

Preface

*Science and technology and the various forms of art,
all unite humanity in a single and interconnected system.*
—Zhores Aleksandrovich Medvede

Live in fragments no longer, only connect.
—Edgar Morgan Foster

Build a better mousetrap, and the world will beat a path to your door.
—Ralph Waldo Emerson

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

Android How to Program

With this unique book—the 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 introduction to the Java core concepts that you'll need when programming Android apps.

Android How to Program was formed by merging

- our professional book *Android for Programmers: An App-Driven Approach*
- condensed core content on Java and object-oriented programming from our college textbook *Java How to Program, 9/e*
- approximately 700 new Android short-answer and app-development exercises we created for this book—most are in the book and approximately 200 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 documentation and forums where you can find additional information. We'll also encourage you to read, modify and enhance open-source code as part of your learning process.

xx Preface**Intended Audiences**

There are several audiences for this book. Most commonly, the book 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 who want to learn Android app development.

Most uniquely, though, the book may be used in introductory courses like CS1, intended for programming novices. We recommend that schools that typically offer many sections of CS1 in Java consider offering one or two sections to ambitious students who have at least some prior programming experience and who want to work hard to master both Java and Android in an aggressively paced one-semester or two-quarter course. The schools may want to list the courses with “honors” or “accelerated” designations.

App-Development Courses

A few years ago, Stanford offered a new course called Creating Engaging Facebook Apps (www.stanford.edu/group/captology/cgi-bin/facebook/). Students worked in teams developing apps, some of which landed in Facebook’s top 10, earning some of the student developers millions of dollars.¹ 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. We encourage you to read the online syllabi and check out the YouTube videos created by instructors and students for many of these courses.

Android Marketplace: 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. By October 2011, a comScore study showed that Android had over 46% of the U.S. smartphone market share, compared to 28% for Apple’s iPhone and 17% for Blackberry.²

Billions of Android apps have been downloaded from Android Market. More than 700,000 Android devices are being activated daily.³ The opportunities for Android app developers are enormous. This book will give you what you need to create, market and monetize your own Android apps.

The demand for mobile devices is increasing as more people rely on smartphones and tablets to stay connected and be productive while away from their personal computers. According to comScore, 234 million Americans used mobile devices in a three-month period ending in July 2011. Of those subscribers, 40.6% used apps.⁴

Fierce competition among popular mobile platforms (Android, iPhone, BlackBerry, Windows Phone 7 and others) and among mobile communications carriers is leading to rapid innovation and falling prices. Competition among the dozens of Android device

-
1. www.businessinsider.com/these-stanford-students-made-millions-taking-a-class-on-facebook-2011-5
 2. www.comscore.com/Press_Events/Press_Releases/2011/12/comScore_Reports_October_2011_U.S._Mobile_Subscriber_Market_Share
 3. www.informationweek.com/news/mobility/smart_phones/232300932
 4. www.comscore.com/Press_Events/Press_Releases/2011/8/comScore_Reports_July_2011_U.S._Mobile_Subscriber_Market_Share

manufacturers is driving hardware and software innovation within the Android community. There are now over 300 different Android devices.

App-Driven Approach

At the heart of the book is our *app-driven approach*. Rather than using code snippets, we present concepts in the context of *12 complete working Android apps* in the print book and several more online. Most of the apps were developed for the native Android environment; one of the online apps is in HTML5 for the *portable* world of the web—this app runs in a browser on Android *and* iPhone/iPad devices.

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*. 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 www.deitel.com/books/androidhttp/ and at the book’s Companion Website www.pearsonhighered.com/deitel/. The Companion Website contains *additional app-development chapters*. Figures 1–2 list the book’s apps and the key technologies we used to build each.

