C++, Visual C++®, C, Visual C#®, Visual Basic®, XML®, Python®, object technology, Internet and web programming and a growing list of additional programming and software development courses.

Through its 37-year publishing partnership with Prentice Hall/Pearson, Deitel & Associates, Inc., publishes leading-edge programming professional books, college textbooks and *LiveLessons* video courses. Deitel & Associates, Inc. and the authors can be reached at:

```
deitel@deitel.com
```

To learn more about Deitel's *Dive-Into® Series* Corporate Training curriculum, visit:

```
www.deitel.com/training
```

To request a proposal for worldwide on-site, instructor-led training at your organization, e-mail `deitel@deitel.com`.

Individuals wishing to purchase Deitel books and *LiveLessons* video training can do so through `www.deitel.com`. Bulk orders by corporations, the government, the military and academic institutions should be placed directly with Pearson. For more information, visit

```
www.informit.com/store/sales.aspx
```

Before You Begin

In this section, you'll set up your computer for use with this book. The Android development tools are frequently updated. Before reading this section, check the book's website

```
http://www.deitel.com/books/AndroidFP2/
```

to see if we've posted an updated version.

Font and Naming Conventions

We use fonts to distinguish between on-screen components (such as menu names and menu items) and Java code or commands. Our convention is to show on-screen components in a sans-serif bold **Helvetica** font (for example, **Project** menu) and to show file names, Java code and commands in a sans-serif `Lucida` font (for example, the keyword `public` or class `Activity`). When specifying commands to select in menus, we use the **>** notation to indicate a menu item to select. For example, **Window > Preferences** indicates that you should select the **Preferences** menu item from the **Window** menu.

Software and Hardware System Requirements

To develop Android apps you need a Windows®, Linux or Mac OS X system. To view the latest operating-system requirements visit:

```
http://developer.android.com/sdk/index.html
```

and scroll down to the **SYSTEM REQUIREMENTS** heading. We developed the apps in this book using the following software:

- Java SE 7 Software Development Kit
- Android SDK/ADT Bundle based on the Eclipse IDE
- Android SDK versions 4.3 and 4.4

You'll see how to obtain each of these in the next sections.

Installing the Java Development Kit (JDK)

Android requires the *Java Development Kit* (*JDK*) version 7 (JDK 7) or 6 (JDK 6). *We used JDK 7*. To download the JDK for Windows, OS X or Linux, go to

```
http://www.oracle.com/technetwork/java/javase/downloads/index.html
```

You need only the JDK. Choose the 32-bit or 64-bit version based on your computer hardware and operating system. Most recent computers have 64-bit hardware—check your system's specifications. If you have a 32-bit operating system, you must use the 32-bit JDK. Be sure to follow the installation instructions at

```
http://docs.oracle.com/javase/7/docs/webnotes/install/index.html
```

Android Integrated Development Environment (IDE) Options

Google now provides two Android IDE options:

- Android SDK/ADT bundle—a version of the *Eclipse IDE* that comes preconfigured with the latest Android Software Development Kit (SDK) and the latest Android Development Tools (ADT) plugin. At the time of this writing, these were Android SDK version 4.4 and ADT version 22.3.
- Android Studio—Google's new Android IDE based on IntelliJ® IDEA and their preferred future IDE.

The Android SDK/ADT bundle has been widely used in Android app development for several years. Android Studio, introduced in May 2013, is an *early access version* and will be evolving rapidly. For this reason, we'll stay with the widely used Android SDK/ADT bundle in the book, and as online supplements at

```
http://www.deitel.com/books/AndroidFP2
```

we'll provide Android Studio versions of the Chapter 1 Test-Drive section and the Building the GUI section for each app, as appropriate.

