

Preface

Welcome to the Visual Basic® 2012 computer programming language and the world of Microsoft® Windows® and Internet and web programming with Microsoft's .NET platform. Please read the book's back cover and inside back cover—these concisely capture the book's essence. In this Preface we provide more details.

This book is appropriate for information technology and business students in novice-level and intermediate-level Visual Basic courses. The book is also used by professional programmers.

At the heart of the book is the Deitel signature *live-code approach*—rather than using code snippets, we present concepts in the context of complete working programs followed by sample executions. Read the Before You Begin section after this Preface for instructions on setting up your computer to run the code examples. The source code is available at www.deitel.com/books/vb2012http and www.pearsonhighered.com/deitel. Use the source code we provide to *compile and run each program* as you study it—this will help you master Visual Basic and related Microsoft technologies faster and at a deeper level.

We believe that this book and its supplements for students and instructors will give you an informative, engaging, challenging and entertaining introduction to Visual Basic. If you have questions, we're easy to reach at deitel@deitel.com—we'll respond promptly. For book updates, visit www.deitel.com/books/vb2012http, join our social media communities on Facebook (www.deitel.com/DeitelFan), Twitter (@deitel), Google+ ([gplus.to/deitel](https://plus.google.com/+deitel)) and LinkedIn (bit.ly/DeitelLinkedIn), and subscribe to the *Deitel® Buzz Online* newsletter (www.deitel.com/newsletter/subscribe.html).

Visual Basic® 2012, the Visual Studio® 2012 IDE, .NET 4.5, Windows® 7 and Windows® 8

The new Visual Basic 2012 and its associated technologies motivated us to write *Visual Basic 2012 How to Program*. These are some of the key features of this new edition:

- **Use with Windows 7, Windows 8 or both.** The book is designed so that you can continue to use Windows 7 now and begin to evolve to Windows 8, if you like, or you can move right to Windows 8. All of the code examples in Chapters 1–19 and 23–31 were tested on *both* Windows 7 and Windows 8. The code examples for the Windows-8-specific chapters—Chapter 20 (Windows 8 UI and XAML), Chapter 21 (Windows 8 Graphics and Multimedia) and Chapter 22 (Building a Windows Phone 8 App)—were tested *only* on Windows 8.
- **Modular multi-GUI treatment with Windows Forms, Windows 8 UI and WPF.** The printed book features Windows Forms GUI; optional online chapters contain treatments of WPF GUI and the new Windows 8 UI. Windows 8 UI apps are

called *Windows Store apps*. In Chapter 20, you'll learn how to create and test Windows Store apps and upload them to Microsoft's Windows Store.

- **Modular treatment of graphics and multimedia with Windows 8 and WPF.** The book features optional online chapters on both Windows 8 Graphics and Multimedia (Chapter 21) and WPF Graphics and Multimedia (Chapter 28).
- **Database with LINQ to Entities.** In the previous edition of this book, we discussed LINQ (Language Integrated Query) to SQL (Microsoft's SQL Server database system). Microsoft stopped further development on LINQ to SQL in 2008 in favor of the newer and more robust LINQ to Entities and the ADO.NET Entity Framework, which we've switched to in this edition, keeping the discussion friendly for novices.
- **SQL Server database.** We use Microsoft's free SQL Server Express 2012 (which installs with the free Visual Studio Express 2012 for Windows Desktop) to present the fundamentals of database programming. Chapters 12–13 and online Chapters 24–25 use database and LINQ capabilities to build an address-book desktop app, a web-based guestbook app, a bookstore app and an airline reservation system app.
- **ASP.NET 4.5.** Microsoft's .NET server-side technology, ASP.NET, enables you to create robust, scalable web-based apps. In Chapter 13, you'll build several apps, including a web-based guestbook that uses ASP.NET and the ADO.NET Entity Framework to store data in a database and display data in a web page. The chapter also discusses the IIS Express web server for testing your web apps on your local computer.
- **Building a Windows Phone 8 App.** Windows Phone 8 is Microsoft's latest operating system for smartphones. It features multi-touch support for touchpads and touchscreen devices, enhanced security features and more. In online Chapter 22, you'll build a complete working Windows Phone 8 app and test it on the Windows Phone simulator; we'll discuss how to upload apps to the Windows Phone Store.
- **Building a Windows Azure™ Cloud Computing App.** Windows Azure is a cloud computing platform that allows you to develop, manage and distribute your apps in the cloud. Online Chapter 26 shows you how to build a Windows Azure app that can store data in the cloud.
- **Asynchronous programming with *async* and *await*.** Asynchronous programming is simplified in Visual Basic 2012 with the new *Async* and *Await* capabilities. We introduce asynchronous programming with *Async* and *Await* in online Chapter 23.

