

Join the discussion @ p2p.wrox.com



Wrox Programmer to Programmer™



Beginning C#[®] 6 Programming with Visual Studio[®] 2015

Benjamin Perkins, Jacob Vibe Hammer, Jon D. Reid

BEGINNING

**C#[®] 6 Programming with
Visual Studio[®] 2015**

BEGINNING

C#[®] 6 Programming with Visual Studio[®] 2015

Benjamin Perkins

Jacob Vibe Hammer

Jon D. Reid



Beginning C#® 6 Programming with Visual Studio® 2015

Published by
John Wiley & Sons, Inc.
10475 Crosspoint Boulevard
Indianapolis, IN 46256
www.wiley.com

Copyright © 2016 by John Wiley & Sons, Inc., Indianapolis, Indiana

Published simultaneously in Canada

ISBN: 978-1-119-09668-9
ISBN: 978-1-119-09655-9 (ebk)
ISBN: 978-1-119-09656-6 (ebk)

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permissions>.

Limit of Liability/Disclaimer of Warranty: The publisher and the author make no representations or warranties with respect to the accuracy or completeness of the contents of this work and specifically disclaim all warranties, including without limitation warranties of fitness for a particular purpose. No warranty may be created or extended by sales or promotional materials. The advice and strategies contained herein may not be suitable for every situation. This work is sold with the understanding that the publisher is not engaged in rendering legal, accounting, or other professional services. If professional assistance is required, the services of a competent professional person should be sought. Neither the publisher nor the author shall be liable for damages arising herefrom. The fact that an organization or Web site is referred to in this work as a citation and/or a potential source of further information does not mean that the author or the publisher endorses the information the organization or Web site may provide or recommendations it may make. Further, readers should be aware that Internet Web sites listed in this work may have changed or disappeared between when this work was written and when it is read.

For general information on our other products and services please contact our Customer Care Department within the United States at (877) 762-2974, outside the United States at (317) 572-3993 or fax (317) 572-4002.

Wiley publishes in a variety of print and electronic formats and by print-on-demand. Some material included with standard print versions of this book may not be included in e-books or in print-on-demand. If this book refers to media such as a CD or DVD that is not included in the version you purchased, you may download this material at <http://book-support.wiley.com>. For more information about Wiley products, visit www.wiley.com.

Library of Congress Control Number: 2015957031

Trademarks: Wiley, the Wiley logo, Wrox, the Wrox logo, Programmer to Programmer, and related trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries, and may not be used without written permission. Visual C# is a registered trademark of Microsoft Corporation. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc., is not associated with any product or vendor mentioned in this book.

ABOUT THE AUTHORS

BENJAMIN PERKINS (MBA, MCSD, ITIL) is currently employed at Microsoft in Munich, Germany, as a Senior Technical Advisor for IIS, ASP.NET, and Azure App Services. He has been working professionally in the IT industry for over two decades. He started computer programming with QBasic at the age of 11 on an Atari 1200XL desktop computer. He takes pleasure in the challenges that troubleshooting technical issues have to offer and savors in the rewards of a well written program. After completing high school he joined the United States Army. After successfully completing his military service, he attended Texas A&M University in College Station, Texas, where he received a Bachelor of Business Administration in Management Information Systems.

His roles in the IT industry have spanned the entire spectrum including programmer, system architect, technical support engineer, team leader, and mid-level management. While employed at Hewlett-Packard, he received numerous awards, degrees, and certifications. He has a passion for technology and customer service and looks forward to troubleshooting and writing more world-class technical solutions.

“My approach is to write code with support in mind, and to write it once correctly and completely so we do not have to come back to it again, except to enhance it.”

Benjamin is married to Andrea and has two wonderful children, Lea and Noa.

JACOB VIBE HAMMER is a software architect and developer at Kamstrup, where he is helping the company develop world-class Smart Grid solutions for large public utilities. He started programming just about the time when he was able to spell the word “BASIC” — which, incidentally, is the first programming language he ever used. Since then, he has worked with numerous programming languages and solution architectures; however, since the turn of the century, he has worked primarily with the .NET platform. Today, his programming time is spent working primarily with C# and WPF, as well as toying with NoSQL databases. A Danish citizen, Jacob lives in Aarhus, Denmark, with his wife and two sons.