Installing the Android SDK/ADT Bundle

To download the Android SDK/ADT bundle, go to

```
http://developer.android.com/sdk/index.html
```

and click the **Download the SDK ADT Bundle** button. When the download completes, extract the ZIP file's contents to your system. The resulting folder has an `eclipse` subfolder containing the Eclipse IDE and an `sdk` subfolder containing the Android SDK. As with the JDK, you can choose a 32-bit or 64-bit version. The Android SDK/ADT bundle 32-bit version should be used with the 32-bit JDK, and the 64-bit version with the 64-bit JDK.

Installing Android Studio

The IDE instructions in the printed book use the Android SDK/ADT bundle. You can also optionally install and use Android Studio. To download Android Studio, go to

```
http://developer.android.com/sdk/installing/studio.html
```

and click the **Download Android Studio** button. When the download completes, run the installer and follow the on-screen instructions to complete the installation. [*Note:* For Android 4.4 development in Android Studio, Android now supports Java SE 7 language features, including the diamond operator, multi-`catch`, `Strings` in `switch` and `try`-with-resources.]

Set the Java Compiler Compliance Level and Show Line Numbers

Android does not fully support Java SE 7. To ensure that the book's examples compile correctly, configure Eclipse to produce files that are compatible with Java SE 6 by performing the following steps:

1. Open Eclipse (or), which is located in the `eclipse` subfolder of the Android SDK/ADT bundle's installation folder.
2. When the **Workspace Launcher** window appears, click **OK**.

3. Select **Window > Preferences** to display the **Preferences** window. On Mac OS X, select **ADT > Preferences...**.
4. Expand the **Java** node and select the **Compiler** node. Under **JDK Compliance**, set the **Compiler compliance level** to 1.6 (to indicate that Eclipse should produce compiled code that's compatible with Java SE 6).
5. Expand the **General > Editors** node and select **TextEditors**, then ensure that **Show line numbers** is selected and click **OK**.
6. Close Eclipse.

Android 4.3 SDK

This book's examples were written using the Android 4.3 and 4.4 SDKs. At the time of this writing, 4.4 was the version included with the Android SDK/ADT bundle and Android Studio. You should also install Android 4.3 (and any other versions you might want to support in your apps). To install other Android platform versions, perform the following steps (skipping Steps 1 and 2 if Eclipse is already open):

1. Open Eclipse. Depending on your platform, the icon will appear as  or  .
2. When the **Workspace Launcher** window appears, click **OK**.
3. On Mac OS X, if you see a window indicating "**Could not find SDK folder '/Users/***YourAccount*/**android-sdk-macosx/'**," click **Open Preferences** then **Browse...** and select the `sdk` folder located where you extracted the Android SDK/ADT bundle.
4. Select **Window > Android SDK Manager** to display the **Android SDK Manager** (Fig. 1).

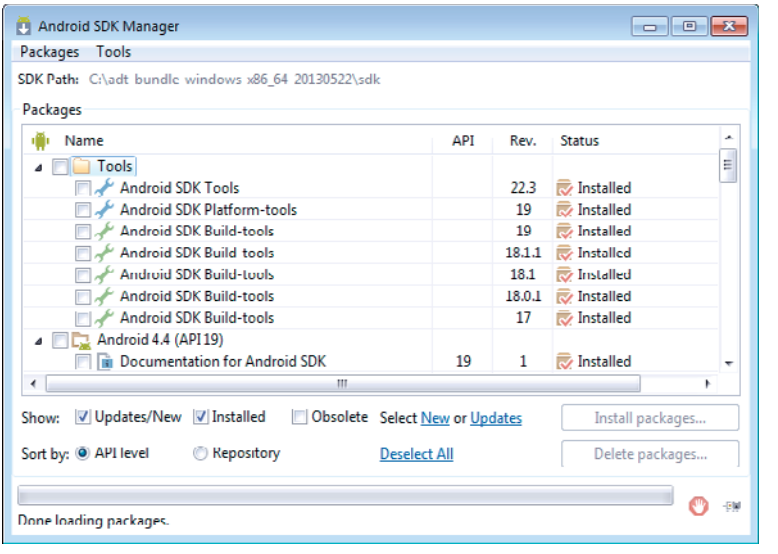

Fig. 1 | **Android SDK Manager** window.

