

Microsoft

Core Reference



MICROSOFT
VISUAL C#[™]
.NET

Microsoft
.net

Mickey Williams

Microsoft



MICROSOFT
VISUAL C#TM
.NET

Microsoft
.net

Mickey Williams

PUBLISHED BY

Microsoft Press

A Division of Microsoft Corporation

One Microsoft Way

Redmond, Washington 98052-6399

Copyright © 2002 by Microsoft Corporation

All rights reserved. No part of the contents of this book may be reproduced or transmitted in any form or by any means without the written permission of the publisher.

Library of Congress Cataloging-in-Publication Data

Williams, Mickey.

Microsoft Visual C# (core reference) / Mickey Williams.

p. cm.

Includes index.

ISBN 0-7356-1290-0

1. Microsoft Visual C#. 2. C# (Computer program language) I. Title.

QA76.736.C153 .W55235 2002

005.26'8--dc21

2001059058

Printed and bound in the United States of America.

1 2 3 4 5 6 7 8 9 QWT 7 6 5 4 3 2

Distributed in Canada by Penguin Books Canada Limited.

A CIP catalogue record for this book is available from the British Library.

Microsoft Press books are available through booksellers and distributors worldwide. For further information about international editions, contact your local Microsoft Corporation office or contact Microsoft Press International directly at fax (425) 936-7329. Visit our Web site at www.microsoft.com/mspress. Send comments to mspinput@microsoft.com.

ActiveX, Authenticode, Encarta, FrontPage, IntelliMouse, IntelliSense, JScript, Microsoft, Microsoft Press, MSDN, .NET logo, OpenType, Verdana, Visual Basic, Visual C++, Visual C#, Visual Studio, Win32, Windows, Windows NT, and Xbox are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Other product and company names mentioned herein may be the trademarks of their respective owners.

The example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious. No association with any real company, organization, product, domain name, e-mail address, logo, person, place, or event is intended or should be inferred.

Acquisitions Editor: Danielle Bird

Project Editor: Dick Brown

Technical Editors: Donnie Cameron and Julie Xiao

Body Part No. X08-06141

*For my racing buddies Ali and Mackenzie,
and for my wife René,
who has been waiting for a trip to Banff for 10 years now.*

Acknowledgments

No author can create a book of this scope working alone—the number of people who come together on a project like this is amazing. An author agrees to write a book, and as the process begins, people start coming out of the woodwork to help. It's a lot like the Microsoft Office Assistant: "Hey! It looks like you're writing a book! Want some help? How about some great graphics for the part openers in your book? This heading would be great if it was in a gerund form!"

At Microsoft Press, I'd like to thank Danielle Bird for asking me to write this book. Danielle, I promise to stop hounding you for an Xbox console. I'd also like to thank Dick Brown, who is doomed to a life of trying to get authors like me to hit deadlines. I'm sure Dick is happy to see the last bits of this book cross his desk. Jennifer Harris was responsible for turning my geeky text into readable English sentences. Jennifer, I'm sorry about forcing you to read my long-winded digression on the significance of the Battle of Agincourt. Thanks to Donnie Cameron and Julie Xiao for their excellent work as technical editors. I've never been so thoroughly reviewed and cross-checked. Thanks for your diligence and support.

A big thank you to the other authors, who provided me some much-needed help: Brad Jones, Brad Wist, Mike Pope, and Christoph Wille. Thanks, guys.

Thanks to the people I work with at Interactive Software Engineering and .NET Experts in Santa Barbara: Bertrand and Annie Meyer, Raphaël Simon, Manu Stapf, and Bill Navickas. Thanks for working around my schedule and for providing a great place for me to train and consult with other developers. Special thanks to Hal Webre, my training cohort at .NET Experts. Hal kept me well supplied with CDs and cookbooks. While we were at the ACM SIGCSE conference recently, Hal took me out for Cincinnati-style chili, which was interesting, but why it's called chili is beyond me.

