



PREFACE



Welcome to Java, the Internet and World-Wide-Web programming! This book, the second in our new *Simply* series, was a joy to create. Our goal was to write a book that focuses on core concepts and features of Java while keeping the discussion of this highly technical subject as simple as possible.

To achieve these goals, we implemented an innovative teaching methodology. We present the core concepts of leading-edge computing technologies using the tutorial-based, APPLICATION-DRIVEN approach, combined with the DEITEL® signature LIVE-CODE approach of teaching programming using complete, working, real-world applications. We merged the notion of a lab manual with that of a conventional textbook, creating a book works well in a traditional classroom setting or with students sitting at computers and building each example application as they read the tutorials.

As students work through the tutorials, they learn about Java and its fundamental features, such as graphical-user-interface (GUI) components, multimedia (audio, images, animation and video), file processing, database processing and Internet and World-Wide-Web-based client/server networking. At the end of most sections, we provide self-review questions with answers so that students receive immediate feedback on their understanding of the material. Hundreds of additional self-review questions with answers are available on this book's Companion Web Site.

Features in *Simply Java Programming*

This book is loaded with pedagogic features, including:

- **APPLICATION-DRIVEN Tutorial Approach.** Each tutorial uses a contemporary, real-world application to teach programming concepts. The examples and exercises are up-to-the-minute with Internet/Web-related examples and with popular applications, such as game playing, graphics, multimedia and even a three-tier Web-based bookstore. Most examples have a business focus. At the beginning of each tutorial, students “test-drive” the completed application so they can see how it works. Then, they build the application by following step-by-step instructions. The book concentrates on the principles of good software design and stresses program clarity.
- **LIVE-CODE Approach.** This book uses LIVE-CODE examples. Each tutorial ends with the complete, working application code and the students can run the application that they just created. We call this method of teaching and writing the **LIVE-CODE Approach**. We feel that this approach is more effective than presenting only snippets of code out of the context of a complete application.
- **Real-World Technologies.** This text incorporates today's technologies to develop useful applications. For example, we use the Unified Modeling Language™ (UML) to replace flowcharts—an older standard. The UML has become the preferred graphical modeling language for designing object-oriented applications. In *Simply Java*, we use UML to show the flow of control for several applications, so students gain practice reading the type of diagrams that are used in industry.

- **Graphical User Interface (GUI) Programming.** From the first tutorial, we immerse students in GUI programming techniques and modifying Java GUIs. Students who learn these techniques can create graphical applications more quickly and easily. The early tutorials provide students with a foundation for designing GUIs—concepts that they will apply throughout the book as we teach core programming concepts. Many tutorials contain *GUI Design Tips* that are summarized at the end of the tutorials for easy reference. Additionally, Appendix C, GUI Design Guidelines, compiles these tips to help students as they prepare for exams.
- **Full-Color Presentation.** This book is in full color so that students can see sample outputs as they would appear on a monitor. Also, we syntax color the Java code, similar to the way Java integrated development environments (IDEs) color the code in their editor windows. This way, students can match what they see in the book with what they see on their screens. Our syntax-coloring conventions are as follows:

```
comments appear in green
keywords appear in dark blue
literal values and constants appear in light blue
text, class, method and variable names appear in black
errors appear in red
```

- **Graphics and Multimedia.** Graphics and multimedia make applications fun to create and use. In our introduction to graphics, Tutorial 20, we discuss basic concepts and features of graphics. Part of Java's initial appeal was that it supported graphics, enabling Java programmers to visually enhance their applications. You will learn several of Java's capabilities for drawing two-dimensional shapes and controlling colors. In Tutorial 27, we expand our discussion of graphics by introducing additional methods of the `Graphics` class to outline and fill in different types of shapes. In Tutorial 28, you will explore the Java Speech API, which produces synthetic speech from text inputs. You will create a phone book application in which the user selects a name and the application speaks the corresponding phone number.
- **Databases.** Databases are crucial to businesses today, and we use real-world applications to teach the fundamentals of database programming. Tutorial 26 and Tutorial 31 familiarize students with databases, presented in the context of two applications—an ATM and a three-tier Web-based bookstore. In Tutorial 26, you will learn how to connect to a database and retrieve information from a database using the JDBC API.
- **Case Study.** This book concludes with a sequence of four tutorials in which the student builds a Web-based bookstore application. In Tutorial 29, you will learn the multi-tier architecture that is used to create Web applications. You will learn about Web servers and install the Apache Tomcat Web server, which you will need to run your bookstore application. In Tutorial 30, you will use HTML to create the client tier (also called the top tier)—the user interface of your application. In Tutorial 31, you will create the application's information tier and create the connections with your database as well as use SQL statements to obtain information from the database. Finally in Tutorial 32, you will create the middle tier of the Web-based bookstore and complete the application.
- **Object-Oriented Programming.** Object-oriented programming is the most widely employed technique for developing robust, reusable software, and Java offers advanced object-oriented programming features. This book introduces students to declaring classes and using objects, laying a solid foundation for future programming courses.

