



# Preface

*Build a better mousetrap, and the world will beat a path to your door.*

—Ralph Waldo Emerson

Science and technology and the various forms of art,  
all unite humanity in a single and interconnected system.

—Zhores Aleksandrovich Medvede

Welcome to the dynamic world of Android smartphone and tablet app development with the Android Software Development Kit (SDK), the Java™ programming language, the Android Development Tools IDE, and the new and rapidly evolving Android Studio. We present leading-edge mobile computing technologies for students, instructors and professional software developers.

## ***Android How to Program, 2/e***

With this unique book—the second edition of the world’s first Android computer science textbook—you can learn Android even if you don’t know Java and even if you’re a programming novice. This book includes a complete, 300-page introduction to the Java core programming concepts that you’ll need when developing Android apps. The Java content is appropriate for programming novices.

*Android How to Program, 2/e* was formed by merging

- our professional book *Android for Programmers: An App-Driven Approach, 2/e, Volume 1*
- additional online chapters selected from *Android for Programmers: An App-Driven Approach, 2/e, Volume 2*
- condensed, introductory core content on object-oriented Java programming from our college textbook *Java How to Program, 9/e*
- hundreds of Android short-answer questions and app-development exercises we created for this book—most are in the book and many of the short-answer questions are in the test-item file for instructors.

We scoured the Android material, especially the fully coded Android apps, and enumerated the Java features that you’ll need to build these and similar apps. Then we extracted the corresponding Java content from *Java How to Program, 9/e*. That’s a 1500-page book, so it was challenging to whittle down that much content and keep it friendly, even for programming novices.

When you study the Android content, you’ll be thinking like a developer from the start. You’re going to study and build lots of real stuff and you’ll face the kinds of challenges professional developers must deal with. We’ll point you to the online documenta-

tion and forums where you can find additional information and get answers to your questions. We'll also encourage you to read, modify and enhance open-source code as part of your learning process.

## **Intended Audiences**

There are several audiences for this book. Most commonly, it will be used in upper-level elective college courses and industry professional courses for people familiar with object-oriented programming but who may or may not know Java and want to learn Android app development.

Uniquely, the book can also be used in introductory courses like CS1, intended for programming novices. We recommend that schools typically offering many sections of CS1 in Java consider designating one or two sections for ambitious students who have at least some prior programming experience and who want to work hard to learn a good amount of Java and Android in an aggressively paced one-semester course. The schools may want to list the courses with “honors” or “accelerated” designations. The book works especially well in two-semester introductory programming sequences where the introduction to Java is covered first.

## **App-Development Courses**

In 2007, Stanford offered a new course called Creating Engaging Facebook Apps. Students worked in teams developing apps, some of which landed in Facebook's top 10, earning some of the student developers millions of dollars.<sup>1</sup> This course gained wide recognition for encouraging student creativity and teamwork. Scores of colleges now offer app-development courses across many social networking and mobile platforms such as Android and iOS. We encourage you to read the online mobile app development syllabi and check out the YouTube™ videos created by instructors and students for many of these courses.

## **Android Ecosystem: Competition, Innovation, Explosive Growth and Opportunities**

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.<sup>2</sup> 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.<sup>3</sup>

There are now over one billion Android smartphones and tablets in use,<sup>4</sup> and more than 1.5 million Android devices are being activated daily.<sup>5</sup> According to IDC, Samsung

---

1. <http://www.businessinsider.com/these-stanford-students-made-millions-taking-a-class-on-facebook-2011-5>.

2. <http://blogs.strategyanalytics.com/WSS/post/2013/10/31/Android-Captures-Record-81-Percent-Share-of-Global-Smartphone-Shipments-in-Q3-2013.aspx>.

3. <http://www.idc.com/getdoc.jsp?containerId=prUS24093213>.

4. <http://www.android.com/kitkat>.

5. <http://www.technobuffalo.com/2013/04/16/google-daily-android-activations-1-5-million>.

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.