Object-Oriented Programming

- **Late objects approach.** We defer the discussion of creating custom classes until Chapter 9, but in the early chapters, we still use lots of existing objects. Chapter 10 discusses how to create powerful new classes quickly by using inheritance to “absorb” the capabilities of existing classes, and presents the crucial concepts of polymorphism, abstract classes and interfaces.
- **Rich coverage of programming fundamentals.** Chapters 4 and 5 present a friendly treatment of control statements and problem solving.

- *A clear, example-driven presentation of classes, objects, inheritance, polymorphism and interfaces.*
- *Optional case study: Using the UML to develop an object-oriented design and Visual Basic implementation of an Automated Teller Machine (ATM).* The UML™ (Unified Modeling Language™) is the industry-standard graphical language for modeling object-oriented systems. We introduce the UML in the early chapters. Optional online Chapters 30 and 31 include an *optional* case study on object-oriented design using the UML. We design and implement the software for a simple automated teller machine. We analyze a typical *requirements document* that specifies the system to be built. We determine the *classes* needed to implement that system, the *attributes* the classes need to have, the *behaviors* the classes need to exhibit and we specify how the classes must *interact* with one another to meet the system requirements. From the design we produce a complete working Visual Basic implementation. Students often report a “light bulb moment”—the case study helps them “tie it all together” and truly understand object orientation.
- *Three programming paradigms.* We discuss *structured programming*, *object-oriented programming* and *generic programming*.

Interesting, Entertaining and Challenging Exercises

- Extensive self-review exercises *and* answers are included for self-study.
- Many chapters include a multiple-choice Quick Quiz.
- Each chapter concludes with a substantial set of exercises, which generally includes simple recall of important terminology and concepts, identifying the errors in code samples, writing individual program statements, writing small portions of Visual Basic classes, writing complete programs and implementing major projects. Figure 1 lists a small sampling of the book’s hundreds of exercises, including selections from our *Making a Difference* exercises set, which encourage you to use computers and the Internet to research and solve significant social problems—we hope you’ll approach these exercises with *your own* values, politics and beliefs.

Exercises		
Abstract Methods	Cafeteria Survey App	DateInformation Class
Account Information App	Calculator GUI	Diameter, Circumference and Area
Account Inheritance Hierarchy	Carbon Footprint Calculator	Digit Extraction
Airline Reservations system	Car-Pool Savings Calculator	Displaying Tabular Data
Alarm Clock GUI	Coin Tossing	Concatenating Strings
Arithmetic Calculator App	Computer-Assisted Instruction: Reducing Student Fatigue	Duplicate Elimination
Array Sorting App	Computerization of Health Records	Duplicate Word Removal
Average Calculator App	Concentric Circles	Employee Class
Baseball Database App	Credit Checker App	Enforcing Privacy with Cryptography
Blackjack Modification		
Body Mass Index Calculator		

Fig. 1 | A sampling of the book’s exercises. (Part 1 of 2.)