5. The **Android SDK Manager**'s **Name** column shows all of the tools, platform versions and extras (such as APIs for interacting with Google services, like Maps) that you

can install. Uncheck the **Installed** checkbox. Then, if any of **Tools**, **Android 4.4 (API19)**, **Android 4.3 (API18)** and **Extras** appear in the **Packages** list, ensure that they're checked and click **Install # packages...** (# is the number of items to be installed) to display the **Choose Packages to Install** window. Most items in the **Extras** node are optional. For this book, you'll need the **Android Support Library** and **Google Play services**. The **Google USB Driver** is necessary for Windows users who wish to test apps on Android devices.]

6. In the **Choose Packages to Install** window, read the license agreements for each item. When you're done, click the **Accept License** radio button, then click the **Install** button. The status of the installation process will be displayed in the **Android SDK Manager** window.

Creating Android Virtual Devices (AVDs)

The **Android emulator**, included in the Android SDK, allows you to test apps on your computer rather than on an actual Android device. This is useful if you're learning Android and don't have access to Android devices, but can be *very* slow, so a real device is preferred if you have one. There are some hardware acceleration features that can improve emulator performance (`developer.android.com/tools/devices/emulator.html#acceleration`). Before running an app in the emulator, you must create an **Android Virtual Device (AVD)** which defines the characteristics of the device you want to test on, including the screen size in pixels, the pixel density, the physical size of the screen, size of the SD card for data storage and more. To test your apps for multiple Android devices, you can create AVDs that emulate each unique device. For this book, we use AVDs for Google's Android reference devices—the Nexus 4 phone, the Nexus 7 small tablet and Nexus 10 large tablet—which run unmodified versions of Android. To do so, perform the following steps:

1. Open Eclipse.
2. Select **Window > Android Virtual Device Manager** to display the **Android Virtual Device Manager** window, then select the **Device Definitions** tab (Fig. 2).

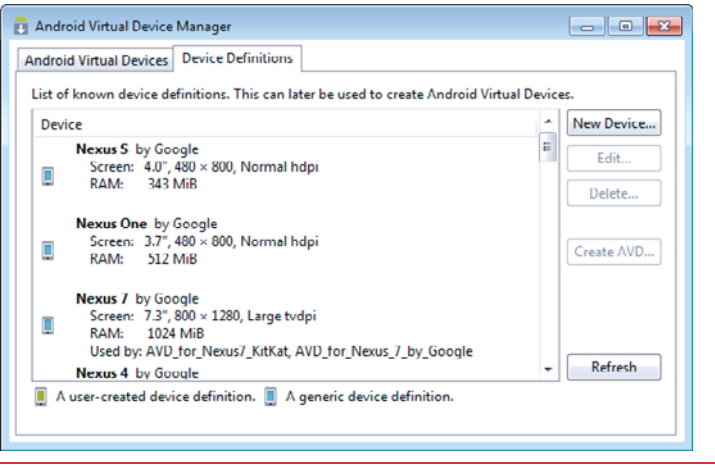

Fig. 2 | **Android Virtual Device Manager** window.

3. Google provides preconfigured devices that you can use to create AVDs. Select **Nexus 4 by Google**, then click **Create AVD...** to display the **Create new Android Virtual Device (AVD)** window (Fig. 3), then configure the options as shown and click **OK** to create the AVD. If you check **Hardware keyboard present**, you'll be able to use your computer's keyboard to type data into apps that are running in the AVD, but this may prevent the soft keyboard from displaying on the screen. If your computer does not have a camera, you can select **Emulated** for the **Front Camera** and **Back Camera** options. Each AVD you create has many other options specified in its `config.ini`. You can modify this file as described at

```
http://developer.android.com/tools/devices/managing-avds.html
```

to more precisely match the hardware configuration of your device.