## App-Driven Approach

At the heart of the book is our *app-driven approach*—we present concepts in the context of *seven complete working Android apps* in the print book and more online. We begin each of the app chapters with an *introduction* to the app, an app *test-drive* showing one or more *sample executions*, and a *technologies overview*. We build the app’s GUI and resource files. Then we proceed with a detailed *code walkthrough* of the app’s source code in which we discuss the programming concepts and demonstrate the functionality of the Android APIs used in the app. All the source code is available at <http://www.deitel.com/books/AndroidHTP2> and at the book’s Companion Website <http://www.pearsonhighered.com/deitel>. We recommend that you have the source code open in the IDE as you read the book. Figure 1 lists the book’s apps and the key technologies we used to build each.

App	Technologies
Chapter 2, <b>Welcome App</b>	The Android Developer Tools (the Eclipse IDE and the ADT Plugin), visual GUI design, layouts, TextViews, ImageViews, accessibility and internationalization.
Chapter 3, <b>Tip Calculator App</b>	GridLayout, LinearLayout, EditText, SeekBar, event handling, NumberFormat and defining app functionality with Java.
Chapter 4, <b>Twitter® Searches App</b>	SharedPreferences, collections, ImageButton, ListView, ListActivity, ArrayAdapter, implicit intents and AlertDialogs.
Chapter 5, <b>Flag Quiz App</b>	Fragments, menus, preferences, AssetManager, tweened animations, Handler, Toasts, Explicit Intents, layouts for multiple device orientations.
Chapter 6, <b>Cannon Game App</b>	Listening for touches, frame-by-frame animation, graphics, sound, threading, SurfaceView and SurfaceHolder.
Chapter 7, <b>Doodlz App</b>	Two-dimensional graphics, Canvas, Bitmap, accelerometer, SensorManager, multitouch events, MediaStore, printing and Immersive Mode.
Chapter 8, <b>Address Book App</b>	AdapterViews and Adapters

**Fig. 1** | *Android How to Program* apps in the print book.

## Online Chapters and Book Updates

The Companion Website 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, Internet-enabled apps, audio recording and playback, Bluetooth<sup>®</sup>, HTML5 mobile apps and more. **Most of these chapters will be available for fall 2014 courses. For the status of the online chapters and for continuing book updates, visit**

<http://www.deitel.com/books/AndroidHTP2>

Join the Deitel communities on Facebook<sup>®</sup> (<http://www.deitel.com/deitelfan>), Twitter<sup>®</sup> (@deitel), LinkedIn<sup>®</sup> (<http://bit.ly/DeitelLinkedIn>) Google+<sup>™</sup> (<http://google.com/+DeitelFan>), and YouTube<sup>™</sup> (<http://youtube.com/user/DeitelTV>) and subscribe to the *Deitel<sup>®</sup> Buzz Online* newsletter (<http://www.deitel.com/newsletter/subscribe.html>).

## Copyright Notice and Code License

*All of the Android code and Android apps in the book are copyrighted by Deitel & Associates, Inc. The sample Android apps in the book are licensed under a Creative Commons Attribution 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](mailto:deitel@deitel.com).*

## Getting up to Speed in Java and XML

The Android portion of this book assumes that you already know Java and object-oriented programming. If you're not familiar with these, the appendices provide a condensed, friendly introduction to Java and the object-oriented programming techniques you'll need to develop Android apps. If you're interested in learning Java in more depth, you may want to check out the comprehensive treatment in our textbook *Java How to Program, 10/e* (<http://www.deitel.com/books/jhttp10>).