And it's not just to the people who deal with me professionally that I owe thanks. Thanks to my sister, Jennifer, and her husband, Mitch, for providing a base where I could write while I was in San Diego, as well as for scheduling new additions to their family around my book writing timetable.

And finally, thanks to my daughters, Ali and Mackenzie, for remembering what I look like. And thanks to my wife, René for being far more patient than I deserve. Next year we're definitely going to Banff.

Introduction

When I first started working with Microsoft Visual C# .NET, I was immediately struck by how productive my coding sessions became. Even though I was working with the pre-beta release, the development environment and the .NET Framework simplified much of the work that I was doing. The increased productivity was due to four factors:

- The C# programming language, which is well-suited for component-based development
- The simplified programming and deployment model offered by the common language runtime
- The .NET Framework and its associated class library
- The developer automation features included in the Microsoft Visual Studio .NET environment

The tools for development that are realized today with Visual C# .NET are light-years ahead of the tools I was using in the mid-1980s when I started developing for Microsoft Windows. When I was asked to write a book about Visual C# .NET, I just couldn't refuse. I'm sure that you'll find working with Visual C# .NET a rewarding experience, and I'm pleased that you've selected this book to help get you started.

I enjoy talking to readers, and I'm happy to answer any questions you might have after reading my books. If you have any feedback or questions about this text, feel free to contact me at mickey.williams@codevtech.com. I can't always respond immediately, but I try to answer all of my e-mail.

Who Should Read This Book

This book is intended for developers who want to learn the C# programming language using Microsoft Visual C# .NET. The book is intended to provide a broad overview of programming with Visual C# .NET, so it explores a wide variety of topics. In addition to a discussion of the C# programming language, this

book covers the more commonly used classes in the .NET Framework, including Windows Forms programming, ASP.NET, Extensible Markup Language (XML), XML Web services, and more.

System Requirements

To use this book, you'll need to have Visual C# .NET or Visual Studio .NET installed on your computer. Any computer that satisfies the minimum requirements for Visual C# .NET will work effectively with most of the examples in this book. Be aware that Microsoft Windows XP Home Edition and Microsoft Windows NT 4 don't support the hosting of XML Web services or ASP.NET Web applications with the .NET Framework. Although you can build these projects on these operating systems, you'll need to upload the projects to a properly configured host to execute them. I recommend that development be done on the Microsoft Windows XP Professional or Microsoft Windows 2000 Professional operating systems. For Chapter 20, you need IIS version 4 or later in order to run the sample code.

Organization

This book is arranged in five parts, with each part covering a different aspect of programming using Visual C# .NET:

- **Part I** Introduces the Visual C# .NET environment and examines the core portions of the C# programming language.
- **Part II** Describes the more advanced aspects of the C# programming language and the .NET Framework. In addition to chapters on delegates and exception handling, we'll cover topics such as collection classes, debugging, and writing multithreaded code.
- **Part III** Here you'll learn how to write graphical user interface (GUI) applications using Visual C# .NET. The .NET Framework includes extensive support for writing Windows Forms–based applications, and you'll see that many advanced controls are very easy to use.
- **Part IV** Examines ADO.NET and XML. You'll learn how to use ADO.NET for data access to databases such as SQL Server. We'll also look at the .NET Framework classes that implement a new model for manipulating XML, in which data is pulled from an XML stream. With the .NET Framework and Visual C# .NET, it's easy to manage data using either ADO.NET or XML.

- **Part V** The book wraps up with a discussion of creating Web applications using ASP.NET and Visual C# .NET and shows you how to use XML Web services.

Sample Files

Sample files for this book can be found at the Microsoft Press Web site at <http://www.microsoft.com/mspress/books/5029.asp>. Clicking the Companion Content link takes you to a page from which you can download the samples.