- **Java Debugger.** The Java 2 Software Development Kit (J2SDK) provides software called a debugger, which allows you to analyze the behavior of your applications to locate logic errors. At the ends of several tutorials, we provide *Using the Debugger* sections in which you will learn to detect and remove logic errors by using the Java debugger.

Notes to the Instructor *Focus of the Book*

Our goal was clear: Produce a Java textbook for introductory-level courses in computer programming for students with little or no programming experience. This book teaches computer programming principles and the Java language, including data types, control statements, object-oriented programming, classes, GUI concepts, event-driven programming, graphics, database, Web-applications development and more. After mastering the material in this book, students will be able to program in Java and to employ its cross-platform capabilities.

Lab Setup

Before you can compile and run the applications in this book, the Java 2 Software Development Kit (J2SDK), or an appropriate Java development tool, must be installed. We discuss installing the J2SDK in the *Before You Begin* section that follows the Preface. For computer labs in which students are not allowed to install software, instructors and system administrators must ensure that appropriate Java software is installed on the lab computers in advance of the course. Several tutorials require additional software. Tutorial 26 requires IBM's Cloudscape database software, which is included on the CD that accompanies this book. The Cloudscape installation instructions appear in Tutorial 26. Tutorial 28 uses speech synthesis software, which must be installed to run and develop the **Phone Book** application. Download and installation instructions for this software appear in Tutorial 28. Configuring and executing the **Bookstore** case study in Tutorials 29–32 require IBM's Cloudscape database software and Apache Tomcat software, which is also included on the CD that accompanies this book. Installation instructions for Apache Tomcat appear in Tutorial 29. [Note: For instructors and students who prefer to use Microsoft Access for the database applications, we will post instructions at www.deitel.com/books/simplyJava1/index.html.]

Note Regarding the Platform We Used to Develop the Book

We assume that students are using Windows platform computers (Windows 2000 or Windows XP, in particular), so all directory names, instructions and sample screen captures appear in Windows format. However, the instructions and concepts presented work well on most computer platforms. [Note: All windows that show source code were created in Sun™ ONE Studio 4 Community Edition. © Copyright 2003 Sun Microsystems, Inc. All rights reserved. Used by permission.]

Note Regarding Terminology Used for Event Handlers in the Book

Each event handler in this book calls another method, which actually contains the code that processes the event. For a JButton named `calculateJButton`, for example, the event handler `actionPerformed` calls method `calculateJButtonActionPerformed` to process the event. We implemented the code in this manner so that our code would be similar to the code that is generated by the GUI designers provided with many of today's popular Java IDEs. In the early chapters, we refer to methods such as `calculateJButtonActionPerformed` as “event handlers.” Chapter 13 presents a more thorough introduction to event handling. At that point, we refer to methods like `calculateJButtonActionPerformed` as “methods” and use the term “event handler” only for those methods that are declared by event-listener interfaces.

Objectives

Each tutorial begins with objectives that inform students of what to expect and give them an opportunity, after reading the tutorial, to determine whether they have met the intended goals.

Outline

The tutorial outline enables students to approach the material in top-down fashion. Along with the tutorial objectives, the outline helps students anticipate future topics and set a comfortable and effective learning pace.

Example Applications (with Application Outputs)

We present Java features in the context of complete, working Java applications. We call this our LIVE-CODE approach. All examples are available on the CD that accompanies the book or as downloads from our Web site, www.deitel.com/books/simplyJava1/index.html.

Illustrations/Figures

An abundance of charts, line drawings and application outputs are included. The discussion of control statements, for example, features carefully drawn UML activity diagrams. [Note: We do not teach UML diagramming as a program-development tool, but we do use UML diagrams to explain the precise operation of many of Java's control statements.]

Programming Tips

Hundreds of programming tips help students focus on important aspects of application development. These tips and practices represent the best the authors have gleaned from a combined seven decades of programming and teaching experience.

**Good Programming Practices**

Good Programming Practices highlight techniques that help students write applications that are clearer, more understandable and more maintainable.

**Common Programming Errors**

Students learning a language—especially in their first programming course—frequently make errors. Pointing out these *Common Programming Errors* in the text reduces the likelihood that students will make the same mistakes.

