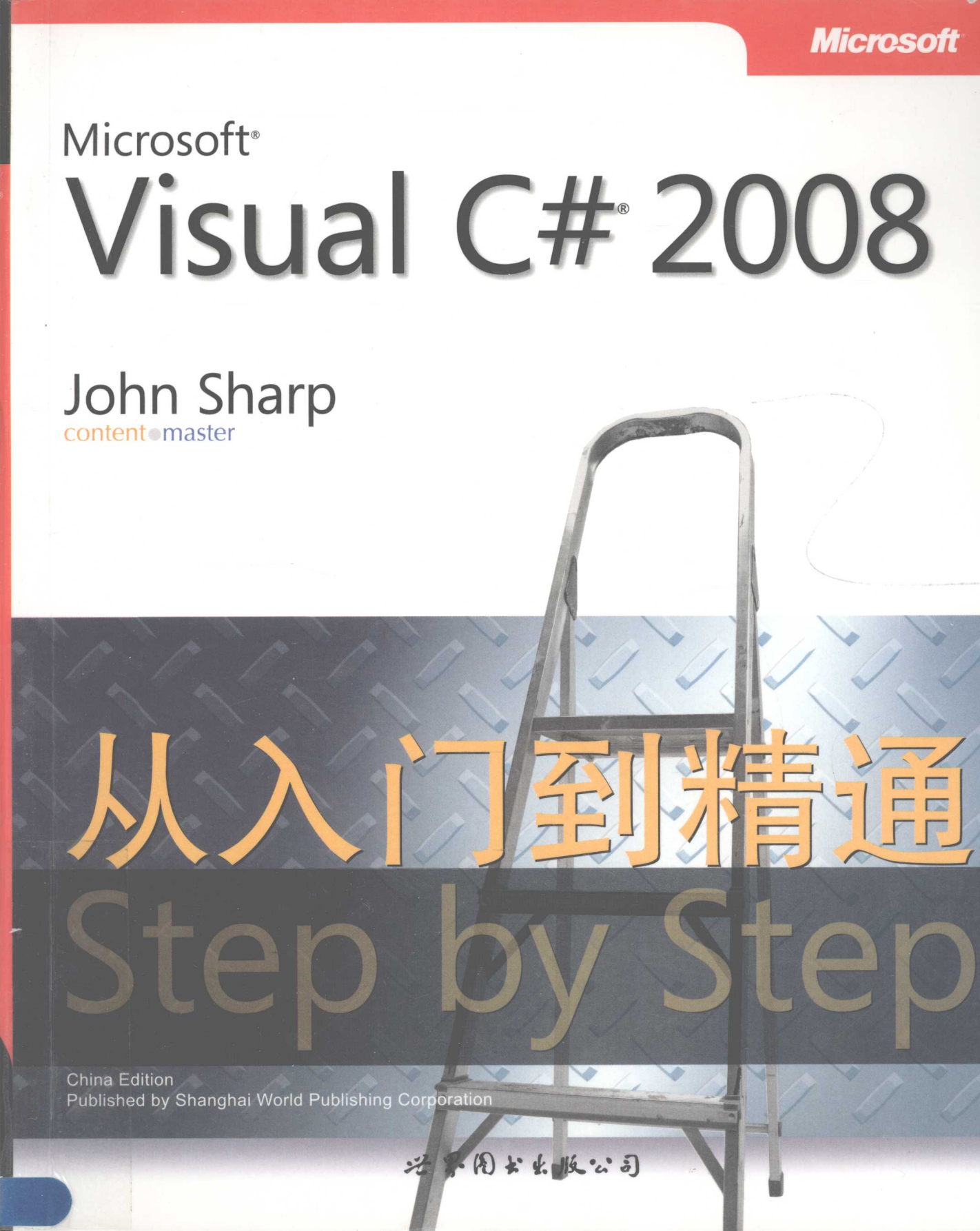
Microsoft

Microsoft®

Visual C#® 2008

John Sharp
content master



从入门到精通

Step by Step

China Edition
Published by Shanghai World Publishing Corporation

世界图书出版公司

Microsoft

Microsoft® Visual C#® 2008 Step by Step

John Sharp

图书在版编目(CIP)数据

Microsoft Visual C# 2008 从入门到精通:英文/(美)夏普(Master,C.)著. -上海:上海世界图书出版公司,2009.1

ISBN 978-7-5062-9172-9

Ⅰ.M… Ⅱ.夏… Ⅲ.C语言-程序设计-英文 Ⅳ.TP312

中国版本图书馆CIP数据核字(2008)第180648号

Microsoft Visual C# 2008 从入门到精通

[美]约翰·夏普(Content Master) 著

上海世界图书出版公司出版发行
上海市尚文路185号B楼
邮政编码200010
(公司电话:021-63783016转发行部)
上海竟成印务有限公司印刷
如发现印装质量问题,请与印刷厂联系
(质检科电话:021-56422678)
各地新华书店经销

开本:787×960 1/16 印张:44 字数:1400 000
2009年1月第1版 2009年1月第1次印刷
ISBN 978-7-5062-9172-9/T·187
图字:09-2008-627号
定价:188.00元
http://www.wpcsh.com.cn
http://www.mspress.com.cn

Acknowledgments

An old Latin proverb says "Tempora mutantur, nos et mutantur in illis," which roughly translates into English as "Times change, and we change with them." This proverb has a quaint, sedate feel and was obviously penned before the Romans had heard of Microsoft, Windows, the .NET Framework, and C#; otherwise, they would have written something more like "Times change, and we run like mad trying to keep up!" When I look back over the last seven or eight years, I am absolutely flabbergasted to see how much the .NET Framework, and the C# language in particular, has evolved. I am also very thankful, because it keeps me in gainful employment, performing biannual updates on this book. I am not complaining because the .NET Framework is a superb platform for building applications and services, and I thank the visionaries in the various product groups at Microsoft who have dedicated several millennia of person-years of effort in its development. In my opinion, C# is the greatest vehicle for taking full advantage of the .NET Framework. I have thoroughly enjoyed watching its development and learning the new features that each new release provides. This book is my attempt to convey my enthusiasm for the language to other programmers who are just starting along the C# path of discovery.

As with all projects of this type, writing a book is a group effort. The team I have had the pleasure of working with at Microsoft Press is second to none. In particular, I would like to single out Lynn Finnel who has kept the faith in me over several editions of this book, Christina Palaia and Jennifer Harris for their thorough editing of my manuscripts, and Stephen Sagman who has worked like a Trojan keeping us all in order and on schedule. I must pay special thanks to Kurt Meyer for his sterling efforts in reviewing my work, correcting my mistakes, and suggesting modifications, and of course to Jon Jagger who coauthored the first edition of this book with me back in 2001.