JON D. REID is a Product Solution Manager for IFS Field Service Management (www.IFSWORLD.com). He has coauthored a number of books, including *Beginning Visual C# 2010*, *Fast Track C#*, *Pro Visual Studio .NET*, and many others.

ABOUT THE TECHNICAL EDITOR

JOHN MUELLER is a freelance author and technical editor. He has writing in his blood, having produced 99 books and more than 600 articles to date. The topics range from networking to artificial intelligence and from database management to heads-down programming. Some of his current books include books on topics such as Python for beginners, Python for data scientists, and MATLAB. He has also written a Java e-learning kit, a book on HTML5 development with JavaScript, and another on CSS3. His technical editing skills have helped more than 63 authors refine the content of their manuscripts. John has provided technical editing services to both *Data Based Advisor* and *Coast Compute* magazines. Be sure to read John's blog at <http://blog.johnmuellerbooks.com/>.

ACKNOWLEDGMENTS

It takes a lot of work to get content into a presentable format for students and IT professionals to read and get value from. The authors indeed have technical knowledge and experiences to share, but without the technical writers, technical reviewers, developers, editors, publishers, graphic designers, the list goes on, providing their valuable input, a book of high quality could not be written. The rate of change occurs too quickly for an individual to perform all these tasks and still publish a book that is valid before the technology becomes stale. This is why authors worked together with a great team to get all the components of the book together quickly. It was done to ensure that the most up to date information gets to the reader while the features are still fresh and current. I would like to thank Kelly Talbot for his great project management and technical review of the content as well as John Mueller for his technical review and suggestions throughout the process. Lastly, I would like to thank all the numerous people behind the scenes who helped get this book together.

CREDITS

Senior Acquisitions Editor

Kenyon Brown

Project Editor

Kelly Talbot

Technical Editor

John Mueller

Production Editor

Joel Jones

Copy Editor

Kelly Talbot Editing Services

Manager of Content Development & Assembly

Mary Beth Wakefield

Production Manager

Kathleen Wisor

Marketing Director

David Mayhew

Marketing Manager

Carrie Sherrill

Professional Technology & Strategy Director

Barry Pruett

Business Manager

Amy Knies

Associate Publisher

Jim Minatel

Project Coordinator, Cover

Brent Savage

Proofreader

Nancy Bell

Indexer

Jack Lewis

Cover Designer

Wiley

Cover Image

© Nomad_Soul/Shutterstock

CONTENTS

INTRODUCTION

xix

PART I: THE OOP LANGUAGE

CHAPTER 1: INTRODUCING C#	3
What Is the .NET Framework?	4
What's in the .NET Framework?	4
Writing Applications Using the .NET Framework	5
What Is C#?	8
Applications You Can Write with C#	9
C# in this Book	10
Visual Studio 2015	10
Visual Studio Express 2015 Products	10
Solutions	11
CHAPTER 2: WRITING A C# PROGRAM	13
The Visual Studio 2015 Development Environment	14
Console Applications	17
The Solution Explorer	20
The Properties Window	21
The Error List Window	22
Desktop Applications	22
CHAPTER 3: VARIABLES AND EXPRESSIONS	29
Basic C# Syntax	30
Basic C# Console Application Structure	33
Variables	34
Simple Types	34
Variable Naming	39
Literal Values	39
Expressions	42
Mathematical Operators	42
Assignment Operators	47
Operator Precedence	48
Namespaces	49

CHAPTER 4: FLOW CONTROL	53
Boolean Logic	54
Boolean Bitwise and Assignment Operators	56
Operator Precedence Updated	58
Branching	59
The Ternary Operator	59
The if Statement	59
The switch Statement	63
Looping	66
do Loops	66
while Loops	69
for Loops	71
Interrupting Loops	72
Infinite Loops	73
CHAPTER 5: MORE ABOUT VARIABLES	77
Type Conversion	78
Implicit Conversions	78
Explicit Conversions	80
Explicit Conversions Using the Convert Commands	83
Complex Variable Types	85
Enumerations	85
Structs	89
Arrays	92
String Manipulation	99
CHAPTER 6: FUNCTIONS	107
Defining and Using Functions	108
Return Values	110
Parameters	112
Variable Scope	119
Variable Scope in Other Structures	122
Parameters and Return Values versus Global Data	123
The Main() Function	125
Struct Functions	127
Overloading Functions	128
Using Delegates	130

