Ivor Horton's Beginning Visual C++® 2008

Ivor Horton

Wiley Publishing, Inc.

Ivor Horton's Beginning Visual C++® 2008

Ivor Horton's Beginning Visual C++® 2008

Ivor Horton

Wiley Publishing, Inc.

Ivor Horton's Beginning Visual C++®2008

Published by
Wiley Publishing, Inc.
10475 Crosspoint Boulevard
Indianapolis, IN 46256
www.wiley.com

Published by Wiley Publishing, Inc., Indianapolis, Indiana

Published simultaneously in Canada

ISBN: 978-0-470-22590-5

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

Library of Congress Cataloging-in-Publication Data is available from the publisher.

This book is dedicated to Isabella Williams, who joined the growing band of beautiful girls in my life in January 2007.

About the Author

Ivor Horton graduated as a mathematician and was lured into information technology by promises of great rewards for very little work. In spite of the reality being usually a great deal of work for relatively modest rewards, he has continued to work with computers to the present day. He has been engaged at various times in programming, systems design, consultancy, and the management of the implementation of projects of considerable complexity.

Horton has many years of experience in the design and implementation of computer systems applied to engineering design and to manufacturing operations in a variety of industries. He has considerable experience developing occasionally useful applications in a wide variety of programming languages, and teaching primarily scientists and engineers to do likewise. He has been writing books on programming for more than 10 years now, and his currently published works include tutorials on C, C++, and Java. At the present time, when he is not writing programming books or providing advice to others, he spends his time fishing, traveling, and trying to speak better French.

Acknowledgments

I'd like to acknowledge the efforts and support of the John Wiley & Sons and Wrox Press editorial and production team in the production of this book, especially my development editor, Ed Connor, who has been there from way back at the beginning and has stayed through to the end. I'd also like to thank my technical editor, John Mueller, once again for doing such an outstanding job of reviewing the text and checking out all the examples in the book; his many constructive comments undoubtedly helped make the book a much better tutorial.

As always, the love and support of my wife, Eve, have been fundamental to making it possible for me to write this book. She has pandered to my every need throughout and has remained patient and cheerful in spite of the hours I spend locked away in my office and my intermittent grumblings about the workload I imposed upon myself.

Credits

Executive Editor
Robert Elliott

Development Editor
Ed Connor

Technical Editor
John Mueller

Copy Editor
Mildred Sanchez

Editorial Manager
Mary Beth Wakefield

Production Manager
Tim Tate

Vice President and Executive Group Publisher
Richard Swadley

Vice President and Executive Publisher
Joseph B. Wikert

Project Coordinator, Cover
Lynsey Stanford

Compositor
Laurie Stewart, Happenstance Type-O-Rama

Proofreaders
Jeremy Bagai
Sheilah Ledwidge
Carrie Hunter
Lee Ewert
Will DeRooy

Indexer
Jack Lewis

Contents

Introduction

Welcome to *Beginning Visual C++® 2008*. With this book you can become an effective C++ programmer. The latest development system from Microsoft, Visual Studio 2008, supports two distinct but closely related flavors of the C++ language; it fully supports the original ISO/ANSI standard C++, and you also get support for a new version of C++ called C++/CLI that was developed by Microsoft and is now an ECMA standard. These two versions of C++ are complementary and fulfill different roles. ISO/ANSI C++ is there for the development of high-performance applications that run natively on your computer whereas C++/CLI has been developed specifically for writing applications that target the .NET Framework. This book will teach you how to write applications in both versions of C++.

You get quite a lot of assistance from automatically generated code when writing ISO/ANSI C++ programs, but you still need to write a lot of C++ yourself. You need a solid understanding of object-oriented programming techniques, as well as a good appreciation of what's involved in programming for Windows. Although C++/CLI targets the .NET Framework, it also is the vehicle for the development of Windows Forms applications that you can develop with little or in some cases no explicit code writing. Of course, when you do have to add code to a Windows Forms application, even though it may be a very small proportion of the total, you still need an in-depth knowledge of the C++/CLI language. ISO/ANSI C++ remains the language of choice for many professionals, but the speed of development that C++/CLI and Windows Forms applications bring to the table make that essential, too. For this reason I cover both flavors of C++ in this book.