App	Technologies
<i>App-Development Chapters in the Print Book</i>	
Chapter 3, Welcome App	Dive-Into® Eclipse and the ADT
Chapter 4, Tip Calculator App	Building an Android App with Java
Chapter 5, Favorite Twitter® Searches App	Collections, Widgets and Views
Chapter 6, Flag Quiz Game App	Intents and Menus
Chapter 7, Cannon Game App	Frame-By-Frame Animation and Handling User Events
Chapter 8, Spot-On Game App	Tweened Animation and Listening for Touches
Chapter 9, Doodlz App	Graphics and Accelerometer
Chapter 10, Address Book App	AdapterViews and Adapters
Chapter 11, Route Tracker App	Maps API and Compass
Chapter 12, Slideshow App	Photos and Audio Library Access
Chapter 13, Enhanced Slideshow App	Serializing Objects and Playing Video
Chapter 14, Weather Viewer App	Internet-Enabled Applications, Web Services and App Widgets

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

Android How to Program

Android How to Program was fun to write! We developed lots of Android apps. The book’s apps were carefully designed to introduce you to a broad range of Android features and related technologies, including audio, video, animation, telephony, Bluetooth®, speech recognition, the accelerometer, GPS, the compass, widgets, App Widgets, 3D graphics and more. You’ll quickly learn what you’ll need to start building Android apps—begin-

App	Technologies
<i>App-Development Chapters Online</i>	
Chapter 15, Pizza Ordering App	Android Telephony and Speech APIs
Chapter 16, Voice Recorder App	Audio Recording and Playback
Chapter 17, Enhanced Address Book App	Managing Persistent Data with SQLite 3 and Transferring Data Via Bluetooth
Chapter 18, 3D Art App	3D Graphics and Animation with OpenGL ES
Chapter 19, Favorite Twitter® Searches App using HTML5 Technologies	Bonus Chapter: HTML5, CSS3 and JavaScript for Experienced Web Developers
Chapter 20, Android 4 App	Under development—an Android 4 app that scales nicely for smartphone and tablet devices.

Fig. 2 | *Android How to Program* apps in the online chapters.

ning with a test-drive of the **Doodlz** app in Chapter 1, then creating your first app in Chapter 3. Chapter 2, **Android Market and App Business Issues** walks you through designing great apps, uploading your apps to Google’s Android Market and other online app stores, deciding whether to sell your apps or offer them for free, and marketing them using the Internet and word-of-mouth, and more.

Staying in Contact with the Authors

As you read the book, we’d appreciate your comments, criticisms, corrections and suggestions for improvement. Please address all correspondence to deitel@deitel.com—we’ll respond promptly. For updates on this book, visit www.deitel.com/books/androidhttp/; follow us on Facebook (www.deitel.com/deitelfan), Twitter (@deitel) and Google+ ([plus.to/deitel](https://plus.google.com/+deitel)) and subscribe to the *Deitel® Buzz Online* newsletter (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 (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.

Getting up to Speed in Java and XML

The Android portion of this book assumes that you’re a Java programmer with object-oriented programming experience and that you’re familiar with XML.

If you're not familiar with Java, the appendices provide a condensed, friendly introduction to the Java technologies you'll need to develop Android apps. If you're interested in learning Java in more depth, you may want to check out the more comprehensive treatment in our textbook *Java How to Program, 9/e* (www.deitel.com/books/jhttp9/).

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

- docs.oracle.com/javase/1.4/tutorial/doc/IntroXML2.html
- www.ibm.com/developerworks/xml/newto/
- www.w3schools.com/xml/xml_what_is.asp
- www.deitel.com/articles/xml_tutorials/20060401/XMLBasics/
- www.deitel.com/articles/xml_tutorials/20060401/XMLStructuringData/

Key Features

- **Android Smartphone Apps.** We cover many of the features included in the Android Software Development Kit (SDK), including Bluetooth, Google Maps, the Camera APIs, graphics APIs and support for multiple screen sizes and resolutions.
- **Android Tablet Apps.** We cover many Android features for developing tablet apps, including property animation, action bar and fragments.
- **Android Maps APIs.** The **Route Tracker** App uses the Android Maps APIs which allow you to incorporate Google™ Maps in your app. *Before developing any app using the Maps APIs, you must agree to the Android Maps APIs Terms of Service (including the related Legal Notices and Privacy Policy) at code.google.com/android/maps-api-tos.pdf.*
- **Eclipse.** The free Eclipse integrated development environment (IDE) combined with the free Android SDK and the free Java Development Kit (JDK), provide everything you'll need to develop and test Android apps.
- **Testing on Android Smartphones, Tablets and the Android Emulator.** For the best experience in this course, you should test your apps on actual Android smartphones and tablets. But you can still have a meaningful experience just using the Android emulator (see the Before You Begin section).
- **Multimedia.** The apps use a broad range of Android multimedia capabilities, including graphics, images, frame-by-frame animation, property animation, audio, video, speech synthesis and speech recognition.
- **Android Best Practices.** We adhere to accepted Android best practices, pointing them out in the detailed code walkthroughs. Check out our Android Best Practices Resource Center at www.deitel.com/AndroidBestPractices/.
- **Web Services.** Web services allow you to use the web as an extraordinary collection of services—many of which are free. Chapter 11's **Route Tracker** app uses the built-in Android Maps APIs to interact with the Google Maps web services.⁵ Chapter 14's **Weather Viewer** app uses WeatherBug's web services.⁶ The exercises encourage you to explore the vast array of available web services.