**Error Prevention Tips**

These tips describe aspects of Java that prevent errors from getting into applications in the first place, which simplifies the testing and debugging process.

**GUI Design Tips**

The *GUI Design Tips* highlight graphical-user-interface conventions to help students design attractive, user-friendly GUIs and use GUI features.

**Performance Tips**

Teaching students to write clear and understandable applications is the most important goal for a first programming course. But students want to write applications that run the fastest, use the least memory, require the smallest number of keystrokes, etc. *Performance Tips* highlight opportunities for improving application performance.



Portability Tips

The *Portability Tips* provide insights on how Java achieves its high degree of portability among different platforms.



Software Design Tips

The *Software Design Tips* highlight architectural and design issues that affect the construction of object-oriented software systems.

Skills Summary

Each tutorial includes a bullet-list-style summary of the new programming concepts presented. This reinforces key actions taken to build the application in each tutorial.

Key Terms

Each tutorial includes a list of important terms defined in the tutorial. These terms also appear in the index and in a book-wide glossary, so the student can locate terms and their definitions quickly.

240 Self-Review Questions and Answers

Self-review multiple-choice questions and answers are included after most sections to build students' confidence with the material and prepare them for the regular exercises. Students should be encouraged to attempt all the self-review exercises and check their answers.

506 Exercises (Solutions in Instructor's Manual)

Each tutorial concludes with exercises. Typical exercises include 10 multiple-choice questions, a "What does this code do?" exercise, a "What's wrong with this code?" exercise, three regular programming exercises and a programming challenge. [*Note:* In the "What does this code do?" and "What's wrong with this code?" exercises, we show only portions of the code in the text, but the instructor's manual contains full applications with outputs.] Several tutorials also include exercises that require students to use the Java debugger to locate and fix logic errors in applications.

The questions involve simple recall of important terminology and concepts, writing individual Java statements, writing small portions of Java applications and writing complete Java methods, classes and applications. Every programming exercise uses a step-by-step methodology to guide the student. The solutions for the exercises are *available only to instructors* through their Prentice-Hall representatives. [**NOTE: Please do not write to us requesting the instructor's manual. Distribution of this publication is strictly limited to instructors teaching from the book. Instructors may obtain the solutions manual only from their regular Prentice Hall representatives. We regret that we cannot provide the solutions to professionals.**]

GUI Design Guidelines

Consistent and proper graphical user interface design is crucial to visual programming. In each tutorial, we summarize the GUI design guidelines that were introduced. Appendix C, GUI Design Guidelines, presents a cumulative list of these GUI design guidelines for easy reference.

Java Library Reference Summaries

Each tutorial includes a summary of the components, classes, methods and other Java Library objects discussed in the tutorial. The summary includes a picture of each component, shows the component "in action" and lists the component's properties, events and methods that were discussed up to and including that tutorial. In addition, Appendix D groups the controls by tutorial for easy reference.

Index

The extensive index includes important terms both under main headings and as separate entries so that students can search for any term or concept by keyword. The code examples and the exercises are also included in the index. For every Java source code application in the book, we indexed it both under the appropriate application and as a subindex item under “applications.” We have also double-indexed features such as components, methods and classes. This makes it easier to find examples using particular features.

Simply Java Ancillary Package

Simply Java is accompanied by extensive ancillary materials for instructors, including the following:

- *Instructor’s Resource CD (IRCD)* which contains the
 - *Instructor’s Manual* with solutions to the end-of-tutorial exercises and
 - *Test-Item File* of multiple-choice questions (approximately two per tutorial section).
- *Customizable PowerPoint® Slides* containing all the code and figures in the text, and bulleted items that summarize the key points in the text. The slides are downloadable from www.deitel.com/books/simplyJava1/index.html and are available as part of Prentice Hall’s *Companion Web Site* for *Simply Java Programming*, which offers resources for both instructors and students. The *Companion Web Site* is located at www.prenhall.com/deitel.

Companion Web Site

For instructors, the *Companion Web Site* offers a *Syllabus Manager*, which helps instructors plan courses interactively and create online syllabi. Students also benefit from the functionality of the *Companion Web Site*. Book-specific resources for students include:

- PowerPoint® slides
- Example source code
- Reference materials from the book appendices
- Tutorial objectives
- Tutorial summaries
- Tutorial outlines
- Programming tips from each tutorial
- Online Study Guide—contains additional short-answer self-review exercises with answers
- Students can track their results and course performance on quizzes using the *Student Profile* feature, which records and manages all feedback and results from tests taken on the *Companion Web Site*. To access the *Companion Web Site* for *Simply Java Programming*, visit www.prenhall.com/deitel.