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

- [http://www.deitel.com/articles/xml\\_tutorials/20060401/XMLBasics/](http://www.deitel.com/articles/xml_tutorials/20060401/XMLBasics/)
- [http://www.deitel.com/articles/xml\\_tutorials/20060401/XMLStructuringData/](http://www.deitel.com/articles/xml_tutorials/20060401/XMLStructuringData/)
- <http://www.ibm.com/developerworks/xml/newto/>
- [http://www.w3schools.com/xml/xml\\_what\\_is.asp](http://www.w3schools.com/xml/xml_what_is.asp)

## Key Features of *Android How to Program, 2/e*

- *Android SDK 4.3 and 4.4.* We cover various new Android Software Development Kit (SDK) 4.3 and 4.4 features. [*Note:* The apps in this book are configured to run on Android devices with Android 4.3 and higher; however, most apps will work in 4.0 and higher by changing their minimum required SDK.]
- *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 advantage 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) 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.* This is the preferred IDE for the future of Android app development. Because this IDE is evolving quickly, we put our discussions of it online at:

<http://www.deitel.com/books/AndroidHTP2>

- *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 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 in the print book use a broad range of Android multimedia capabilities, including graphics, images, frame-by-frame animation and audio. The apps in the online chapters use property animation, video, speech synthesis and speech recognition.
- *Android Best Practices.* We adhere to accepted Android best practices, pointing them out in the detailed code walkthroughs. For more information, visit <http://developer.android.com/guide/practices/index.html>.
- *Java Content in the Appendices Can Be Used With Java SE 6 or Higher.*

- *Java Exception Handling.* We integrate basic exception handling early in the Java content then present a richer treatment in Appendix H; we use exception handling throughout the Android chapters.
- *Classes Arrays and ArrayList; Collections.* Appendix E covers class Arrays—which contains methods for performing common array manipulations—and generic class ArrayList—which implements a dynamically resizable array-like data structure. Appendix J introduces Java’s generic collections that are used frequently in our Android treatment.
- *Java Multithreading.* Maintaining app responsiveness is a key to building robust Android apps and requires extensive use of Android multithreading. Appendix J introduces multithreading fundamentals so that you can understand our use of the Android AsyncTask class in Chapter 8.
- *GUI Presentation.* Appendix I introduces Java GUI development. Android provides its own GUI components, so this appendix presents a few Java GUI components and focuses on nested classes and anonymous inner classes, which are used extensively for event-handling in Android GUIs.

## 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. Throughout the book you’ll find programming exercises that ask you to modify or enhance existing open-source apps. Our goal is to give you handles on interesting problems that may also inspire you to create new apps using the same technologies. **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.**

## Pedagogic Features

*Syntax Shading.* For readability, we syntax shade the code, similar to Eclipse’s and Android Studio’s use of syntax coloring. Our syntax-shading conventions are as follows:

```

comments appear in gray
constants and literal values appear in bold darker gray
keywords appear in bold black
all other code appears in non-bold black

```

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

*Using Fonts for Emphasis.* We use various font conventions:

- The defining occurrences of key terms appear in **bold** 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:

```
http://www.deitel.com/books/AndroidHTP2
http://www.pearsonhighered.com/deitel
```

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

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

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

*Self-Review Exercises and Answers.* Extensive self-review exercises *and* answers are included for self study.

*Exercises with a Current Flair.* We've worked hard to create topical Android app-development exercises. You'll develop apps using a broad array of current technologies. All of the Android programming exercises require the implementation of complete apps. You'll be asked to enhance the existing chapter apps, develop similar apps, use your creativity to develop your own apps that use the chapter technologies and build new apps based on open-source apps available on the Internet (and again, **be sure to read and comply with the open-source code-license terms for each app**). The Android exercises also include short-answer fill-in and true/false questions.

In the Java exercises, you'll be asked to recall important terms and concepts; indicate what code segments do; indicate what's wrong with a portion of code; write Java statements, methods and classes; and write complete Java programs.

*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**.

## Software Used in *Android How to Program, 2/e*

All the software you'll need for this book is available free for download from the Internet. See the Before You Begin section for the download links.



*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](http://www.eclipse.org/documentation). The documentation for Android Studio is available at <http://developer.android.com/sdk/installing/studio.html>.