The sample files can also be found on the book's companion CD, along with other supplemental content. To access those files, insert the companion CD in your computer's CD-ROM drive and make a selection from the menu that appears. If the AutoRun feature is not enabled on your system (if a menu doesn't appear when you insert the disc in your computer's CD-ROM drive), run StartCD.exe in the root folder of the companion CD. Installing the sample files on your hard disk requires approximately 7.8 MB of disk space. If you have trouble running any of these files, refer to the text in the book that describes these programs.

Aside from the sample files that this book discusses, the book's supplemental content includes two eBook installations: a stand-alone eBook installation, and a Visual Studio Help eBook installation. The stand-alone eBook installation will allow you to access an electronic version of the print book directly from your desktop. The Visual Studio Help eBook installation will allow you to access a second electronic version of the print book directly from the Visual Studio .NET help system.

Visual Studio .NET Professional Trial Version

In addition to the companion CD, this book includes a DVD with an evaluation copy of Microsoft Visual Studio .NET Professional. This evaluation copy can help you follow the examples in this book and get you started learning Microsoft Visual C# .NET, but the software must be activated, and it will expire and stop working 60 days after you install it. Also, your computer must have a DVD-ROM drive. You can learn more about this evaluation copy and its system requirements at <http://msdn.microsoft.com/vstudio/productinfo/trial.asp>.

Support

Every effort has been made to ensure the accuracy of this book and the contents of the companion CD. Microsoft Press provides corrections for books through the World Wide Web at the following address:

<http://www.microsoft.com/mspress/support>

To connect directly to the Microsoft Press Knowledge Base and enter a query regarding a question or an issue that you may have, go to:

<http://www.microsoft.com/mspress/support/search.asp>

If you have comments, questions, or ideas regarding this book or the companion content, or questions that are not answered by querying the Knowledge Base, please send them to Microsoft Press via e-mail to:

mspinput@microsoft.com

Or via postal mail to:

Microsoft Press

Attn: *Microsoft Visual C# .NET (Core Reference)* Editor

One Microsoft Way

Redmond, WA 98052-6399

Please note that product support is not offered through the above addresses. For product support information, please visit the Microsoft Support Web site at:

<http://support.microsoft.com>

Contents at a Glance

Part I	Introducing Microsoft Visual C# .NET	
1	A Tour of Visual Studio .NET and Visual C# .NET	3
2	C# Basics	33
3	Value Types and Reference Types	81
4	Operators, Type Conversions, and Properties	101
5	Flow of Control	133
Part II	Advanced C#	
6	Delegates and Attributes	159
7	Indexers and Enumerators	193
8	Collections and Sorting	211
9	Debugging Techniques	237
10	Advanced Topics	267
Part III	Programming Windows Forms	
11	An Introduction to Windows Forms	299
12	Adding Controls to Forms	325
13	User Input and Feedback	371
14	GDI+	423
15	Advanced Controls	475
16	Grid, Tree, and List Controls	509
17	Enhancing Windows Forms	541
Part IV	Data Access and XML	
18	ADO.NET	567
19	XML	597
Part V	ASP.NET and Web Services	
20	Web Forms	635
21	SOAP and Web Services	683

Table of Contents

	Acknowledgments	xxi
	Introduction	xxiii
Part I	Introducing Microsoft Visual C# .NET	
1	A Tour of Visual Studio .NET and Visual C# .NET	3
	Creating Applications for the .NET Platform	4
	The .NET Framework	4
	Understanding Assemblies	5
	Overview of Visual Studio .NET	9
	The Start Page	9
	Visual Studio .NET Windows	12
	Visual Studio .NET Options	18
	Visual Studio .NET Help	21
	Creating Visual C# Solutions	22
	Using Project Templates	22
	Generating a New Solution	24
	Layout of a Typical Visual C# Program	25
	Commenting Your Source	26
	Using Namespaces	27
	Declaring a Class	28
	Defining a <i>Main</i> Method	28
	Compiling Your Project	29
	Providing Output for Hello World	30
	Conclusion	31
2	C# Basics	33
	Basic Data Types	34
	The .NET Common Type System	35
	The Common Language Specification	36
	Classes	37
	Adding a Class in Visual C#	37
	Declaring a Class	38
	An Introduction to Inheritance	39