Course Management Systems

Selected content from *Simply Java Programming* and other Deitel texts, is available to integrate into various Course Management Systems, including CourseCompass, Blackboard and WebCT. Course Management Systems help faculty create, manage and use sophisticated Web-based educational tools and programs. Blackboard, CourseCompass and WebCT offer:

- Features to create and customize an online course
- Communication tools
- Flexible testing tools
- Support materials

In addition to the tools found in Blackboard and WebCT, CourseCompass from Prentice Hall includes:

- **CourseCompass course home page**, which makes the course as easy to navigate as a book.
- **Hosting on Prentice Hall’s centralized servers**, which allows course administrators to avoid separate licensing fees or server-space issues.
- **“How Do I” online-support sections** are available for users who need help personalizing course sites.
- **Instructor Quick Start Guide**

Acknowledgments

One of the great pleasures of writing a textbook is acknowledging the efforts of many people whose names may not appear on the cover, but whose hard work, cooperation, friendship and understanding were crucial to the production of the book. Many people at Deitel & Associates, Inc., devoted long hours to this project.

- Abbey Deitel, President
- Barbara Deitel, Chief Financial Officer
- Christi Kelsey, Director of Business Development

We would also like to thank several of the participants in the Deitel & Associates, Inc., College Internship Program: Adam Burke, Connor Dutson, Nathan Larson, Brian O’Connor, Mike Oliver and Nick Cassie.¹

We are fortunate to have been able to work on this project with the talented and dedicated team of publishing professionals at Prentice Hall. We especially appreciate the extraordinary efforts of our Computer Science Editor, Kate Hargett and her boss and our mentor in publishing—Marcia Horton, Editorial Director of Prentice-Hall’s Engineering and Computer Science Division. Tom Manshreck and Vince O’Brien did a marvelous job managing the production of the book. Chirag Thakkar and John Lovell served as production editors and Sarah Parker handled editorial responsibilities on the book’s extensive ancillary package.

We wish to acknowledge the efforts of our reviewers and to thank Carole Snyder of Prentice Hall, who managed the review process. The 37 reviewers from colleges, industry and Sun Microsystems helped us to get this book “right.” Adhering to a tight time schedule, these reviewers scrutinized the text and the applications, providing countless suggestions for improving the accuracy and completeness of the presentation. It is a privilege to have the guidance of such talented and busy professionals.

Simply Java Programming reviewers:

Alireza Fazelpour (Palm Beach Community College)
 Amardeep Kahlon (Austin Community College)
 Andrea Shelly (Florida International University)
 Andy Mortensen (Southern Connecticut State University)
 Annette Schoenberger (St. Cloud State University)
 Ayad Boudiab (Georgia Perimeter College)
 Balaji Janamanchi (Texas Tech University)
 Brian Larson (Modesto Junior College)

1. The Deitel & Associates, Inc. College Internship Program offers a limited number of salaried positions to college students majoring in Computer Science, Information Technology, Marketing and English. Students work at our corporate headquarters in Maynard, Massachusetts full-time in the summers and (for those attending college in the Boston area) part-time during the academic year. We also offer full-time internship positions for students interested in taking a semester off from school to gain industry experience. Regular full-time positions are available to college graduates. For more information, please contact Abbey Deitel at deitel@deitel.com, visit our Web site, www.deitel.com and subscribe to our free e-mail newsletter at www.deitel.com/newsletter/subscribe.html.

Carol Buse (Amarillo College)
Catherine Wyman (DeVry University)
Charles Cadenhead (Brookhaven College)
Charles Lake (Faulkner State College)
Christopher Crane (Cisco Systems, Inc.)
Clint Bickmore (Front Range Community College)
Craig Slinkman (University of Texas at Arlington)
Darrel Karbginsky (Chemeketa Community College)
David Zeng (DeVry Institute of Technology, Calgary)
Deborah Shapiro (Cittone Institute)
Doug Kohlert (Sun Microsystems)
Edmund Weihrauch (Community College of Allegheny County)
Elizabeth Branca (Westchester Community College)
Gavin Osborne (Saskatchewan Institute)
Hong Lin (DeVry University)
James Huddleston (Independent Consultant)
Judith Ashworth (Orillion USA, Inc.)
Kristin Enders (Grand Rapids Community College)
Loran Walker (Lawrence Technological University)
Manu Gupta (Patni Computer Systems)
Merrill Parker (Chattanooga State Technical Community College)
Michael Newby (California State University, Fullerton)
Paul McLachlan (Compuware Corporation)
Rama Roberts (Sun Microsystems)
Rekha Bhowmik (Winona State University)
Sachin Korgaonkar (Idealake Technologies Ltd.)
Sergio Davalos (University of Washington- Tacoma)
Susan Fry (Boise State University)
Terry Hull (Sun Certified Architect, Rational Qualified Practitioner)

We would sincerely appreciate your comments, criticisms, corrections and suggestions for improving this textbook. Please address all correspondence to:

deitel@deitel.com

We will respond promptly.