## **Instructor Resources**

The following supplements are available to **qualified college instructors only** through Pearson Education's Instructor Resource Center (<http://www.pearsonhighered.com/irc>):

- *PowerPoint® slides* containing all the code and figures in the text.
- *Test Item File* of short-answer questions.
- *Solutions Manual* with solutions to the end-of-chapter **short-answer exercises** for *both* the Java and Android content. For the Java content, solutions are provided for *most* of the programming exercises.

The suggested Android app-development project exercises are *not* typical homework problems. These tend to be *substantial* projects—many of which could require weeks of effort, possibly with students working in teams. *Selected solutions only* are provided for these project exercises—these will be available on the Pearson Instructor's Resource Center (IRC) for fall semester 2014 classes. Contact us at [deitel@deitel.com](mailto:deitel@deitel.com) if you have any questions.

Please do not write to us requesting access to the Pearson Instructor's Resource Center. Access is restricted to **qualified college instructors teaching from the book**. Instructors may obtain access *only* through their Pearson representatives. If you're not a registered faculty member, contact your Pearson representative or visit <http://www.pearsonhighered.com/educator/relocator/>.

## **Before You Begin**

For information configuring your computer so that you can develop apps with Java and Android, see the Before You Begin section that follows this Preface.

## **Acknowledgments**

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

This book was a cooperative effort between the academic and professional divisions of Pearson. We 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 managed the book's publication. We selected the cover art and Marta Samsel designed the cover.

We also 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 Boe-



digheimer recruited and managed the professional reviewers for the Android content. John Fuller manages the production of all of our Deitel Developer Series books.

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 How to Program and Android for Programmers: An App-Driven Approach 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 University 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).

### ***Reviewers of the Content from Java How to Program Recent Editions***

Lance Andersen (Oracle), Soundararajan Angusamy (Sun Microsystems), Joseph Bowbeer (Consultant), William E. Duncan (Louisiana State University), Diana Franklin (University of California, Santa Barbara), Edward F. Gehringer (North Carolina State University), Huiwei Guan (Northshore Community College), Ric Heishman (George Mason University), Dr. Heinz Kabutz (JavaSpecialists.eu), Patty Kraft (San Diego State University), Lawrence Premkumar (Sun Microsystems), Tim Margush (University of Akron), Sue McFarland Metzger (Villanova University), Shyamal Mitra (The University of Texas at Austin), Peter Pilgrim (Consultant), Manjeet Rege, Ph.D. (Rochester Institute of Technology), Manfred Riem (Java Champion, Consultant, Robert Half), Simon Ritter (Oracle), Susan Rodger (Duke University), Amr Sabry (Indiana University), José Antonio González Seco (Parliament of Andalusia), Sang Shin (Sun Microsystems), S. Sivakumar (Astra Infotech Private Limited), Raghavan “Rags” Srinivas (Intuit), Monica Sweat (Georgia Tech), Vinod Varma (Astra Infotech Private Limited) and Alexander Zuev (Sun Microsystems).

As you read the book, we'd sincerely appreciate your comments, criticisms and suggestions for improving the text. Please address all correspondence to:

`deitel@deitel.com`

We'll respond promptly. We really enjoyed writing this book—we hope you enjoy reading 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 50 years of experience in the computer field. Dr. Deitel earned B.S. and M.S. degrees in Electrical Engineering from MIT and a Ph.D. in Mathematics from Boston University. 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 Programming Languages Training

Deitel & Associates, Inc., founded by Paul Deitel and Harvey Deitel, is an internationally recognized authoring and corporate training organization, specializing in computer programming languages, object technology, mobile app development and Internet and web software technology. The company's training clients include many of the world's largest companies, 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™, XML®, C++, C, Visual C#®, Visual Basic®, Visual C++®, 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 college textbooks and professional books in print and a wide range of electronic formats and *LiveLessons* video courses. Deitel & Associates, Inc. and the authors can be reached at:

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

<http://www.deitel.com/training>

To request a proposal for worldwide on-site, instructor-led training at your organization, e-mail [deitel@deitel.com](mailto:deitel@deitel.com).

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

<http://www.informit.com/store/sales.aspx>