	The <i>object</i> Base Class	44
	Members	45
	Inheritance and Methods	55
	Accessibility	58
	Namespaces	61
	Creating a New Namespace	62
	Using a Namespace	63
	Adding References	64
	Interfaces	66
	Using Interfaces	66
	Classes and Interfaces	67
	Managing Errors with Exceptions	67
	Handling Exceptions	68
	.NET Framework Exceptions	73
	Casting	74
	Structures	76
	Structures and Inheritance	76
	Allocating Structures	77
	Member Functions	77
	Enumerations	78
	Conclusion	79
3	Value Types and Reference Types	81
	Understanding Value Types	82
	Value Type Lifetime	83
	Boxing	84
	Declaring New Value Types	85
	Understanding Reference Types	85
	Working with Arrays	86
	Working with Strings	91
	Reference Type Lifetime and Garbage Collection	93
	Conclusion	100
4	Operators, Type Conversions, and Properties	101
	Examining Operators	102
	Comparing Objects with the Relational Operators	103
	Forming Logical Expressions with the Logical Operators	104
	Doing Math with the Arithmetic Operators	105

	Managing Bit Patterns with the Bitwise Operators	107
	Setting Variables with the Assignment Operators	110
	Using the Conditional Operator	112
	Obtaining Type Information	113
	Addressing Memory Directly	113
	Using Other Operators	114
	Working with Operators	115
	Understanding Operator Precedence	115
	Using the <i>checked</i> and <i>unchecked</i> Keywords	116
	Defining Operators for Your Types	117
	Creating a New Value Type	119
	Controlling the Behavior of <i>&&</i> and <i>//</i>	123
	Converting Types	125
	Performing Implicit Conversions	125
	Performing Explicit Conversions	126
	Using the <i>Convert</i> Class	127
	Performing User-Defined Conversions	127
	Basic String Formatting for Numeric Values	128
	Using Properties as Smart Fields	130
	Conclusion	131
5	Flow of Control	133
	Selecting Code Paths Conditionally	134
	The <i>if</i> Statement	134
	The <i>switch</i> Statement	137
	Building Loops Using Iteration Statements	140
	The <i>for</i> Loop	140
	The <i>foreach</i> Loop	143
	The <i>while</i> Loop	145
	The <i>do</i> Loop	147
	Using Jump Statements to Transfer Control	149
	The <i>break</i> Statement	150
	The <i>continue</i> Statement	151
	The <i>goto</i> Statement	152
	The <i>return</i> Statement	153
	The <i>throw</i> Statement	153
	Transferring Control out of an Exception Handling Block	154
	Conclusion	155

Part II Advanced C#

6	Delegates and Attributes	159
	Using Delegates	160
	Creating Delegates	161
	Using Delegates as Callback Methods	162
	Using Delegates as Functors	165
	Combining Delegates	166
	Iterating Manually over the Delegate Chain	168
	Using Delegates with Non-Static Methods	170
	Handling Events	173
	Using Event Fields	175
	Raising Events	175
	An Example Using Events	176
	Working with Attributes	179
	Using Attributes	179
	Resolving Attributes	180
	Defining Custom Attributes	181
	Creating a Strong Name	190
	Conclusion	192
7	Indexers and Enumerators	193
	Using Indexers	194
	Declaring an Indexer	195
	Declaring Indexers for Interfaces	197
	An Indexer Example	198
	Using Enumerators	201
	Understanding Enumeration Interfaces	202
	Implementing Enumeration Interfaces	203
	Consuming Enumeration Interfaces	207
	Conclusion	208
8	Collections and Sorting	211
	Comparing and Sorting	211
	Creating Comparable Types with the <i>IComparable</i> Interface	212
	Building Comparison Classes with the <i>IComparer</i> Interface	214