CHAPTER 7: DEBUGGING AND ERROR HANDLING	135
Debugging in Visual Studio	136
Debugging in Nonbreak (Normal) Mode	136
Debugging in Break Mode	144
Error Handling	153
try...catch...finally	153
Listing and Configuring Exceptions	160
CHAPTER 8: INTRODUCTION TO OBJECT-ORIENTED PROGRAMMING	163
What Is Object-Oriented Programming?	164
What Is an Object?	165
Everything's an Object	168
The Life Cycle of an Object	168
Static and Instance Class Members	169
OOP Techniques	170
Interfaces	171
Inheritance	172
Polymorphism	175
Relationships between Objects	177
Operator Overloading	179
Events	180
Reference Types versus Value Types	180
OOP in Desktop Applications	180
CHAPTER 9: DEFINING CLASSES	187
Class Definitions in C#	188
Interface Definitions	190
System.Object	193
Constructors and Destructors	195
Constructor Execution Sequence	196
OOP Tools in Visual Studio	200
The Class View Window	200
The Object Browser	202
Adding Classes	203
Class Diagrams	204

Class Library Projects	206
Interfaces versus Abstract Classes	209
Struct Types	212
Shallow Copying versus Deep Copying	214
CHAPTER 10: DEFINING CLASS MEMBERS	217
Member Definitions	218
Defining Fields	218
Defining Methods	219
Defining Properties	220
Refactoring Members	225
Automatic Properties	226
Additional Class Member Topics	227
Hiding Base Class Methods	227
Calling Overridden or Hidden Base Class Methods	229
Using Nested Type Definitions	230
Interface Implementation	232
Implementing Interfaces in Classes	233
Partial Class Definitions	235
Partial Method Definitions	237
Example Application	238
Planning the Application	238
Writing the Class Library	239
A Client Application for the Class Library	246
The Call Hierarchy Window	248
CHAPTER 11: COLLECTIONS, COMPARISONS, AND CONVERSIONS	251
Collections	252
Using Collections	253
Defining Collections	258
Indexers	259
Adding a Cards Collection to CardLib	262
Keyed Collections and IDictionary	264
Iterators	266
Iterators and Collections	270
Deep Copying	271
Adding Deep Copying to CardLib	273
Comparisons	275

Type Comparisons	275
Value Comparisons	279
Conversions	295
Overloading Conversion Operators	295
The as Operator	297
CHAPTER 12: GENERICS	301
<hr/>	
What Are Generics?	302
Using Generics	303
Nullable Types	303
The System.Collections.Generic Namespace	311
Defining Generic Types	321
Defining Generic Classes	322
Defining Generic Interfaces	332
Defining Generic Methods	333
Defining Generic Delegates	334
Variance	335
Covariance	336
Contravariance	336
CHAPTER 13: ADDITIONAL C# TECHNIQUES	341
<hr/>	
The :: Operator and the Global Namespace Qualifier	342
Custom Exceptions	343
Adding Custom Exceptions to CardLib	343
Events	345
What Is an Event?	345
Handling Events	347
Defining Events	350
Expanding and Using CardLib	357
Attributes	365
Reading Attributes	366
Creating Attributes	367
Initializers	368
Object Initializers	368
Collection Initializers	371
Type Inference	374
Anonymous Types	376
Dynamic Lookup	380
The dynamic Type	381

Advanced Method Parameters	384
Optional Parameters	385
Named Parameters	386
Lambda Expressions	391
Anonymous Methods Recap	391
Lambda Expressions for Anonymous Methods	392
Lambda Expression Parameters	396
Lambda Expression Statement Bodies	396
Lambda Expressions as Delegates and Expression Trees	398
Lambda Expressions and Collections	399