My long-suffering family have been wonderful, as they always are. Diana is now familiar with terms such as "DLINQ" and "lambda expression" and throws them into conversation with effortless aplomb. (For example, "Will you ever stop talking about DLINQ and lambda expressions?") James is still convinced that I spend my life playing computer games rather than working. Francesca has developed a frowning nod that says, "I have no idea what you are talking about, but I will nod anyway in the hope that you might stop." And Ginger, my arch-competitor for the chair in my study, has tried her best to completely distract me and delay my efforts in the ways that only a cat can.

As ever, "Up the Gills!"

—John Sharp

Introduction

Microsoft Visual C# is a powerful but simple language aimed primarily at developers creating applications by using the Microsoft .NET Framework. It inherits many of the best features of C++ and Microsoft Visual Basic but few of the inconsistencies and anachronisms, resulting in a cleaner and more logical language. With the advent of C# 2.0 in 2005, several important new features were added to the language, including generics, iterators, and anonymous methods. C# 3.0, available as part of Microsoft Visual Studio 2008, adds further features, such as extension methods, lambda expressions, and, most famously of all, the Language Integrated Query facility, or LINQ. The development environment provided by Visual Studio 2008 makes these powerful features easy to use, and the many new wizards and enhancements included in Visual Studio 2008 can greatly improve your productivity as a developer.

Who This Book Is For

The aim of this book is to teach you the fundamentals of programming with C# by using Visual Studio 2008 and the .NET Framework version 3.5. You will learn the features of the C# language, and then use them to build applications running on the Microsoft Windows operating system. By the time you complete this book, you will have a thorough understanding of C# and will have used it to build Windows Presentation Foundation (WPF) applications, access Microsoft SQL Server databases, develop ASP.NET Web applications, and build and consume a Windows Communication Foundation service.