Whom This Book Is For

This book is aimed at teaching you how to write C++ applications for the Microsoft Windows operating system using Visual C++ 2008 or any edition of Visual Studio 2008. I make no assumptions about prior knowledge of any particular programming language. This tutorial is for you if:

- You have a little experience programming in some other language, such as BASIC for example, and you are keen to learn C++ and develop practical Microsoft Windows programming skills.
- You have some experience in C or C++, but not in a Microsoft Windows context and want to extend your skills to program for the Windows environment using the latest tools and technologies.
- You have some knowledge of C++ and you want to extend your C++ skills to include C++/CLI.
- You are a newcomer to programming and sufficiently keen to jump in the deep end with C++. To be successful you need to have at least a rough idea of how your computer works, including the way in which the memory is organized and how data and instructions are stored.

What This Book Covers

My objective with this book is to teach you the essentials of C++ programming using both of the technologies supported by Visual C++ 2008. The book provides a detailed tutorial on both flavors of the C++ language, on native ISO/ANSI C++ Windows application development using the Microsoft Foundation Classes (MFC), and on the development of C++/CLI Windows applications using Windows Forms.

Because of the importance and pervasiveness of database technology today, the book also includes introductions to the techniques you can use for accessing data sources in both MFC and Windows Forms applications. MFC applications are relatively coding-intensive compared to Windows Forms applications. This is because you create the latter using a highly developed design capability in Visual C++ 2008 that enables you to assemble the entire graphical user interface (GUI) for an application graphically and have all the code that creates it generated automatically. For this reason, there are more pages in the book devoted to MFC programming than to Windows Forms programming.

How This Book Is Structured

The contents of this book are structured as follows:

- Chapter 1 introduces you to the basic concepts you need to understand for programming in C++ for native applications and for .NET Framework applications, together with the main ideas embodied in the Visual C++ 2008 development environment. It describes how you use the capabilities of Visual C++ 2008 for creating the various kinds of C++ applications you learn about in the rest of the book.
- Chapters 2 to 9 are dedicated to teaching you both versions of the C++ language. The content of each of the Chapters 2 through 9 is structured in a similar way; the first half of each chapter deals with ISO/ANSI C++ topics, and the second half deals with C++/CLI.
- Chapter 10 teaches you how you use the Standard Template Library (STL), which is a powerful and extensive set of tools for organizing and manipulating data in your native C++ programs. The STL is application-neutral so you will be able to apply it in a wide range of contexts. Chapter 10 also teaches you the STL/CLR, which is new in Visual C++ 2008. This is a version of the STL for C++/CLI applications.
- Chapter 11 introduces you to techniques for finding errors in your C++ programs.
- Chapter 12 discusses how Microsoft Windows applications are structured and describes and demonstrates the essential elements that are present in every Windows application. The chapter explains elementary examples of Windows applications using ISO/ANSI C++ and the Windows API and the MFC, as well as an example of a basic Windows Forms application in C++/CLI.
- Chapters 13 to 18 describe in detail the capabilities provided by the MFC for building a GUI and how you use the equivalent facilities in a program for the .NET Framework. You learn how you create and use common controls to build the graphical user interface for your application and how you handle the events that result from user interactions with your program. In the process, you create a substantial working application in native C++, and a program with essentially the same functionality in C++/CLI. In addition to the techniques you learn for building a GUI, the applications that you develop also show you how you print documents and how you save them on disk.

- Chapter 19 teaches you the essentials you need to know for creating your own libraries using MFC. You learn about the different kinds of libraries you can create, and you develop working examples of these that work with the application that you have evolved over the preceding six chapters.
- In Chapters 20 and 21, you learn about accessing data sources in an MFC application. You gain experience in accessing a database in read-only mode; then you learn the fundamental programming techniques for updating a database using MFC. The examples use the Northwind database that can be downloaded from the Web, but you can also apply the techniques described to your own data source.
- In Chapter 22 you work with Windows Forms and C++/CLI to build an example that teaches you how to create, customize, and use more Windows Forms controls in an application. You gain practical experience by building a second C++/CLI application incrementally throughout the chapter.
- Chapter 23 builds on the knowledge you gain in Chapter 22 and shows how the controls available for accessing data sources work, and how you customize them. You also learn how you can create an application for accessing a database with virtually no coding at all on your part.