PART II: WINDOWS PROGRAMMING

CHAPTER 14: BASIC DESKTOP PROGRAMMING **407**

XAML	408
Separation of Concerns	409
XAML in Action	409
The Playground	411
WPF Controls	412
Properties	413
Events	417
Control Layout	422
Stack Order	423
Alignment, Margins, Padding, and Dimensions	423
Border	424
Canvas	424
DockPanel	426
StackPanel	428
WrapPanel	429
Grid	430
The Game Client	433
The About Window	433
The Options Window	439
Data Binding	448
Starting a Game with the ListBox Control	453

CHAPTER 15: ADVANCED DESKTOP PROGRAMMING **461**

The Main Window	462
The Menu Control	462
Routed Commands with Menus	462

Creating and Styling Controls	466
Styles	467
Templates	467
Value Converters	472
Triggers	473
Animations	475
WPF User Controls	478
Implementing Dependency Properties	478
Putting It All Together	489
Refactoring the Domain Model	489
The View Models	494
Completing the Game	502

PART III: CLOUD PROGRAMMING

CHAPTER 16: BASIC CLOUD PROGRAMMING 515

The Cloud, Cloud Computing, and the Cloud Optimized Stack	516
Cloud Patterns and Best Practices	519
Using Microsoft Azure C# Libraries to Create a Storage Container	520
Creating an ASP.NET 4.6 Web Site That Uses the Storage Container	530

CHAPTER 17: ADVANCED CLOUD PROGRAMMING AND DEPLOYMENT 539

Creating an ASP.NET Web API	540
Deploying and Consuming an ASP.NET Web API on Microsoft Azure	544
Scaling an ASP.NET Web API on Microsoft Azure	551

PART IV: DATA ACCESS

CHAPTER 18: FILES 561

File Classes for Input and Output	562
The File and Directory Classes	563
The FileInfo Class	564
The DirectoryInfo Class	566
Path Names and Relative Paths	566

Streams	567
Classes for Using Streams	567
The FileStream Object	568
The StreamWriter Object	575
The StreamReader Object	577
Asynchronous File Access	580
Reading and Writing Compressed Files	580
Monitoring the File System	584
CHAPTER 19: XML AND JSON	593
XML Basics	594
JSON Basics	594
XML Schemas	595
XML Document Object Model	597
The XmlDocument Class	598
The XmlElement Class	598
Changing the Values of Nodes	603
Converting XML to JSON	609
Searching XML with XPath	611
CHAPTER 20: LINQ	617
LINQ to XML	618
LINQ to XML Functional Constructors	618
Working with XML Fragments	621
LINQ Providers	624
LINQ Query Syntax	624
Declaring a Variable for Results Using the var Keyword	626
Specifying the Data Source: from Clause	627
Specify Condition: where Clause	627
Selecting Items: select Clause	627
Finishing Up: Using the foreach Loop	628
Deferred Query Execution	628
LINQ Method Syntax	628
LINQ Extension Methods	629
Query Syntax versus Method Syntax	629
Lambda Expressions	630
Ordering Query Results	632
Understanding the orderby Clause	633
Querying a Large Data Set	634
Using Aggregate Operators	636

Using the Select Distinct Query	640
Ordering by Multiple Levels	642
Using Group Queries	644
Using Joins	646
CHAPTER 21: DATABASES	651
<hr/>	
Using Databases	651
Installing SQL Server Express	652
Entity Framework	652
A Code First Database	653
But Where Is My Database?	660
Navigating Database Relationships	661
Handling Migrations	668
Creating and Querying XML from an Existing Database	669
<hr/>	
PART V: ADDITIONAL TECHNIQUES	
<hr/>	
CHAPTER 22: WINDOWS COMMUNICATION FOUNDATION	677
<hr/>	
What Is WCF?	678
WCF Concepts	679
WCF Communication Protocols	679
Addresses, Endpoints, and Bindings	680
Contracts	682
Message Patterns	683
Behaviors	683
Hosting	683
WCF Programming	684
The WCF Test Client	690
Defining WCF Service Contracts	693
Self-Hosted WCF Services	700
CHAPTER 23: UNIVERSAL APPS	709
<hr/>	
Getting Started	709
Universal Apps	710
App Concepts and Design	711
Screen Orientation	711
Menus and Toolbars	711
Tiles and Badges	712