xx Preface**Exercises**

Enhanced Drawing App	Polling	Table of Decimal, Octal, Hexadecimal and Binary Equivalents
Enhanced Exam Analysis App	Polymorphism and Extensibility	Table of Powers App
Evaluating Poker Hands	Present Value Calculator App	Target-Heart-Rate Calculator
Find the Smallest and Largest Values	Protected vs. Private Access	Tax Plan Alternatives; The “Fair Tax”
Gas Pump	Pyramid	Telephone-Number Word Generator
Guess the Number App	Querying an Array of Invoice Objects	Temperature Converter App
Image Flasher	Quiz Average App	Triangles of Asterisks
Image Reflector App	Radio GUI	Using the Debugger: Discount Calculator App
Inheritance Advantages	Random Lines	Using the Debugger: Factorial App
Invoice Class	Reading Grades into a Two-Dimensional Array	Using the Debugger: Savings Calculator App
Large-Type Displays for People with Low Vision	Retail Sales Calculator App	Using the Debugger: Sibling Survey App
Lottery Number Generator	Road Sign Test App	Vending Machine App
MDI Text Editor	Sales Commissions	Wage Calculator with Tax Calculations
Miles Per Gallon App	Savings Account Class	Web-Based Address Book
Modifying the Internal Data Representation of a Class	Screen Saver App	World Population Growth
Multiples	Shape Hierarchy	Writing a Grade Report to a File
Notepad GUI	Simple Calculator App	
Nutrition Information XML Document	Simple Drawing App	
Package Inheritance Hierarchy	Snake PolyLine App	
Page Hit Counter	Sorting and Ascending and Descending Order	
Parking Charges	Speech-Controlled Drawing App	
Phone-Book Web Service	Square Class	
Pig Latin	Student Inheritance Hierarchy	

Fig. 1 | A sampling of the book's exercises. (Part 2 of 2.)**Other Features**

- *Illustrations and figures.* Abundant tables, line drawings, UML diagrams, programs and program outputs are included.
- *Focus on business and personal utility examples.*
- *Windows Forms GUI is integrated throughout the core chapters.* The core content focuses on Windows Forms GUI apps.
- *We use LINQ to query files, databases, XML and collections.* The introductory LINQ to Objects chapter (Chapter 11), is intentionally simple and brief to encourage instructors to begin covering LINQ technology early. We take a deeper look, using LINQ to Entities (Chapters 12–13 and online Chapters 24–25) and LINQ to XML (online Chapters 19, 25 and 26).
- *Integrated Using the Debugger sections and exercises* in the core printed book. Students use the debugger to locate and fix logic errors.
- *Strings, files and databases are covered early.*

- *Introduction to Web app development with ASP.NET is in the core print book.*
- *Local type inference.* When you initialize a local variable in its declaration, you can omit the variable's type—the compiler *infers* it from the initializer value.
- *Object initializers.* For new objects, you can use object initializer syntax (similar to array initializer syntax) to assign values to the new object's `public` properties and `public` instance variables.
- *We emphasize the IDE's IntelliSense feature* that helps you write code faster and with fewer errors.
- *Optional parameters.* You can specify method parameters with default values—if a corresponding method argument is not provided in the method call, the compiler inserts the optional parameter's default value in the call.
- *“Quick Fix” window.* We show how to use the IDE's **Error Correction Options** window to quickly fix certain common programming errors simply by clicking the suggested fix, which is displayed in a window in the code editor.
- *We show how to use DataTips and visualizers* to view object contents in the code window during debugging.
- *Integrated exception handling.* We introduce exception handling early (Chapter 7, Arrays) to ensure that we do not access an array element outside the array's bounds. Chapter 9, Object-Oriented Programming: Classes and Objects, shows how to indicate an exception when a member function receives an invalid argument. We cover the complete details of exception handling in online Chapter 16, Exception Handling: A Deeper Look.
- *Visual Basic XML capabilities.* Extensible Markup Language (XML) is pervasive in the software-development industry, e-business and throughout the .NET platform. In optional online Chapter 19, we introduce XML syntax and programmatically manipulate the elements of an XML document using LINQ to XML. XAML is an XML vocabulary that's used to describe graphical user interfaces, graphics and multimedia. We discuss XAML in optional online Chapters 20–21 and 27–28.
- *Web app development with ASP.NET 4.5 and ASP.NET AJAX.* Optional online Chapter 24 extends Chapter 13's ASP.NET discussion with a case study on building a password-protected, web-based bookstore app. We also introduce in Chapter 24 ASP.NET AJAX controls and use them to add AJAX functionality to web apps to give them a look and feel similar to that of desktop apps.