5. code.google.com/apis/maps/documentation/webservices/

6. apireg.weatherbug.com/defaultAPI.aspx

xxiv Preface

- *(Early Objects) Java Content Can Be Used With Java SE 6 or Java SE 7.* The Java Standard Edition (SE) 7 features are in modular easy-to-include-or-omit sections. Studying objects and classes early helps novice readers master these concepts more thoroughly before attempting the object-oriented Android material.
- *Exception Handling.* We integrate basic exception handling early in the Java content then present a richer treatment in Appendix H.
- *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 and for which Android has some similar classes.
- *Multithreading.* Maintaining app responsiveness is a key to building robust Android apps and requires extensive use of Android multithreading. Appendix J introduces multithreading fundamentals, showing the features that we use in several of our Android apps. In addition, we present Java’s `SwingWorker` class for multithreading in GUI apps. Android’s `AsyncTask` class, which is used in several of our Android apps, closely parallels the `SwingWorker` class.
- *GUI Presentation.* Appendix I introduces Java GUI development. Android provides its own GUI components, so this appendix presents only a few Java GUI components, focussing on event-handling techniques that are used in all Android GUIs. The appendix introduces nested classes and anonymous inner classes, which are frequently used in Android programming.

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 these apps and read the 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 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 place defining occurrences of key terms in **bold** text for easy reference. We identify on-screen components in the **bold Helvetica** font (e.g., the **File** menu) and Java and Android program text in the **Lucida** font (e.g., `int x = 5;`). In this book you'll create GUIs using a combination of visual programming (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 an app, we place its variable name and class name in a **Lucida** font—e.g., “**Button**” or “**myEditText**.”
- 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/androidhtp
www.pearsonhighered.com/deitel

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

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

Software Engineering. We stress program clarity 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 almost 200 topical Android app-development exercises. You'll develop apps using a broad array of current technologies, even including multiplayer social gaming, mashups, speech synthesis and recognition, location-based services, web services, database, open source, and a variety of multimedia capabilities.

The Android exercises include hundreds of short-answer fill-in and true/false questions. 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.

In the Java exercises, you'll be asked to recall important terms and concepts; indicate what some code does; 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**.

xxvi Preface

Software Used in *Android How to Program*

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 developer.android.com and www.oracle.com/technetwork/java/javase/downloads/index.html. The documentation for Eclipse is available at www.eclipse.org/documentation.

Instructor Resources

The following supplements are available to *qualified college instructors only* through Pearson Education's Instructor Resource Center (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. Solutions are not provided for the suggested Android app-development project exercises.

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 www.pearsonhighered.com/educator/replocator/.

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. If you're starting with the Java content, also see the Test Drive posted at www.deitel.com/books/androidhttp.

The Deitel Online Android and Java Resource Centers

Our Android Resource Centers include links to tutorials, documentation, software downloads, articles, blogs, podcasts, videos, code samples, books, e-books and more. Check out the growing list of Android-related Resource Centers, including:

- Android (www.deitel.com/android/)
- Android Best Practices (www.deitel.com/androidbestpractices/)
- Java (www.deitel.com/java/)
- Eclipse (www.deitel.com/Eclipse/)
- SQLite 3 (www.deitel.com/SQLite3/)

CourseSmart Web Books

Students and instructors have increasing demands on their time and money. Pearson has responded by offering digital texts and course materials online through CourseSmart. Fac-

ulty can now review course materials online. Students can access a digital version of a text for less than the cost of a print book and can see the same content as in the print textbook enhanced by search, note-taking and printing tools. For detailed information on the CourseSmart version of *Android How to Program*, visit www.coursesmart.com.