	Storing Objects in Collection Classes	219
	Commonly Used Collection Interfaces	219
	Using the <i>Queue</i> Class	224
	Using the <i>Stack</i> Class	229
	Using the <i>Hashtable</i> Class	231
	Using the Specialized Collection Classes	234
	Conclusion	236
9	Debugging Techniques	237
	Compiling Code Conditionally	237
	Tracing the Stack	241
	Displaying the Call Stack	241
	Accessing Detailed Method Information	243
	Using the <i>Trace</i> and <i>Debug</i> Classes	244
	Generating Program Trace Information	245
	Displaying Messages with the <i>Trace</i> and <i>Debug</i> Classes	245
	Controlling Output with Switches	249
	Consuming Trace Messages with Trace Listeners	252
	Debugging with Visual Studio .NET	261
	Setting Breakpoints	261
	Using the Call Stack	264
	Conclusion	265
10	Advanced Topics	267
	Writing Multithreaded Code	267
	Understanding Application Domains	268
	Working with Threads	268
	Synchronizing Access to Shared Objects	273
	Working Asynchronously Using the Thread Pool	284
	Using Unsafe Code	289
	Examining the Need for Unsafe Code	289
	Declaring Unsafe Contexts	290
	Conclusion	296

Part III Programming Windows Forms

11	An Introduction to Windows Forms	299
	Understanding Windows Forms	300
	Using Forms as Dialog Boxes	300
	Using Forms as Views	301
	Developing a Simple Windows Forms Project	301
	Examining Files Created by Visual C# .NET	302
	Executing a Windows Forms Project	306
	Adding a New Form to a Project	307
	Modal Forms vs. Modeless Forms	308
	Determining the <i>DialogResult</i> Value	308
	Passing Values to Forms	309
	Displaying Information with Message Boxes	311
	Specifying Message Box Buttons	311
	Adding Icons to Message Boxes	312
	Defining Default Buttons for Message Boxes	313
	Controlling Special Cases for Message Boxes	313
	Specifying a Parent Window for a Message Box	314
	Controlling a Windows Forms Application	315
	Starting an Application	315
	Determining Application Path Information	317
	Performing Idle Work	318
	Closing an Application	318
	Using Form Properties to Affect Form Behavior	320
	Setting Border Styles	320
	Defining Other Form Properties	321
	Conclusion	323
12	Adding Controls to Forms	325
	An Introduction to Controls	326
	Using the Button Classes	327
	The <i>ButtonBase</i> Class	327
	The <i>Button</i> Class	327
	The <i>RadioButton</i> Class	334
	The <i>CheckBox</i> Class	335

Basic Event Handling	336
Event Handling Architecture	336
Adding New Event Handlers	337
Removing Event Handlers	340
Translating Key Presses into Button Clicks	340
Validating the Contents of Controls	341
Using the <i>TextBox</i> Class	343
Hiding Passwords in a Text Box Control	344
Working with Multiline Text Box Controls	344
Using Other Text Box Properties	346
Using the <i>Label</i> Class	347
Using the <i>LinkLabel</i> Class	347
Handling the <i>Click</i> Event	348
Unique <i>LinkLabel</i> Properties	348
Using the <i>ListBox</i> Class	350
Adding Items to a List Box	351
Removing Items from a List Box	353
Preventing Repainting When Updating a List Box	354
Selecting List Box Items	354
Using Other List Box Properties	359
Using the <i>CheckedListBox</i> Class	360
Checking Items in a Checked List Box	361
Handling Events from the Checked List Box	362
Using the <i>ComboBox</i> Class	363
Using Combo Box Properties	363
Handling Combo Box Events	365
Scrolling Controls with the <i>ScrollBar</i> Classes	366
Controls That Act as Containers	367
Embedding Controls	368
The <i>GroupBox</i> Class	369
The <i>Panel</i> Class	369
Conclusion	370