Companion Website

The printed book contains the core content (Chapters 1–15) for introductory course sequences. Several optional online chapters are available for advanced courses and professionals. Figure 2 lists the chapters that are available in searchable PDF format on the book's password-protected Companion Website at:

www.pearsonhighered.com/deitel

See the inside front cover of the book for an access code.

Online chapters

Chapter 16, Exception Handling: A Deeper Look	Chapter 26, Building a Windows Azure Cloud Computing App
Chapter 17, Strings and Characters: A Deeper Look	Chapter 27, Windows Presentation Foundation (WPF) GUI
Chapter 18, Files and Streams: A Deeper Look	Chapter 28, WPF Graphics and Multimedia
Chapter 19, XML and LINQ to XML	Chapter 29, Data Structures and Generic Collections
Chapter 20, Windows 8 UI	Chapter 30, ATM Case Study, Part 1: Object-Oriented Design with the UML
Chapter 21, Windows 8 Graphics and Multimedia	Chapter 31, ATM Case Study, Part 2: Implementing an Object-Oriented Design
Chapter 22, Windows Phone 8 Case Study	Index (The online index includes the content from the printed book and the online content. The printed book index covers only the printed material.)
Chapter 23, Introduction to Concurrency: Async and Await	
Chapter 24, Web App Development with ASP.NET: A Deeper Look	
Chapter 25, Web Services	

Fig. 2 | Optional online chapters in *Visual Basic 2012 How to Program*.

VideoNotes

The Companion Website also includes extensive *VideoNotes*—watch and listen as co-author Paul Deitel discusses key code examples in the core chapters of the book. VideoNotes allow for self-paced instruction with easy navigation, including the ability to select, play, rewind, fast-forward and stop within each video.

We've created a jump table that maps each VideoNote to the corresponding figures in the book (www.deitel.com/books/vb2012http/jump_table.pdf). VideoNotes are free with the purchase of a *new* textbook. If you have a *used* book you can purchase access to the VideoNotes for this book as follows:

1. Go to www.pearsonhighered.com/deitel/.
2. Scroll to *Visual Basic 2012 How to Program* and click **Companion Website**.
3. Click the **Register** button.
4. On the registration page, enter your student access code found beneath the scratch-off panel on the inside front cover of this book. Do not type the dashes. You can use lower- or uppercase. The access code can be used *only once*. This subscription is valid for twelve months upon activation and is *not transferable*. If this access code on your book has already been revealed, it may no longer be valid. If this is the case, click the **Website Purchase** link and follow the instructions.
5. Once your personal Login Name and Password are confirmed, you can begin using the *Visual Basic 2012 How to Program* Companion Website.

Book Overview and Chapter Dependencies

This section discusses the book's modular organization to help instructors plan their syllabi.

Introduction to Visual Basic and Visual Studio 2012 Express

Chapter 1, Introduction to Computers, the Internet and Visual Basic, introduces computing fundamentals and Microsoft's .NET platform. If you do not need to cover these fundamentals, you should still cover the **Painter** app test-drive. The vast majority of the book's examples will run on Windows 7 and Windows 8 using *Visual Studio Express 2012 for Windows Desktop*, which we test-drive in Section 1.14. Online Chapters 20–21 can be run *only* on Windows 8 using *Visual Studio Express 2012 for Windows 8*. There are other versions of *Visual Studio Express 2012* for web development and Windows Phone development—we cover these in the corresponding chapters.

Chapter 2, Dive Into[®] Visual Studio Express 2012 for Windows Desktop, shows how to develop a simple GUI app that displays text and an image. We'll look at Visual Studio Express 2012 for Windows 8 in more depth in online Chapter 20.

Introduction to Visual Basic Fundamentals

The chapters in this module of the book:

- Chapter 3, Introduction to Visual Basic Programming
- Chapter 4, Introduction to Problem Solving and Control Statements
- Chapter 5, Problem Solving and Control Statements: Part 2
- Chapter 6, Methods
- Chapter 7, Arrays
- Chapter 8, Files

present Visual Basic programming fundamentals (data types, operators, control statements, methods, arrays and files). These chapters should be covered in order. Chapter 7 introduces exception handling with an example that demonstrates accessing an element outside an array's bounds.