App Lifetime	712
Lock Screen Apps	712
App Development	712
Adaptive Displays	713
Sandboxed Apps	721
Navigation between Pages	725
The CommandBar Control	728
Managing State	729
Common Elements of Windows Store Apps	732
The Windows Store	733
Packaging an App	733
Creating the Package	734
APPENDIX: EXERCISE SOLUTIONS	737
<hr/>	
INDEX	781

INTRODUCTION

C# IS A RELATIVELY NEW LANGUAGE that was unveiled to the world when Microsoft announced the first version of its .NET Framework in July 2000. Since then its popularity has rocketed, and it has arguably become the language of choice for desktop, web, and cloud developers who use the .NET Framework. Part of the appeal of C# comes from its clear syntax, which derives from C/C++ but simplifies some things that have previously discouraged some programmers. Despite this simplification, C# has retained the power of C++, and there is now no reason not to move into C#. The language is not difficult and it's a great one to learn elementary programming techniques with. This ease of learning, combined with the capabilities of the .NET Framework, make C# an excellent way to start your programming career.

The latest release of C#, C# 6, which is included with version 4.6 of the .NET Framework, builds on the existing successes and adds even more attractive features. The latest release of Visual Studio (Visual Studio 2015) and the Visual Studio Express/Community 2015 line of development tools also bring many tweaks and improvements to make your life easier and to dramatically increase your productivity.

This book is intended to teach you about all aspects of C# programming, including the language itself, desktop and cloud programming, making use of data sources, and some new and advanced techniques. You'll also learn about the capabilities of Visual Studio 2015 and all the ways that this product can aid your application development.

The book is written in a friendly, mentor-style fashion, with each chapter building on previous ones, and every effort is made to ease you into advanced techniques painlessly. At no point will technical terms appear from nowhere to discourage you from continuing; every concept is introduced and discussed as required. Technical jargon is kept to a minimum; but where it is necessary, it, too, is properly defined and laid out in context.

The authors of this book are all experts in their field and are all enthusiastic in their passion for both the C# language and the .NET Framework. Nowhere will you find a group of people better qualified to take you under their collective wing and nurture your understanding of C# from first principles to advanced techniques. Along with the fundamental knowledge it provides, this book is packed full of helpful hints, tips, exercises, and full-fledged example code (available for download at p2p.wrox.com) that you will find yourself returning to repeatedly as your career progresses.

We pass this knowledge on without begrudging it and hope that you will be able to use it to become the best programmer you can be. Good luck, and all the best!

WHO THIS BOOK IS FOR

This book is for everyone who wants to learn how to program in C# using the .NET Framework. It is for absolute beginners who want to give programming a try by learning a clean, modern, elegant programming language. But it is also for people familiar with other programming languages who want to explore the .NET platform, as well as for existing .NET developers who want to give Microsoft's .NET flagship language a try.

WHAT THIS BOOK COVERS

The early chapters cover the language itself, assuming no prior programming experience. If you have programmed in other languages before, much of the material in these chapters will be familiar. Many aspects of C# syntax are shared with other languages, and many structures are common to practically all programming languages (such as looping and branching structures). However, even if you are an experienced programmer, you will benefit from looking through these chapters to learn the specifics of how these techniques apply to C#.

If you are new to programming, you should start from the beginning, where you will learn basic programming concepts and become acquainted with both C# and the .NET platform that underpins it. If you are new to the .NET Framework but know how to program, you should read Chapter 1 and then skim through the next few chapters before continuing with the application of the C# language. If you know how to program but haven't encountered an object-oriented programming language before, you should read the chapters from Chapter 8 onward.

Alternatively, if you already know the C# language, you might want to concentrate on the chapters dealing with the most recent .NET Framework and C# language developments, specifically the chapters on collections, generics, and C# language enhancements (Chapters 11 to 13), or skip the first section of the book completely and start with Chapter 14.

