



REFACE



Welcome to Visual Basic .NET and the world of Windows, Internet and World-Wide-Web programming with Visual Studio and the .NET platform! This book, which is part of our new *Simply* series, has been updated based on Visual Studio .NET 2003. Our goal was to write a book that focuses on core concepts and features of Visual Basic .NET 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 that is best used by students sitting at computers and building each example application as they read the tutorials.

As students work through the tutorials, they learn about Visual Basic .NET and its fundamental features, such as visual programming concepts, 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 Visual Basic .NET 2003

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 ATMs, game playing, graphics, multimedia and even a 3-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 our step-by-step instructions. The book concentrates on the principles of good software engineering and stresses program clarity.
- **LIVE-CODE Approach.** This book contains several LIVE-CODE examples. Each tutorial ends with the complete, working program 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 Visual Basic .NET 2003*, we use UML to show the flow of control for several control statements, so students gain practice reading the type of diagrams that are used in industry.

- **Visual Programming and Graphical User Interface (GUI).** From the first tutorial, we immerse students in visual programming techniques and modifying Visual Basic .NET GUIs. Students who learn these techniques can create graphical programs 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. Appendix C compiles all the GUI Design 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 Visual Basic .NET code, similar to the way Visual Studio .NET colors the code in its editor window. This way, students can match what they see in the book with what they see on their own screens. Our syntax-coloring conventions are as follows:

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

- **Graphics and Multimedia.** Graphics make applications fun to create and use. In our introduction to graphics, Tutorial 26, we discuss Graphical Device Interface (GDI+)—the Windows service that provides the graphical features used by .NET—to teach students to personalize a bank check. In Tutorial 27, we use a fun technology called Microsoft Agent to add interactive, animated characters to a phone book application. With Microsoft Agent, your applications can speak to users and even respond to their voice commands!
- **Databases.** Databases are crucial to businesses today, and we use real-world applications to teach the fundamentals of database programming. Tutorials 25 and 30 familiarize students with databases, presented in the context of two applications—an ATM and a Web-based bookstore.
- **Case Study.** This book includes a sequence of four tutorials in which the student builds a Web-based bookstore application. Tutorial 28 familiarizes readers with Microsoft's Internet Information Services (which enables Web publishing), multi-tier architecture and simple Web transactions. Tutorials 29–31 use ASP.NET and ADO.NET to build an application that retrieves information from a database and displays the information in a Web page.
- **Object-Oriented Programming.** Object-oriented programming is the most widely employed technique for developing robust, reusable software, and Visual Basic .NET offers advanced object-oriented programming features. This book introduces students to defining classes and using objects, laying a solid foundation for future programming courses.
- **Visual Studio .NET Debugger.** Debuggers help programmers find and correct logic errors in program code. Visual Studio .NET 2003 contains a powerful debugging tool that allows programmers to analyze their programs line-by-line as they execute. Throughout the book, we teach the Visual Studio .NET 2003 Debugger; we explain how to use its key features and offer many debugging exercises.

To the Instructor Focus of the Book

Our goal was clear: Produce a Visual Basic .NET textbook for introductory-level courses in computer programming aimed at students with little or no programming experience. This book teaches computer programming principles and the Visual Basic .NET language, including data types, control statements, object-oriented programming, Visual Basic .NET classes, GUI concepts, event-driven programming and more. After mastering the material in this book, students will be able to program in Visual Basic .NET and to employ many key capabilities of the .NET platform.

We also wanted a textbook that was up-to-date with Microsoft's latest release of Visual Studio—Visual Studio .NET 2003, which includes an updated version of Visual Basic .NET. We have rebuilt every application in this book using the 2003 software. All applications and solutions have been fully tested and run on this new platform.

A Note Regarding Software for the Book

For the educational market only, this textbook is available in a “value pack” with the Microsoft® Visual Basic® .NET Standard Edition version 2003 integrated development environment as a free supplement. The standard edition is fully functional and is shipped on 5 CDs. There is no time limit for using the software. [*Note:* If you are a professional using this publication, you will have to purchase the necessary software to build and run the applications in this textbook.]