Object-Oriented Programming

The chapters in this module of the book:

- Chapter 9, Object-Oriented Programming: Classes and Objects
- Chapter 10, Object-Oriented Programming: Inheritance and Polymorphism
- Chapter 11, Introduction to LINQ
- Chapter 16, Exception Handling: A Deeper Look
- Chapter 30, ATM Case Study, Part 1: Object-Oriented Design with the UML
- Chapter 31, ATM Case Study, Part 2: Implementing an Object-Oriented Design

discuss object-oriented programming, including classes, objects, inheritance, polymorphism, interfaces and exception handling. Chapter 11, Introduction to LINQ, introduces Microsoft's Language Integrated Query (LINQ) technology, which provides a uniform syntax for manipulating data from various data sources, such as arrays and, as you'll see in later chapters, collections, XML and databases. This chapter can be deferred, but it's required for many of the later chapters starting with Chapter 12, Databases and LINQ. Online Chapters 30–31 present an *optional* object-oriented design and implementation case study

xxiv Preface

that requires the Visual Basic and object-oriented programming concepts presented in Chapters 3–7 and 9–10.

Windows Forms Graphical User Interfaces (GUIs), Graphics and Multimedia

There are now three GUI technologies in Windows—Windows Forms (which is a legacy technology), Windows 8 UI (available *only* on Windows 8) and Windows Presentation Foundation (WPF). We surveyed instructors teaching Visual Basic and they still prefer Windows Forms for their classes, so Windows Forms GUI is integrated throughout most of the book. Chapter 14, Windows Forms GUI: A Deeper Look, covers additional Windows Forms GUI controls and Chapter 15, Graphics and Multimedia, introduces graphics and multimedia. For those who wish to present or study Microsoft’s more recent GUI, graphics and multimedia technologies, we provide online introductions to Windows 8 UI, graphics and multimedia (online Chapters 20–21) and WPF GUI, graphics and multimedia (online Chapters 27–28).

Strings and Files

We introduce `Strings` beginning in Chapter 3 and use them throughout the book. We introduce files beginning in Chapter 8. Online Chapter 17, Strings and Characters: A Deeper Look, investigates `Strings` in more depth, and online Chapter 18, Files and Streams: A Deeper Look, discusses files in more depth.

Databases and an Introduction to Web App Development

Chapter 12, Databases and LINQ, introduces database app development using the ADO.NET Entity Framework and LINQ to Entities. The chapter’s examples require Visual Basic, object-oriented programming and Windows Forms concepts presented in Chapters 3–11. The final example in Chapter 13, Web App Development with ASP.NET requires the LINQ and database techniques presented in Chapter 12.

Extensible Markup Language (XML)

Online Chapter 19, XML and LINQ to XML, introduces XML, which is used in several later chapters. The first few sections of this chapter are required to understand the XAML markup that’s used to build Windows 8 GUI, graphics and multimedia apps (Chapters 20–21), Windows Phone 8 apps (Chapter 22) and WPF GUI, graphics and multimedia apps (Chapters 27–28). The remainder of the chapter discusses LINQ to XML, which allows you to manipulate XML using LINQ syntax. These capabilities are used in Chapters 25 and 26.

Windows 8 UI, Graphics and Multimedia; Windows Phone

The online chapters in this module of the book:

- Chapter 20, Windows 8 UI
- Chapter 21, Windows 8 Graphics and Multimedia
- Chapter 22, Windows Phone 8 Case Study

present Windows 8 UI, graphics and multimedia, and Windows Phone 8 app development. These chapters can be used *only* on computers running Windows 8—they depend on event-handling concepts that are presented throughout the early chapters and the introduction to XML at the beginning of online Chapter 19 (see Section 19.1 for details). Developing a Windows Phone 8 app is similar to developing a Windows 8 UI app.

Asynchronous Programming

Online Chapter 23, Introduction to Concurrency: Async and Await, demonstrates .NET's and Visual Basic's new simplified asynchronous programming capabilities. These are commonly used in Web app and Web service development among many other uses.