Finding Your Best Starting Point in This Book

This book is designed to help you build skills in a number of essential areas. You can use this book if you are new to programming or if you are switching from another programming language such as C, C++, Sun Microsystems Java, or Visual Basic. Use the following table to find your best starting point.

If you are	Follow these steps
New to object-oriented programming	1. Install the practice files as described in the next section, "Installing and Using the Practice Files." 2. Work through the chapters in Parts I, II, and III sequentially. 3. Complete Parts IV, V, and VI as your level of experience and interest dictates.
Familiar with procedural programming languages such as C, but new to C#	1. Install the practice files as described in the next section, "Installing and Using the Practice Files." Skim the first five chapters to get an overview of C# and Visual Studio 2008, and then concentrate on Chapters 6 through 21. 2. Complete Parts IV, V, and VI as your level of experience and interest dictates.
Migrating from an object-oriented language such as C++ or Java	1. Install the practice files as described in the next section, "Installing and Using the Practice Files." 2. Skim the first seven chapters to get an overview of C# and Visual Studio 2008, and then concentrate on Chapters 8 through 21. 3. For information about building Windows-based applications and using a database, read Parts IV and V. 4. For information about building Web applications and Web services, read Part VI.
Switching from Visual Basic 6	1. Install the practice files as described in the next section, "Installing and Using the Practice Files." 2. Work through the chapters in Parts I, II, and III sequentially. 3. For information about building Windows-based applications, read Part IV. 4. For information about accessing a database, read Part V. 5. For information about creating Web applications and Web services, read Part VI. 6. Read the Quick Reference sections at the end of the chapters for information about specific C# and Visual Studio 2008 constructs.
Referencing the book after working through the exercises	1. Use the index or the table of contents to find information about particular subjects. 2. Read the Quick Reference sections at the end of each chapter to find a brief review of the syntax and techniques presented in the chapter.

Conventions and Features in This Book

This book presents information using conventions designed to make the information readable and easy to follow. Before you start, read the following list, which explains conventions you'll see throughout the book and points out helpful features that you might want to use.

Conventions

- Each exercise is a series of tasks. Each task is presented as a series of numbered steps (1, 2, and so on). A round bullet (•) indicates an exercise that has only one step.
- Notes labeled "tip" provide additional information or alternative methods for completing a step successfully.
- Notes labeled "important" alert you to information you need to check before continuing.
- Text that you type appears in bold.
- A plus sign (+) between two key names means that you must press those keys at the same time. For example, "Press Alt+Tab" means that you hold down the Alt key while you press the Tab key.

Other Features

- Sidebars throughout the book provide more in-depth information about the exercise. The sidebars might contain background information, design tips, or features related to the information being discussed.
- Each chapter ends with a Quick Reference section. The Quick Reference section contains quick reminders of how to perform the tasks you learned in the chapter.

System Requirements

You'll need the following hardware and software to complete the practice exercises in this book:

- Windows Vista Home Premium Edition, Windows Vista Business Edition, or Windows Vista Ultimate Edition. The exercises will also run using Microsoft Windows XP Professional Edition with Service Pack 2

Important If you are using Windows XP, some of the dialog boxes and screen shots described in this book might look a little different from those that you see. This is because of differences in the user interface in the Windows Vista operating system and the way in which Windows Vista manages security.

- Microsoft Visual Studio 2008 Standard Edition, Visual Studio 2008 Enterprise Edition, or Microsoft Visual C# 2008 Express Edition and Microsoft Visual Web Developer 2008 Express Edition
- Microsoft SQL Server 2005 Express Edition, Service Pack 2
- 1.6-GHz Pentium III+ processor, or faster
- 1 GB of available, physical RAM
- Video (800 × 600 or higher resolution) monitor with at least 256 colors
- CD-ROM or DVD-ROM drive
- Microsoft mouse or compatible pointing device

You will also need to have Administrator access to your computer to configure SQL Server 2005 Express Edition and to perform the exercises.

Code Samples

The companion CD inside this book contains the code samples that you'll use as you perform the exercises. By using the code samples, you won't waste time creating files that aren't relevant to the exercise. The files and the step-by-step instructions in the lessons also let you learn by doing, which is an easy and effective way to acquire and remember new skills.