The chapters in this book have been written with a dual purpose in mind: They can be read sequentially to provide a complete tutorial in the C# language, and they can be dipped into as required reference material.

In addition to the core material, starting with Chapter 3 each chapter also includes a selection of exercises at the end, which you can work through to ensure that you have understood the material. The exercises range from simple multiple choice or true/false questions to more complex exercises that require you to modify or build applications. The answers to all the exercises are provided in Appendix A. You can also find these exercises as part of the wrox.com code downloads for this book at www.wrox.com/go/beginningvisualcsharp2015programming.

This book also gives plenty of love and attention to coincide with the release of C# 6 and .NET 4.6. Every chapter received an overhaul, with less relevant material removed, and new material added. All of the code has been tested against the latest version of the development tools used, and all of the screenshots have been retaken in Windows 8.1/10 to provide the most current windows and dialog boxes.

New highlights of this edition include the following:

- Additional and improved code examples for you to try out
- Coverage of everything that's new in C# 6 and .NET 4.6, including how to create Universal Windows Apps
- Examples of programming cloud applications and using Azure SDK to create and access cloud resources

HOW THIS BOOK IS STRUCTURED

This book is divided into six sections:

- **Introduction** — Purpose and general outline of the book's contents
- **The C# Language** — Covers all aspects of the C# language, from the fundamentals to object-oriented techniques
- **Windows Programming** — How to write and deploy desktop applications with the Windows Presentation Foundation library (WPF)
- **Cloud Programming** — Cloud application development and deployment, including the creation and consumption of a Web API
- **Data Access** — How to use data in your applications, including data stored in files on your hard disk, data stored in XML format, and data in databases
- **Additional Techniques** — An examination of some extra ways to use C# and the .NET Framework, including Windows Communication Foundation (WCF) and Universal Windows Applications

The following sections describe the chapters in the five major parts of this book.

The C# Language (Chapters 1–13)

Chapter 1 introduces you to C# and how it fits into the .NET landscape. You'll learn the fundamentals of programming in this environment and how Visual Studio 2015 (VS) fits in.

Chapter 2 starts you off with writing C# applications. You'll look at the syntax of C# and put the language to use with sample command-line and Windows applications. These examples demonstrate just how quick and easy it can be to get up and running, and along the way you'll be introduced to the Visual Studio development environment and the basic windows and tools that you'll be using throughout the book.

Next you'll learn more about the basics of the C# language. You'll learn what variables are and how to manipulate them in Chapter 3. You'll enhance the structure of your applications with flow control (looping and branching) in Chapter 4, and you'll see some more advanced variable types

such as arrays in Chapter 5. In Chapter 6 you'll start to encapsulate your code in the form of functions, which makes it much easier to perform repetitive operations and makes your code much more readable.

By the beginning of Chapter 7 you'll have a handle on the fundamentals of the C# language, and you will focus on debugging your applications. This involves looking at outputting trace information as your applications are executed, and at how Visual Studio can be used to trap errors and lead you to solutions for them with its powerful debugging environment.

From Chapter 8 onward you'll learn about object-oriented programming (OOP), starting with a look at what this term means and an answer to the eternal question, "What is an object?" OOP can seem quite difficult at first. The whole of Chapter 8 is devoted to demystifying it and explaining what makes it so great, and you won't actually deal with much C# code until the very end of the chapter.

Everything changes in Chapter 9, when you put theory into practice and start using OOP in your C# applications. This is where the true power of C# lies. You'll start by looking at how to define classes and interfaces, and then move on to class members (including fields, properties, and methods) in Chapter 10. At the end of that chapter you'll start to assemble a card game application, which is developed over several chapters and will help to illustrate OOP.

Once you've learned how OOP works in C#, Chapter 11 moves on to look at common OOP scenarios, including dealing with collections of objects, and comparing and converting objects. Chapter 12 takes a look at a very useful feature of C# that was introduced in .NET 2.0: generics, which enable you to create very flexible classes. Next, Chapter 13 continues the discussion of the C# language and OOP with some additional techniques, notably events, which become very important in, for example, Windows programming. Chapter 13 wraps up the fundamentals by focusing on C# language features that were introduced with versions 3.0, 4, 5, and 6 of the language.