Well, that's it for now. Welcome to the exciting world of Java programming. We hope you enjoy this look at leading-edge computer applications development. Good luck!

Dr. Harvey M. Deitel

Paul J. Deitel

Jeff A. Listfield

Cheryl H. Yaeger

Su Zhang

About the Authors

Dr. Harvey M. Deitel, Chairman of Deitel & Associates, Inc., has 42 years experience in the computing field, including extensive industry and academic experience. Dr. Deitel earned B.S. and M.S. degrees from the Massachusetts Institute of Technology and a Ph.D. from Boston University. He worked on the pioneering virtual-memory operating-systems projects at IBM and MIT that developed techniques now widely implemented in systems such as UNIX, Linux and Windows XP. He has 20 years of college teaching experience and served as the Chairman of the Computer Science Department at Boston College before founding Deitel & Associates, Inc., with his son, Paul J. Deitel. He is the author or co-author of several dozen books and multimedia packages. With translations published in numerous foreign languages, Dr. Deitel's texts have earned international recognition. Dr. Deitel has

delivered professional seminars to major corporations, government organizations and various branches of the military.

Paul J. Deitel, CEO and Chief Technical Officer of Deitel & Associates, Inc., is a graduate of the Massachusetts Institute of Technology's Sloan School of Management, where he studied information technology. Through Deitel & Associates, Inc., he has delivered professional seminars to numerous industry and government clients and has lectured on C++ and Java for the Boston Chapter of the Association for Computing Machinery. He and his father, Dr. Harvey M. Deitel, are the world's best-selling Computer Science textbook authors.

Jeff A. Listfield is a Computer Science graduate of Harvard University. His coursework included classes in computer graphics, networks and computational theory and he has programming experience in several languages. Jeff has co-authored *C# How to Program*, *C# A Programmer's Introduction* and *C# for Experienced Programmers*, and contributed to *Perl How to Program* and *Java How to Program, 5/e*.

Cheryl H. Yaeger, Director of Microsoft Software Publications with Deitel & Associates, Inc., graduated from Boston University in three years with a bachelor's degree in Computer Science. Cheryl has co-authored various Deitel & Associates, Inc. publications, including *C# How to Program*, *C#: A Programmer's Introduction*, *C# for Experienced Programmers* and *Visual Basic.NET for Experienced Programmers*. Cheryl also contributed to other Deitel & Associates publications including *Perl How to Program*, *Wireless Internet and Mobile Business How to Program*, *Internet and World Wide Web How to Program, 2/e*, *Visual Basic .NET How to Program, 2/e* and *Simply Visual Basic .NET*.

Su Zhang holds B.Sc. and M.Sc. degrees in Computer Science from McGill University. Her graduate research included modeling and simulation, real-time systems and Java technology. She worked on Java and Web-technologies-related projects prior to joining Deitel. She has contributed to several Deitel Publications including *Advanced Java 2 Platform How to Program*, *Python How to Program*, *Java How to Program, 5/e* and she co-authored *Java Web Services for Experienced Programmers*.

About Deitel & Associates, Inc.

Deitel & Associates, Inc., is an internationally recognized corporate-training and content-creation organization specializing in computer programming languages education, object technology and Internet/World Wide Web software technology. Through its 27-year publishing partnership with Prentice Hall, Deitel & Associates, Inc. publishes leading-edge programming textbooks, professional books, interactive CD-ROM-based multimedia *Cyber Classrooms*, *Complete Training Courses* and course management systems e-content. To learn more about Deitel & Associates, Inc., its publications and its worldwide corporate on-site curriculum, visit:

www.deitel.com

Individuals wishing to purchase Deitel books, *Cyber Classrooms* and *Complete Training Courses* can do so through bookstores, or online booksellers and:

www.deitel.com
www.prenhall.com/deitel
www.InformIT.com/deitel

Bulk orders by corporations and academic institutions should be placed directly with Prentice Hall. For ordering information, please visit:

www.prenhall.com/deitel

The Deitel® Buzz Online e-Mail Newsletter

Our free e-mail newsletter includes commentary on industry trends and developments, links to articles and resources from our published books and upcoming publications, information on future publications, product-release schedules and more. For quick opt-in registration, visit www.deitel.com/newsletter/subscribe.html.