Installing the Code Samples

Follow these steps to install the code samples and required software on your computer so that you can use them with the exercises.

1. Remove the companion CD from the package inside this book and insert it into your CD-ROM drive.

Note An end-user license agreement should open automatically. If this agreement does not appear, open My Computer on the desktop or Start menu, double-click the icon for your CD-ROM drive, and then double-click StartCD.exe.

2. Review the end-user license agreement. If you accept the terms, select the accept option, and then click *Next*.

   A menu will appear with options related to the book.

3. Click *Install Code Samples*.

4. Follow the instructions that appear.

   The code samples are installed to the following location on your computer:

   Documents\Microsoft Press\Visual CSharp Step By Step

Using the Code Samples

Each chapter in this book explains when and how to use any code samples for that chapter. When it's time to use a code sample, the book will list the instructions for how to open the files.

Important The code samples have been tested by using an account that is a member of the local Administrators group. It is recommended that you perform the exercises by using an account that has Administrator rights.

For those of you who like to know all the details, here's a list of the code sample Visual Studio 2008 projects and solutions, grouped by the folders where you can find them.

Project	Description
Chapter 1	
TextHello	This project gets you started. It steps through the creation of a simple program that displays a text-based greeting.
WPFHello	This project displays the greeting in a window by using Windows Presentation Foundation.
Chapter 2	
PrimitiveDataTypes	This project demonstrates how to declare variables by using each of the primitive types, how to assign values to these variables, and how to display their values in a window.
MathsOperators	This program introduces the arithmetic operators (+ – * / %).

Project	Description
Chapter 3	
Methods	In this project, you'll reexamine the code in the previous project and investigate how it uses methods to structure the code.
DailyRate	This project walks you through writing your own methods, running the methods, and stepping through the method calls by using the Visual Studio 2008 debugger.
Chapter 4	
Selection	This project shows how to use a cascading *if* statement to implement complex logic, such as comparing the equivalence of two dates.
SwitchStatement	This simple program uses a *switch* statement to convert characters into their XML representations.
Chapter 5	
WhileStatement	This project uses a *while* statement to read the contents of a source file one line at a time and display each line in a text box on a form.
DoStatement	This project uses a *do* statement to convert a decimal number to its octal representation.
Chapter 6	
MathsOperators	This project reexamines the MathsOperators project from Chapter 2, "Working with Variables, Operators, and Expressions," and causes various unhandled exceptions to make the program fail. The *try* and *catch* keywords then make the application more robust so that it no longer fails.
Chapter 7	
Classes	This project covers the basics of defining your own classes, complete with public constructors, methods, and private fields. It also shows how to create class instances by using the *new* keyword and how to define static methods and fields.
Chapter 8	
Parameters	This program investigates the difference between value parameters and reference parameters. It demonstrates how to use the *ref* and *out* keywords.
Chapter 9	
StructsAndEnums	This project defines a *struct* type to represent a calendar date.

Project	Description
Chapter 10	
Cards	This project uses the *ArrayList* collection class to group together playing cards in a hand.
Chapter 11	
ParamsArrays	This project demonstrates how to use the *params* keyword to create a single method that can accept any number of *int* arguments.
Chapter 12	
Vehicles	This project creates a simple hierarchy of vehicle classes by using inheritance. It also demonstrates how to define a virtual method.
ExtensionMethod	This project shows how to create an extension method for the *int* type, providing a method that converts an integer value from base 10 to a different number base.
Chapter 13	
Tokenizer	This project uses a hierarchy of interfaces and classes to simulate both reading a C# source file and classifying its contents into various kinds of tokens (identifiers, keywords, operators, and so on). As an example of use, it also derives classes from the key interfaces to display the tokens in a rich text box in color syntax.
Chapter 14	
UsingStatement	This project revisits a small piece of code from Chapter 5, "Using Compound Assignment and Iteration Statements," and reveals that it is not exception-safe. It shows you how to make the code exception-safe with a *using* statement.
Chapter 15	
WindowProperties	This project presents a simple Windows application that uses several properties to display the size of its main window. The display updates automatically as the user resizes the window.
AutomaticProperties	This project shows how to create automatic properties for a class and use them to initialize instances of the class.
Chapter 16	
Indexers	This project uses two indexers: one to look up a person's phone number when given a name, and the other to look up a person's name when given a phone number.
Chapter 17	
Delegates	This project displays the time in digital format by using delegate callbacks. The code is then simplified by using events.