Windows Programming (Chapters 14–15)

Chapter 14 starts by introducing you to what is meant by Windows programming and looks at how this is achieved in Visual Studio. It focuses on WPF as a tool that enables you to build desktop applications in a graphical way and assemble advanced applications with the minimum of effort and time. You'll start with the basics of WPF programming and build up your knowledge in both this chapter and Chapter 15, which demonstrates how you can use the wealth of controls supplied by the .NET Framework in your applications.

Cloud Programming (Chapters 16–17)

Chapter 16 starts by describing what cloud programming is and discusses the cloud optimized stack. The cloud environment is not identical to the way programs have been traditionally coded, so a few cloud programming patterns are discussed and defined. To complete this chapter, you require an Azure account, which is free, so that you can create an App Services Web App, then using the Azure SDK with C#, you create and access a storage account from an ASP.NET 4.6 web application.

In Chapter 17, you learn how to create and deploy an ASP.NET Web API to the cloud and then consume the Web API from a similar ASP.NET 4.6 web application. The chapter ends discussing two of the most valuable features in the cloud, scaling and the optimal utilization of hardware resources.

Data Access (Chapters 18–21)

Chapter 18 looks at how your applications can save and retrieve data to disk, both as simple text files and as more complex representations of data. You'll also learn how to compress data and how to monitor and act on file system changes.

In Chapter 19 you'll learn about the de facto standard for data exchange — namely, XML — and a rapidly emerging format called JSON. By this point in the book, you'll have touched on XML several times in preceding chapters, but this chapter lays out the ground rules and shows you what all the excitement is about.

The remainder of this part looks at LINQ, which is a query language built in to the latest versions of the .NET Framework. You start in Chapter 20 with a general introduction to LINQ, and then you will use LINQ to access a database and other data in Chapter 21.

Additional Techniques (Chapters 22–23)

Chapter 22 is an introduction to Windows Communication Foundation (WCF), which provides you with the tools you need for enterprise-level programmatic access to information and capabilities across local networks and the Internet. You will see how you can use WCF to expose complex data and functionality to web and desktop applications in a platform-independent way.

Chapter 23 shows you how you can create Universal Windows Apps, which are new to Windows. This chapter builds on the foundation of Chapters 14 and 15 to show you how to create Windows Apps that can run on all windows platforms.

WHAT YOU NEED TO USE THIS BOOK

The code and descriptions of C# and the .NET Framework in this book apply to C# 6 and .NET 4.6. You don't need anything other than the Framework to understand this aspect of the book, but many of the examples require a development tool. This book uses Visual Studio 2015 as its primary development tool; however, if you don't have this, you will be able to use the free Visual Studio Express/Community 2015 line of products. For the first part of the book, Visual Studio Express/Community 2012 for Windows Desktop will enable you to create desktop and console applications. For later chapters, you may also use Visual Studio Express/Community 2015 for Windows 10 in order to create Universal Windows Apps, Visual Studio Express/Community 2015 for Cloud to create cloud applications, and SQL Server Express 2014 for applications that access databases. Some functionality is available only in Visual Studio 2015, but this won't stop you from working through any of the examples in this book.

The source code for the samples is available for download from the Wrox website at:

`www.wrox.com/go/beginningvisualc#2015programming`

CONVENTIONS

To help you get the most from the text and keep track of what's happening, we've used a number of conventions throughout the book.

TRY IT OUT

The *Try It Out* is an exercise you should work through, following the text in the book.

1. They usually consist of a set of steps.
2. Each step has a number.
3. Follow the steps through with your copy of the database.

How It Works

After each *Try It Out*, the code you've typed will be explained in detail.

WARNING Warnings hold important, not-to-be-forgotten information that is directly relevant to the surrounding text.

NOTE Notes indicates notes, tips, hints, tricks, or and asides to the current discussion.