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.

We'd like to thank Michael Morgano, co-author of our professional book *Android for Programmers: An App-Driven Approach*. Michael is a graduate of Northeastern University with B.S. and M.S. degrees in computer science and works as a professional Android developer. Michael also co-authored the first edition of our professional book *iPhone for Programmers: An App-Driven Approach*.

We're fortunate to have worked with the teams of academic and professional publishing professionals at Pearson/Prentice Hall. We appreciate the guidance, savvy and energy of Michael Hirsch, Editor-in-Chief of Computer Science. Michael and his team handle all of our academic publications. Carole Snyder and Bob Engelhardt have done a marvelous job managing the review and production processes, respectively, for the last several editions of *Java How to Program*; Bob did a great job bringing all the pieces together for *Android How to Program*.

We also appreciate the efforts and 16-year mentorship of our friend and professional colleague Mark L. Taub, Editor-in-Chief of Pearson Technology Group. Mark and his team handle all of our professional books and LiveLessons video products. Olivia Basegio did a great job recruiting distinguished members of the Android community and managing the review team for the Android content.

We'd like to thank our friend, Rich Wong (Partner, Accel Partners), who provided us with valuable contacts in the Android development community.

Thanks also to AWS Convergence Technologies, Inc., owners of WeatherBug (weather.weatherbug.com/), for giving us permission to use their web services in Chapter 14's **Weather Viewer** app.

We'd also like to thank our colleague, Eric Kern, a Computer Engineering major at Northeastern University, co-author of our related book, *iPhone for Programmers: An App-Driven Approach*, on which many of the apps in *Android How to Program* are based.

Reviewers

We wish to acknowledge the efforts of our reviewers.

Reviewers of the Content from Android for Programmers: An App-Driven Approach

Paul Beusterien (Principal, Mobile Developer Solutions), Eric J. Bowden, COO (Safe Driving Systems, LLC), Ian G. Clifton (Independent Contractor and Android App Developer), Daniel Galpin (Android Advocate and author of *Intro to Android Application Development*), Douglas Jones (Senior Software Engineer, Fullpower Technologies), Sebastian Nykopp (Chief Architect, Reaktor) and Ronan "Zero" Schwarz (CIO, OpenIntents).

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 (Univer-

xxviii Preface

sity 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).

We hope you enjoy working with *Android How to Program* as much as we enjoyed writing it. We’re looking forward to hearing about your app-development successes!

Paul, Harvey and Abbey Deitel, January 2012

About the Authors

Paul Deitel, CEO and Chief Technical Officer of Deitel & Associates, Inc., is a graduate of MIT, where he studied Information Technology. Through Deitel & Associates, Inc., he has delivered hundreds of programming courses to industry clients, including Cisco, IBM, Siemens, Sun Microsystems, Dell, Lucent Technologies, Fidelity, NASA at the Kennedy Space Center, the National Severe Storm Laboratory, White Sands Missile Range, Rogue Wave Software, Boeing, SunGard Higher Education, Stratus, Cambridge Technology Partners, One Wave, Hyperion Software, Adra Systems, Entergy, CableData Systems, 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 from MIT and a Ph.D. 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 Chinese, Korean, Japanese, German, Russian, Spanish, French, Polish, Italian, Portuguese, Greek, Urdu and Turkish. Dr. Deitel has delivered hundreds of professional programming seminars to major corporations, academic institutions, government organizations and the military.

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 14 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*, *iPhone for Programmers: An App-Driven Approach*, *Internet & World Wide Web How to Program, 5/e* and *Simply Visual Basic 2010, 5/e*.

Corporate Training from Deitel & Associates, Inc.

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

Through its 36-year publishing partnership with Prentice Hall/Pearson, Deitel & Associates, Inc., publishes leading-edge programming college textbooks, professional books 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 company or 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.pearsoned.com/professional/index.htm.