Project	Description
Chapter 18	
BinaryTree	This solution shows you how to use generics to build a typesafe structure that can contain elements of any type.
BuildTree	This project demonstrates how to use generics to implement a typesafe method that can take parameters of any type.
Chapter 19	
BinaryTree	This project shows you how to implement the generic IEnumerator<T> interface to create an enumerator for the generic BinaryTree class.
IteratorBinaryTree	This solution uses an iterator to generate an enumerator for the generic BinaryTree class.
Chapter 20	
QueryBinaryTree	This project shows how to use LINQ queries to retrieve data from a binary tree object.
Chapter 21	
Operators	This project builds three structs, called Hour, Minute, and Second, that contain user-defined operators. The code is then simplified by using a conversion operator.
Chapter 22	
BellRingers	This project is a Windows Presentation Foundation application demonstrating how to define styles and use basic WPF controls.
Chapter 23	
BellRingers	This project is an extension of the application created in Chapter 22, "Introducing Windows Presentation Foundation," but with drop-down and pop-up menus added to the user interface.
Chapter 24	
CustomerDetails	This project demonstrates how to implement business rules for validating user input in a WPF application using customer information as an example.
Chapter 25	
ReportOrders	This project shows how to access a database by using ADO.NET code. The application retrieves information from the Orders table in the Northwind database.
DLINQOrders	This project shows how to use DLINQ to access a database and retrieve information from the Orders table in the Northwind database.

Project	Description
Chapter 26	
Suppliers	This project demonstrates how to use data binding with a WPF application to display and format data retrieved from a database in controls on a WPF form. The application also enables the user to modify information in the Products table in the Northwind database.
Chapter 27	
Litware	This project creates a simple Microsoft ASP.NET Web site that enables the user to input information about employees working for a fictitious software development company.
Chapter 28	
Litware	This project is an extended version of the Litware project from the previous chapter and shows how to validate user input in an ASP.NET Web application.
Chapter 29	
Northwind	This project shows how to use Forms-based security for authenticating the user. The application also demonstrates how to use ADO.NET from an ASP.NET Web form, showing how to query and update a database in a scalable manner, and how to create applications that span multiple Web forms.
Chapter 30	
NorthwindServices	This project implements a Windows Communication Foundation Web service, providing remote access across the Internet to data in the *Products* table in the Northwind database.

Uninstalling the Code Samples

Follow these steps to remove the code samples from your computer.

1. In *Control Panel*, open *Add or Remove Programs*.
2. From the list of Currently Installed Programs, select Microsoft Visual C# 2008 Step by Step.
3. Click *Remove*.
4. Follow the instructions that appear to remove the code samples.

Support for This Book

Every effort has been made to ensure the accuracy of this book and the contents of the companion CD. As corrections or changes are collected, they will be added to a Microsoft Knowledge Base article.

Microsoft Press provides support for books and companion CDs at the following Web site:

http://www.microsoft.com/learning/support/books/

Questions and Comments

If you have comments, questions, or ideas regarding the book or the companion CD, or questions that are not answered by visiting the site above, please send them to Microsoft Press via e-mail to

mspinput@microsoft.com

Or via postal mail to

Microsoft Press
Attn: *Microsoft Visual C# 2008 Step by Step* Series Editor
One Microsoft Way
Redmond, WA 98052-6399

Please note that Microsoft software product support is not offered through the above addresses.

John Sharp

John Sharp is a Principal Technologist at Content Master (*www.contentmaster.com*), part of CM Group, a technical authoring company in the United Kingdom. He researches and develops technical content for training courses, seminars, and white papers. John is deeply involved with Microsoft .NET Framework application development and interoperability. He has written papers and courses, built tutorials, and delivered conference presentations covering distributed systems and Web services, application migration and interoperability between Microsoft Windows/.NET Framework and UNIX/Linux/Java, as well as development using the C# and J# languages. John has also authored *Microsoft Windows Communication Foundation Step by Step* and *Microsoft Visual J# Core Reference*, both published by Microsoft Press.

Contents at a Glance