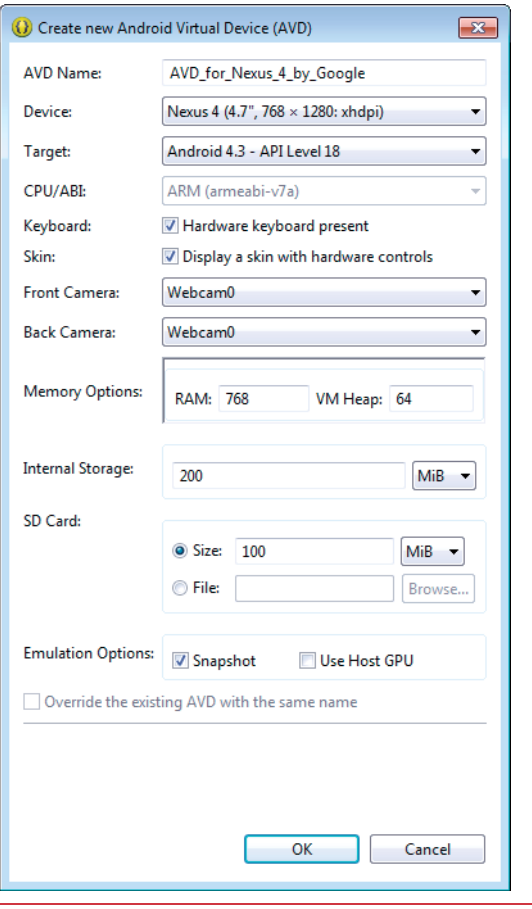

Fig. 3 | Configuring a Nexus 4 smartphone AVD for Android 4.3.

4. We also configured Android 4.3 AVDs that represent Nexus 7 by Google and Nexus 10 by Google for testing our tablet apps. Their settings are shown in Fig. 4. In

addition, we configured Android 4.4 AVDs for the Nexus 4, Nexus 7 and Nexus 10 with the names: `AVD_for_Nexus_4_KitKat`, `AVD_for_Nexus_7_KitKat`, and `AVD_for_Nexus_10_KitKat`,