Web App Development and Web Services

The chapters in this module of the book:

- Chapter 24, Web App Development with ASP.NET: A Deeper Look
- Chapter 25, Web Services
- Chapter 26, Building a Windows Azure™ Cloud Computing App

continue our discussion of Web app development from Chapter 13 and introduce web services, including a case study on cloud computing with Windows Azure. Online Chapters 25 and 26 depend on the LINQ to XML discussion in Chapter 19.

Windows Presentation Foundation (WPF) GUI, Graphics and Multimedia

The chapters in this module of the book

- Chapter 27, Windows Presentation Foundation (WPF) GUI
- Chapter 28, WPF Graphics and Multimedia

discuss Windows Presentation Foundation GUI, graphics and multimedia. These chapters can be used on computers running Windows 7 or Windows 8 and depend on event-handling concepts that are presented throughout the early chapters and the introduction to XML at the beginning of online Chapter 19.

Teaching Approach

Visual Basic 2012 How to Program contains a rich collection of examples. We concentrate on building good software and stress program clarity.

Live-Code Approach. The book is loaded with “live-code” examples. Most new concepts are presented in the context of complete working Visual Basic apps, followed by one or more executions showing program inputs and outputs. In the few cases where we show a code snippet, to ensure correctness we first tested it in a complete working program then copied the code from the program and pasted it into the book.

Syntax Shading. For readability, we syntax shade the code, similar to the way most integrated-development environments and code editors syntax color code. Our syntax-shading conventions are:

```
comments appear like this
keywords appear like this
constants and literal values appear like this
all other code appears in black
```

Code Highlighting. We place light blue rectangles around each program's key code.

Using Fonts for Emphasis. We place the key terms and the index's page reference for each defining occurrence in **bold blue** text for easy reference. We emphasize on-screen compo-

xxvi Preface

nents in the **bold Helvetica** font (for example, the **File** menu) and Visual Basic program text in the Lucida font (for example, Dim count As Integer = 5).

Objectives. The opening quotes are followed by a list of chapter objectives.

Illustrations/Figures. Abundant tables, line drawings, UML diagrams, programs and program outputs are included.

Programming Tips. We include programming tips to help you focus on important aspects of program development. These tips and practices represent the best we've gleaned from a combined seven decades of programming and teaching experience.

**Good Programming Practice**

The Good Programming Practices call attention to techniques that will help you produce programs that are clearer, more understandable and more maintainable.

**Common Programming Error**

Pointing out these Common Programming Errors reduces the likelihood that you'll make them.

**Error-Prevention Tip**

These tips contain suggestions for exposing and removing bugs from your programs; many describe aspects of Visual Basic that prevent bugs from getting into programs.

**Performance Tip**

These tips highlight opportunities for making your programs run faster or minimizing the amount of memory that they occupy.

**Portability Tip**

The Portability Tips help you write code that will run on a variety of platforms.

**Software Engineering Observation**

The Software Engineering Observations highlight architectural and design issues that affect the construction of software systems, especially large-scale systems.

**Look-and-Feel Observation**

These observations help you design attractive, user-friendly graphical user interfaces that conform to industry norms.

Summary Bullets. We present a section-by-section, bullet-list summary of each chapter.

Terminology. We include an alphabetized list of the important terms defined in each chapter with the page number of each term's defining occurrence for easy reference.

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

Exercises. Each chapter concludes with additional exercises including:

- simple recall of important terminology and concepts
- What's wrong with this code?

- What does this code do?
- Using the Debugger
- writing individual statements and small portions of methods and classes
- writing complete methods, classes and programs
- major projects.

Check out our Programming Projects Resource Center for lots of additional exercise and project possibilities (www.deitel.com/ProgrammingProjects/).

Index. We've included an extensive index for reference. Defining occurrences of key terms in the index are highlighted with a **bold blue** page number.

Software Included with Visual Basic 2012 How to Program

This book includes the Microsoft® Visual Studio® Express 2012 for Windows Desktop DVD, which runs on Windows 7 and 8. See the Before You Begin section that follows this preface for information on downloading the other Visual Studio Express 2012 Editions that we use in this book.