Lab Setup

To install some of the required software for this book, students and instructors will need Administrator-level access to the computer. For university computer labs where students do not have Administrator-level access, instructors and system administrators must ensure that the proper software is installed on the lab computers. In Tutorial 27 certain Microsoft Agent software components must be installed to execute and develop the **Phone Book** application. If students are not allowed to install software on lab computers, the Microsoft Agent components discussed in Tutorial 27 must be installed in advance. To configure and execute some of the examples and exercises, such as the **Bookstore** case study in Tutorials 28–31, students will need to have Administrator-level access. In addition, some of the examples in the book require that students have Administrator access to the computer, which is typically required to develop applications in Visual Studio .NET.

A Note Regarding Terminology Used in the Book

In Tutorial 13, we discuss methods as Sub procedures (sometimes called subroutines) and Function procedures (sometimes called functions). We use this terminology for two reasons. First, the keywords Sub and Function are used in procedure definitions, so this naming is logical for students. Second, Visual Basic professionals have used this terminology for years and will continue to do so in the future. We also use the term “function” at certain points in this text to refer to Visual Basic 6 Function procedures that remain in Visual Basic .NET 2003 (such as Val and Pmt). When we introduce object-oriented programming concepts in Tutorial 19, we discuss the difference between procedures and methods and indicate that the procedures defined throughout the text are, in fact, methods. We hope our use of terminology helps you present the material in a simple and understandable manner.

Exception Handling: Bonus Tutorial (Tutorial 32)

Exception handling is one of the most important topics in Visual Basic .NET for building mission-critical and business-critical applications. Programmers need to know how to recognize the exceptions (errors) that could occur in software components and handle those exceptions effectively, allowing programs to deal with problems and continue executing instead of “crashing.” Tutorial 32 overviews the proper use of exception handling, including the termination model of exception handling, throwing exceptions and catching exceptions.

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 Outputs)

We present Visual Basic .NET features in the context of complete, working Visual Basic .NET programs. 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/vbnetSIMPLY1_2003/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 an application-development tool, but we do use UML diagrams to explain the precise operation of many of Visual Basic .NET's control statements.]

Programming Tips

Hundreds of programming tips to 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 programs 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 Visual Basic .NET that prevent errors from getting into programs in the first place, which simplifies the testing and debugging process.



Performance Tips

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



Portability Tips

The *Portability Tips* provide insights on how Visual Basic .NET achieves its high degree of portability among .NET platforms.



Software Design Tips

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



GUI Design Tips

The *GUI Design Tips* highlight graphical-user-interface conventions to help students design attractive, user-friendly GUIs and use GUI features. Appendix C compiles all the GUI Design Tips to help students as they prepare for exams.

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.

236 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.

850 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 programming exercises and a programming challenge. [*Note:* In the "What does this code do?" and "What's wrong with this code?" exercises, we only show portions of the code in the text.]

The questions involve simple recall of important terminology and concepts, writing individual Visual Basic .NET statements, writing small portions of Visual Basic .NET applications and writing complete Visual Basic .NET methods, classes and applications. Every programming exercise uses a step-by-step methodology to suggest how to solve the problems. 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 presents a cumulative list of these GUI design guidelines for easy reference.

Controls, Events, Properties & Methods Summaries

Each tutorial includes a summary of the controls, events, properties and methods covered in the tutorial. The summary includes a picture of each control, shows the control "in action" and lists the control's properties, events and methods that were discussed up to and including that tutorial. In addition, Appendix E 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 also are included in the index. Every Visual Basic .NET 2003 source-code program in the book is indexed it under the appropriate application. We also double-indexed various features, such as controls and properties. This makes it easier to find examples using particular features.

Simply Visual Basic .NET 2003 Ancillary Package

Simply Visual Basic .NET 2003 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/vbnetSIMPLY1_2003/index.html and are available as part of Prentice Hall's *Companion Web Site* (www.prenhall.com/deitel) for *Simply Visual Basic .NET 2003*, which offers resources for both instructors and students.

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's 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 Visual Basic .NET 2003*, visit www.prenhall.com/deitel.

Course Management Systems

Selected content from *Simply Visual Basic .NET 2003* and other Deitel textbooks, 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**

To view free online demonstrations and learn more about Course Management Systems that support Deitel content, visit the following Web sites:

- Blackboard: www.blackboard.com and www.prenhall.com/blackboard.
- WebCT: www.webct.com and www.prenhall.com/webct.
- CourseCompass: www.coursecompass.com and www.prenhall.com/coursecompass.