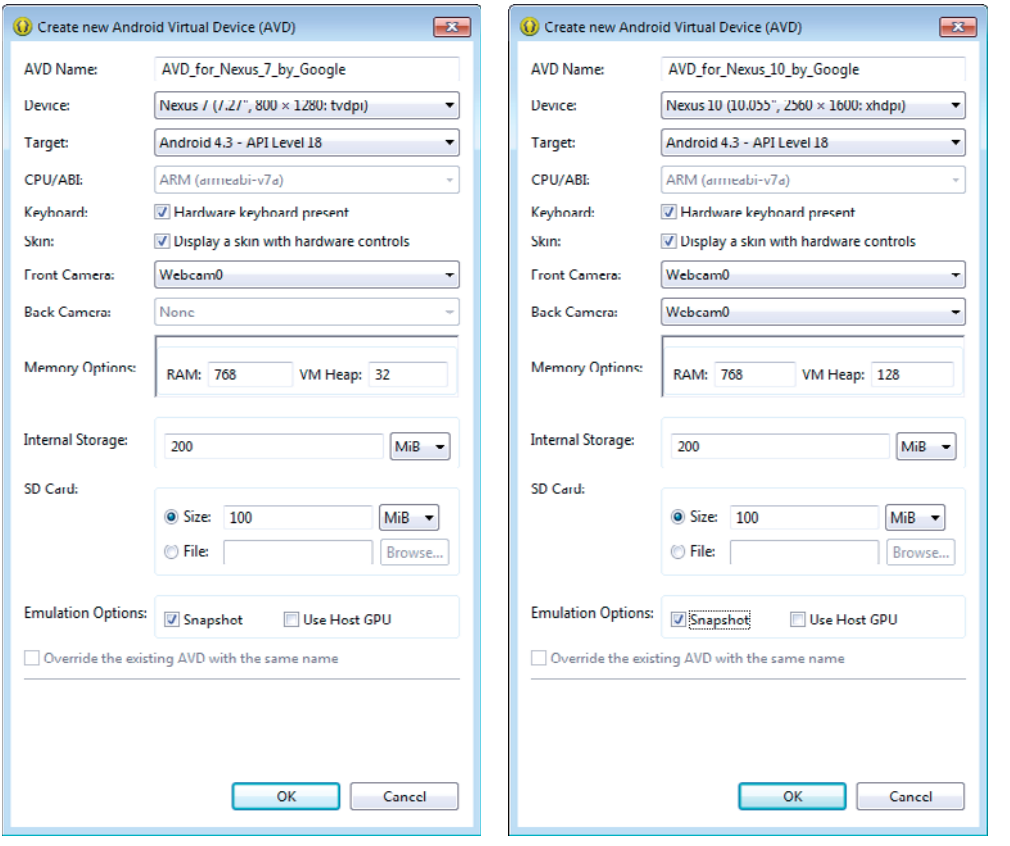

Fig. 4 | Configuring Nexus 7 and Nexus 10 tablet AVDs.

(Optional) Setting Up an Android Device for Development

As we mentioned, testing apps on AVDs can be slow due to AVD performance. If you have an Android device available to you, you should test the apps on that device. In addition, there are some features that you can test only on actual devices. To execute your apps on Android devices, follow the instructions at

```
http://developer.android.com/tools/device.html
```

If you're developing on Microsoft Windows, you'll also need the Windows USB driver for Android devices. In some cases on Windows, you may also need device-specific USB drivers. For a list of USB driver sites for various device brands, visit:

```
http://developer.android.com/tools/extras/oem-usb.html
```

Obtaining the Book's Code Examples

The examples for *Android for Programmers, 2/e, Volume 1* are available for download at

```
http://www.deitel.com/books/AndroidFP2/
```

If you're not already registered at our website, go to `www.deitel.com` and click the **Register** link. Fill in your information. Registration is free, and we do not share your information with anyone. Please verify that you entered your registration e-mail address correctly—you'll receive a confirmation e-mail with your verification code. *You must click the verification link in the e-mail before you can sign in at `www.deitel.com` for the first time*. Configure your e-mail client to allow e-mails from `deitel.com` to ensure that the verification e-mail is not filtered as junk mail. We send only occasional account-management e-mails unless you register separately for our free *Deitel® Buzz Online* e-mail newsletter at

```
http://www.deitel.com/newsletter/subscribe.html
```

Next, visit `www.deitel.com` and sign in using the **Login** link below our logo in the upper-left corner of the page. Go to `http://www.deitel.com/books/AndroidFP2/`. Click the **Examples** link to download a ZIP archive file containing the examples to your computer. Double click the ZIP file to unzip the archive, and make note of where you extract the file's contents on your system.

A Note Regarding the Android Development Tools

Google *frequently* updates the Android development tools. This often leads to problems compiling our apps when, in fact, the apps do not contain any errors. If you import one of our apps into Eclipse or Android Studio and it does not compile, there is probably a minor configuration issue. Please contact us by e-mail at `deitel@deitel.com` or by posting a question to:

- Facebook®—`facebook.com/DeitelFan`
- Google+™—`google.com/+DeitelFan`

and we'll help you resolve the issue.

You've now installed all the software and downloaded the code examples you'll need to study Android app development with *Android for Programmers, 2/e, Volume 1* and to begin developing your own apps. Enjoy!

This page intentionally left blank

1

Introduction to Android

Objectives

In this chapter you'll be introduced to:

- The history of Android and the Android SDK.
- Google Play Store for downloading apps.
- The Android packages used in this book to help you create Android apps.
- Basic object-technology concepts.
- Key software for Android app development, including the Android SDK, the Java SDK, the Eclipse integrated development environment (IDE) and Android Studio.
- Important Android documentation.
- Test-driving an Android drawing app in Eclipse (in the print book) and in Android Studio (online).
- Characteristics of great Android apps.