Instructor Supplements

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

- *Solutions Manual* contains solutions to *most* of the end-of-chapter exercises. **Please do not write to us requesting access to the Pearson Instructor's Resource Center. Access is restricted to 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/. Exercise Solutions are *not* provided for "project" exercises. Check out our Programming Projects Resource Center for lots of additional exercise and project possibilities:

www.deitel.com/ProgrammingProjects

- *Test Item File* of multiple-choice questions (approximately two per book section)
- *Customizable PowerPoint® slides* containing all the code and figures in the text, plus bulleted items that summarize the key points in the text.

Microsoft DreamSpark™

Professional Developer and Designer Tools for Students

Microsoft provides many of its professional developer tools to students for free via a program called DreamSpark (www.dreamspark.com). See the website for details on verifying your student status so you take advantage of this program.

Acknowledgments

We'd like to thank Barbara Deitel of Deitel & Associates, Inc. for long hours devoted to this project. She painstakingly researched the new capabilities of Visual Basic 2012, .NET 4.5, Windows 8, Windows Phone 8, Windows Azure and other key topics.

xxviii Preface

We're fortunate to have worked with the dedicated team of publishing professionals at Pearson Higher Education. We appreciate the guidance, wisdom and energy of Tracy Johnson, Executive Editor, Computer Science. Carole Snyder did an extraordinary job recruiting the book's reviewers and managing the review process. Bob Engelhardt did a wonderful job bringing the book to publication.

Reviewers

We wish to acknowledge the efforts of our reviewers. The book was scrutinized by academics teaching Visual Basic courses and industry experts. They provided countless suggestions for improving the presentation. Any remaining flaws in the book are our own.

Sixth edition reviewers: Wu He (Old Dominion University), Ken Tucker (Microsoft MVP and Software Developer, Sea World), José Antonio González Seco (Parliament of Andalusia) and Jim Wooley (Slalom Consulting, Microsoft Visual Basic MVP, Author of LINQ in Action).

Other recent edition reviewers: Douglas B. Bock (MCSD.NET, Southern Illinois University Edwardsville), Dan Crevier (Microsoft), Amit K. Ghosh (University of Texas at El Paso), Marcelo Guerra Hahn (Microsoft), Kim Hamilton (Software Design Engineer at Microsoft and co-author of *Learning UML 2.0*), Huanhui Hu (Microsoft Corporation), Vitek Karas (Microsoft), Narges Kasiri (Oklahoma State University), James Edward Keyser (Florida Institute of Technology), Helena Kotas (Microsoft), Charles Liu (University of Texas at San Antonio), Chris Lovett (Software Architect at Microsoft), Bashar Lulu (INETA Country Leader, Arabian Gulf), John McIlhinney (Spatial Intelligence; Microsoft MVP 2008 Visual Developer, Visual Basic), Ged Mead (Microsoft Visual Basic MVP, DevCity.net), Anand Mukundan (Architect, Polaris Software Lab Ltd.), Dr. Hamid R. Nemati (The University of North Carolina at Greensboro), Timothy Ng (Microsoft), Akira Onishi (Microsoft), Jeffrey P. Scott (Blackhawk Technical College), Joe Stagner (Senior Program Manager, Developer Tools & Platforms), Erick Thompson (Microsoft) and Jesús Ubaldo Quevedo-Torrero (University of Wisconsin–Parkside, Department of Computer Science)

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

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, 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.

Paul was named as a Microsoft® Most Valuable Professional (MVP) for C# in 2012. According to Microsoft, “the Microsoft MVP Award is an annual award that recognizes exceptional technology community leaders worldwide who actively share their high quality, real world expertise with users and Microsoft.”



2012 C# MVP

Dr. Harvey Deitel, Chairman and Chief Strategy Officer of Deitel & Associates, Inc., has over 50 years of experience in computing. Dr. Deitel earned B.S. and M.S. degrees in Electrical Engineering from MIT and a Ph.D. in Mathematics from Boston University. 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 programming courses to corporate, academic, government and military clients.

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 Visual Basic®, Visual C#®, Visual C++®, C++, C, Java™, XML®, Python®, object technology, Internet and web programming, Android app development, Objective-C and iPhone app development 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, 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 organization, e-mail deitel@deitel.com.

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

www.informit.com/store/sales.aspx