As for styles in the text:

- We *highlight* new terms and important words when we introduce them.
- We show keyboard strokes like this: Ctrl+A.
- We show file names, URLs, and code within the text like so: `persistence.properties`.

We present code in two different ways:

We use a monospace type with no highlighting for most code examples.

We use bold to emphasize code that is particularly important in the present context or to show changes from a previous code snippet.

SOURCE CODE

As you work through the examples in this book, you may choose either to type in all the code manually, or to use the source code files that accompany the book. All the source code used in this book is available for download at www.wrox.com. Specifically for this book, the code download is on the Download Code tab at:

www.wrox.com/go/beginningvisualc#2015programming

You can also search for the book at www.wrox.com by ISBN (the ISBN for this book is 978-1-119-09668-9) to find the code. And a complete list of code downloads for all current Wrox books is available at www.wrox.com/dynamic/books/download.aspx.

Most of the code on www.wrox.com is compressed in a .ZIP, .RAR archive or similar archive format appropriate to the platform. Once you download the code, just decompress it with an appropriate compression tool.

NOTE *Because many books have similar titles, you may find it easiest to search by ISBN; this book's ISBN is 978-1-119-09668-9.*

Alternately, you can go to the main Wrox code download page at www.wrox.com/dynamic/books/download.aspx to see the code available for this book and all other Wrox books.

ERRATA

We make every effort to ensure that there are no errors in the text or in the code. However, no one is perfect, and mistakes do occur. If you find an error in one of our books, like a spelling mistake or faulty piece of code, we would be very grateful for your feedback. By sending in errata, you may save another reader hours of frustration, and at the same time, you will be helping us provide even higher quality information.

To find the errata page for this book, go to

www.wrox.com/go/beginningvisualc#2015programming

And click the Errata link. On this page you can view all errata that has been submitted for this book and posted by Wrox editors.

If you don't spot "your" error on the Book Errata page, go to www.wrox.com/contact/techsupport.shtml and complete the form there to send us the error you have found. We'll check the information and, if appropriate, post a message to the book's errata page and fix the problem in subsequent editions of the book.

P2P.WROX.COM

For author and peer discussion, join the P2P forums at <http://p2p.wrox.com>. The forums are a Web-based system for you to post messages relating to Wrox books and related technologies and interact with other readers and technology users. The forums offer a subscription feature to e-mail you topics of interest of your choosing when new posts are made to the forums. Wrox authors, editors, other industry experts, and your fellow readers are present on these forums.

At <http://p2p.wrox.com>, you will find a number of different forums that will help you, not only as you read this book, but also as you develop your own applications. To join the forums, just follow these steps:

1. Go to <http://p2p.wrox.com> and click the Register link.
2. Read the terms of use and click Agree.
3. Complete the required information to join, as well as any optional information you wish to provide, and click Submit.
4. You will receive an e-mail with information describing how to verify your account and complete the joining process.

NOTE *You can read messages in the forums without joining P2P, but in order to post your own messages, you must join.*

Once you join, you can post new messages and respond to messages other users post. You can read messages at any time on the Web. If you would like to have new messages from a particular forum e-mailed to you, click the Subscribe to this Forum icon by the forum name in the forum listing.

For more information about how to use the Wrox P2P, be sure to read the P2P FAQs for answers to questions about how the forum software works, as well as many common questions specific to P2P and Wrox books. To read the FAQs, click the FAQ link on any P2P page.

PART I

The OOP Language

- ▶ CHAPTER 1: Introducing C#
- ▶ CHAPTER 2: Writing a C# Program
- ▶ CHAPTER 3: Variables and Expressions
- ▶ CHAPTER 4: Flow Control
- ▶ CHAPTER 5: More about Variables
- ▶ CHAPTER 6: Functions
- ▶ CHAPTER 7: Debugging and Error Handling
- ▶ CHAPTER 8: Introduction to Object-Oriented Programming
- ▶ CHAPTER 9: Defining Classes
- ▶ CHAPTER 10: Defining Class Members
- ▶ CHAPTER 11: Collections, Comparisons, and Conversions
- ▶ CHAPTER 12: Generics
- ▶ CHAPTER 13: Additional C# Techniques