Outline

1.1 Introduction

Welcome to Android app development! We hope that working with *Android for Programmers: An App-Driven Approach, 2/e* will be an informative, challenging, entertaining and rewarding experience for you.

This book is geared toward *Java programmers*. We use only complete working apps, so if you don't know Java but have object-oriented programming experience in another language, such as C#, Objective-C/Cocoa or C++ (with class libraries), you should be able to master the material quickly, learning Java and Java-style object-oriented programming as you learn Android app development.

App-Driven Approach

We use an **app-driven approach**—new features are discussed in the context of complete working Android apps, with one app per chapter. For each app, we first describe it, then have you *test-drive* it. Next, we briefly overview the key **Eclipse IDE** (integrated development environment), Java and **Android SDK** (Software Development Kit) technologies we use to implement the app. For apps that require it, we walk through designing the GUI *visually* using Eclipse. Then we provide the complete source-code listing, using line numbers, *syntax coloring* and *code highlighting* to emphasize the key portions of the code. We also show one or more screen shots of the running app. Then we do a detailed code walkthrough, emphasizing the new programming concepts introduced in the app. You can download the source code for all of the book's apps from `http://www.deitel.com/books/androidFP2/`.

For each chapter, we also provide **Android Studio** IDE versions of any Eclipse-specific instructions. Because Android Studio is an early access version and will be evolving rapidly, we provide the Android Studio instructions on the book's website

```
http://www.deitel.com/books/AndroidFP2
```

This will enable us to keep the instructions up to date.

1.2 Android—The World's Leading Mobile Operating System

Android device sales are growing quickly, creating enormous opportunities for Android app developers.

- The first-generation Android phones were released in October 2008. By October 2013, a Strategy Analytics report showed that Android had 81.3% of the global *smartphone* market share, compared to 13.4% for Apple, 4.1% for Microsoft and 1% for Blackberry.[1]
- According to an IDC report, by the end of the first quarter of 2013 Android had 56.5% of the global *tablet* market share, compared to 39.6% for Apple's iPad and 3.7% for Microsoft Windows tablets.[2]
- As of April 2013, more than 1.5 million Android devices (including smartphones, tablets, etc.) were being activated daily.[3]
- At the time of this writing, there were over *one billion* activated Android devices.[4]
- Android devices now include smartphones, tablets, e-readers, robots, jet engines, NASA satellites, game consoles, refrigerators, televisions, cameras, health-care devices, smartwatches, automobile in-vehicle "infotainment" systems (for controlling the radio, GPS, phone calls, thermostat, etc.) and more.[5]

1.3 Android Features

Openness and Open Source

One benefit of developing Android apps is the openness of the platform. The operating system is *open source* and free. This allows you to view Android's source code and see how its features are implemented. You can also contribute to Android by reporting bugs (see `http://source.android.com/source/report-bugs.html`) or by participating in the Open Source Project discussion groups (`http://source.android.com/community/index.html`). Numerous open-source Android apps from Google and others are available on the Internet

1. `http://blogs.strategyanalytics.com/WSS/post/2013/10/31/Android-Captures-Record-81-Percent-Share-of-Global-Smartphone-Shipments-in-Q3-2013.aspx`.
2. `http://www.idc.com/getdoc.jsp?containerId=prUS24093213`.
3. `http://www.technobuffalo.com/2013/04/16/google-daily-android-activations-1-5-million`.
4. `http://venturebeat.com/2013/09/03/android-hits-1b-activations-and-will-be-called-kitkat-in-next-version`.
5. `http://www.businessweek.com/articles/2013-05-29/behind-the-internet-of-things-is-android-and-its-everywhere`.

(Fig. 1.1). Figure 1.2 shows you where you can get the Android source code, learn about the philosophy behind the open-source operating system and get licensing information.