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
- Christi Kelsey, Director of Business Development
- Jeff Listfield, Developer
- David Choffnes, Developer
- Barbara Deitel, Chief Financial Officer
- Christina Courtemarche, Developer
- Betsy DuWaldt, Editorial Director
- Jon Liperi, Developer
- Tem Nieto, Director of Product Development
- Laura Treibick, Director of Multimedia

We would also like to thank the participants in the Deitel & Associates, Inc., College Internship Program: Jon Henry, Emanuel Achildiev, Nicholas Cassie, Thiago Lucas da Silva, Mike Dos’Santos, Jimmy Nguyen and Ngale Truong.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—our mentor in publishing—Marcia Horton, Editorial Director of

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.

Prentice Hall's Engineering and Computer Science Team. 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 would like to thank the design team that created a completely new look and feel for the *Simply* series—Carole Anson, Paul Belfanti, Tom Manshreck, Jonathan Boylan, John Root and Gary Gray. We would like to thank Carol Trueheart, Editor-in-Chief of Development for Prentice Hall's Engineering, Science and Math Division, and her team of developmental editors—Laura Cheu, Deena Cloud and Susan Cohen.

We wish to acknowledge the efforts of our reviewers and to thank Carole Snyder and Jennifer Capello of Prentice Hall, who managed the review process. The 42 reviewers from colleges, industry and Microsoft helped us to get this book "right." Adhering to a tight time schedule, these reviewers scrutinized the text and the programs, 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 Visual Basic .NET reviewers:

Cameron McColl (Microsoft)
Colin Merry (Microsoft)
Jeffrey Welton (Microsoft)
Judith Ashworth (Orillion USA, Inc.)
James Ball (Indiana State University)
Robert Benavides (Collin County Community College)
Chadi Boudiab (Georgia Perimeter College)
Charles Cadenhead (Brookhaven College)
Kunal Cheda (DotNetExtreme.com)
Mave Coxon (Lansing Community College)
Chris Crane (Independent Consultant)
Sergio Davalos (University of Washington-Tacoma)
David Fullerton (Yeshiva University)
George Gintowt (William Rainey Harper College)
James Gips (Boston College)
Manu Gupta (Patni Computer Systems)
Richard Hewer (Ferris State University)
James Huddleston (Independent Consultant)
Terrell Hull (Sun Certified Java Architect, Rational Qualified Practitioner)
Jeff Jones (A.D.A.M. Inc.)
Faisal Kaleem (Florida International University)
Yashavant Kanetkar (KICIT Pvt. Ltd.)
Dhananjay Katre (Patni Computer Systems, Ltd.)
Kurt Kominek (Northeast State Technical CC)
Stan Kurkovsky (Columbus State University)
Brian Larson (Modesto Junior College)
Sukan Makmuri (DeVry-Fremont)
Ken McLean (Northern Virginia CC)
Gordon McNorton (Collin County Community College)
Manish Mehta (Independent Consultant)
Marilyn Meyer (Fresno City College)
John Mueller (DataCon Services)
Narayana Rao Surapaneni (Patni Computer Systems)
Michael Rudisill (Northern Michigan University)
Sara Rushinek (University of Miami)
Praveen Sadhu (Infodat Solutions, Inc.)
Kenneth Schoonover (Chubb Technical Institute)
Andrea Shelly (Florida International University)

Robert Taylor (Lansing Community College)
Yateen Thakkar (Syntel India Ltd.)
Catherine Wyman (DeVry-Phoenix)
David Zeng (DeVry-Calgary)

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

deitel@deitel.com

We will respond promptly.

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

Dr. Harvey M. Deitel
Paul J. Deitel
Cheryl H. Yaeger

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.

Cheryl H. Yaeger, Director of Microsoft Software Publications with Deitel & Associates, Inc., is a graduate of Boston University with a degree in Computer Science. Cheryl has co-authored various Deitel & Associates, Inc. publications, including *Simply C#*, *C# How to Program*, *C#: A Programmer's Introduction*, *C# for Experienced Programmers*, *Visual Basic .NET for Experienced Programmers* and *Simply Java™ Programming* and has contributed to several others.

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 28-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 through:

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
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 opt-in registration, visit www.deitel.com/newsletter/subscribe.html.