All chapters include numerous working examples that demonstrate the programming techniques that are discussed. Every chapter concludes with a summary of the key points that were covered, and most chapters include a set of exercises at the end that you can attempt to apply what you have learned. Solutions to the exercises, together with all the code from the book, are available for download from the publisher's Web site (see the "Source Code" section later in this Introduction for more details).

The tutorial on the C++ language uses examples that are console programs with simple command-line input and output. This approach enables you to learn the various capabilities of C++ without getting bogged down in the complexities of Windows GUI programming. Programming for Windows is really only practicable after you have a thorough understanding of the programming language.

If you want to keep things as simple as possible, you can just learn ISO/ANSI C++ programming in the first instance. Each of the chapters that cover the C++ language (Chapters 2 to 9) first discusses particular aspects of the capabilities of ISO/ANSI C++, followed by the new features introduced by C++/CLI in the same context. The reason for organizing things this way is that C++/CLI is defined as an extension to the ISO/ANSI standard language, so an understanding of C++/CLI is predicated on knowledge of ISO/ANSI C++. Thus, you can just read the ISO/ANSI topics in each of Chapters 2 to 21 and ignore the C++/CLI sections that follow. You then can progress to Windows application development with ISO/ANSI C++ without having to keep the two versions of the language in mind. You can return to C++/CLI when you are comfortable with ISO/ANSI C++. Of course, you can also work straight through and add to your knowledge of both versions of the C++ language incrementally.

What You Need to Use This Book

To use this book you need any of Visual Studio 2008 Standard Edition, Visual Studio 2008 Professional Edition, or Visual Studio 2008 Team System. Note that Visual C++ Express 2008 is *not* sufficient because the MFC is not included.

Visual Studio 2008 requires Windows XP (x86 or x64) with Service Pack 2 or later, Windows Server 2003 with Service Pack 1 or later, or any edition of Windows Vista except Starter Edition. To install any of the

three Visual Studio 2008 editions identified you need to have a 1.6 GHz processor with at least 384MB of memory (at least 768MB for Windows Vista) and at least 2.2GB of hard-disk space available. To install the full MSDN documentation that comes with the product you'll need an additional 1.8GB available on the installation drive.

The database examples in the book use the Northwind Traders database. You can find the download for this database by searching for "Northwind Traders" on `http://msdn.microsoft.com`. Of course, you can also adapt the examples to work with a database of your choice.

Most importantly, to get the most out of this book you need a willingness to learn, and a determination to master the most powerful programming tool for Windows applications presently available. You need the dedication to type in and work through all the examples and try out the exercises in the book. This sounds more difficult than it is, and I think you'll be surprised how much you can achieve in a relatively short period of time. Keep in mind that *everybody* who learns programming gets bogged down from time to time, but if you keep at it, things become clearer and you'll get there eventually. This book helps you to start experimenting on your own and, from there, to become a successful C++ programmer.

Using the Windows Classic Theme

If you're working in Windows Vista with Visual Studio 2008, you may have noticed that the view looks amazing. The transparency offered by the Aero Glass interface is quite breathtaking at first glance (and even many glances afterward). When you add in all of the visual effects that Vista has to offer, you might wonder why anyone would object to such a nice work environment. However, after a few hours of watching windows bursting forth and seeing the display dazzle your vision, you may prefer a setting that is less likely to cause eye fatigue. More importantly, you may notice a significant drop in your productivity because all of this eye candy robs your system of important processing cycles.

Eye candy is nice, but isn't it nicer to get home on time after a long day writing code? That's one reason why this book uses the Windows Classic theme to show Visual Studio 2008 windows. Another reason is that if you are still using Windows XP, the fancy Vista windows would not mean very much to you. The Windows Classic theme is common to both operating systems so it will fit with whatever operating system you are using, and it's definitely friendlier to your eyes than the Aero Glass interface.

If you are using Vista, I encourage you to try the various themes that Vista offers to see if they work for you. However, if you'd like to use the same theme in Vista as I have used for this book, then you can follow these steps to obtain it.

1. Right-click the Desktop and choose Personalize from the context menu. The Personalize window is shown in Figure I-1. This window provides access to all of the display settings you need to obtain the Windows Classic view.
2. Click Theme to display the Theme Settings dialog box shown in Figure I-2.
3. Choose Windows Classic in the Theme field and click OK. At this point, your display will begin looking very much like mine. Of course, you still have all of those special effects to consider. The next set of steps will get rid of the special effects.
4. Close the Personalize window.