URL	Description
`http://en.wikipedia.org/wiki/List_of_open_source_Android_applications`	Extensive list of open-source apps, organized by category (e.g., games, communication, emulators, multimedia, security).
`http://developer.android.com/tools/samples/index.html`	Google's sample apps for the Android platform; includes over 60 apps and games such as Lunar Lander, Snake and Tic Tac Toe.
`http://github.com/`	GitHub allows you to share your apps and source code and contribute to others' open-source projects.
`http://sourceforge.net`	SourceForge also allows you to share apps and source code and contribute to others' open-source projects.
`http://f-droid.org/`	Hundreds of free and open-source Android apps including the Adblock Plus advertisement blocker, aMetro public transportation navigation, AnySoftKeyboard (available in several languages), Apollo music player, Chinese Checkers game, DroidWeight weight tracker, Earth Live Wallpaper and more.
`http://blog.interstellr.com/post/39321551640/14-great-android-apps-that-are-also-open-source`	Lists 14 open-source Android apps with links to the code.
`http://www.openintents.org/en/libraries`	Provides nearly 100 open-source libraries that can be used to enhance app capabilities.
`http://www.androidviews.net`	Customized GUI controls for enhancing your app's appearance.
`http://www.stackoverflow.com`	Stack Overflow is a question-and-answer website for programmers. Users can vote on each answer, and the best responses rise to the top.

Fig. 1.1 | Open-source Android app and library resource sites.

Title	URL
Get Android Source Code	`http://source.android.com/source/downloading.html`
Governance Philosophy	`http://source.android.com/about/philosophy.html`
Licenses	`http://source.android.com/source/licenses.html`
FAQs	`http://source.android.com/source/faqs.html`

Fig. 1.2 | Resources and source code for the open-source Android operating system.

The openness of the platform spurs rapid innovation. Unlike Apple's *proprietary* iOS, which is available only on Apple devices, Android is available on devices from dozens of orig-

inal equipment manufacturers (OEMs) and through numerous telecommunications carriers worldwide. The intense competition among OEMs and carriers benefits customers.

Java

Android apps are developed with Java—one of the world's most widely used programming languages. Java was a logical choice for the Android platform, because it's powerful, free, open source and millions of developers already know it. Experienced Java programmers can quickly dive into Android development, using Google's Android APIs (Application Programming Interfaces) and others available from third parties.

Java is object oriented and has access to extensive class libraries that help you develop powerful apps quickly. GUI programming in Java is *event driven*—in this book, you'll write apps that respond to various user-initiated *events* such as *screen touches*. In addition to directly programming portions of your apps, you'll also use the Eclipse and Android Studio IDEs to conveniently drag and drop predefined objects such as buttons and text-boxes into place on your screen, and label and resize them. Using these IDEs, you can create, run, test and debug Android apps quickly and conveniently.

Multitouch Screen

Android smartphones wrap the functionality of a mobile phone, Internet client, MP3 player, gaming console, digital camera and more into a handheld device with full-color *multitouch screens*. With the touch of your fingers, you can navigate easily between using your phone, running apps, playing music, web browsing and more. The screen can display a keyboard for typing e-mails and text messages and entering data in apps (some Android devices also have physical keyboards).

Gestures

The multi-touch screens allow you to control the device with *gestures* involving one touch or multiple simultaneous touches (Fig. 1.3).

Gesture name	Physical action	Used to
Touch	Tap the screen once.	Open an app, "press" a button or a menu item.
Double touch	Tap the screen twice.	Zoom in on pictures, Google Maps and web pages. Tap the screen twice again to zoom back out.
Long press	Touch the screen and hold your finger in position.	Select items in a view—for example, checking an item in a list.
Swipe	Touch the screen, then move your finger in the swipe direction and release.	Flip item-by-item through a series, such as photos. A swipe automatically stops at the next item.
Drag	Touch and drag your finger across the screen.	Move objects or icons, or scroll through a web page or list.
Pinch zoom	Pinch two fingers together, or spread them apart.	Zoom in and out on the screen (e.g., resizing text and pictures).

Fig. 1.3 | Some common android gestures.