



Wrox Programmer to Programmer™



Beginning

C# and .NET

2021 Edition

Benjamin Perkins and Jon D. Reid

Table of Contents

[COVER](#)

[TITLE PAGE](#)

[INTRODUCTION](#)

[WHO THIS BOOK IS FOR](#)

[WHAT THIS BOOK COVERS](#)

[HOW THIS BOOK IS STRUCTURED](#)

[WHAT YOU NEED TO USE THIS BOOK](#)

[CONVENTIONS](#)

[SOURCE CODE](#)

[ERRATA](#)

[PART I: The C# Language](#)

[1 Introducing C#](#)

[WHAT IS .NET?](#)

[WHAT IS C#?](#)

[VISUAL STUDIO](#)

[► WHAT YOU LEARNED IN THIS CHAPTER](#)

[2 Writing a C# Program](#)

[THE VISUAL STUDIO DEVELOPMENT
ENVIRONMENT](#)

[CONSOLE APPLICATIONS](#)

[DESKTOP APPLICATIONS](#)

[► WHAT YOU LEARNED IN THIS CHAPTER](#)

[3 Variables and Expressions](#)

[BASIC C# SYNTAX](#)

[BASIC C# CONSOLE APPLICATION
STRUCTURE](#)

VARIABLES

EXPRESSIONS

EXERCISES

► WHAT YOU LEARNED IN THIS CHAPTER

4 Flow Control

BOOLEAN LOGIC

BRANCHING

LOOPING

► WHAT YOU LEARNED IN THIS CHAPTER

5 More about Variables

TYPE CONVERSION

COMPLEX VARIABLE TYPES

STRING MANIPULATION

► WHAT YOU LEARNED IN THIS CHAPTER

6 Functions

DEFINING AND USING FUNCTIONS

VARIABLE SCOPE

THE MAIN() FUNCTION

STRUCT FUNCTIONS

OVERLOADING FUNCTIONS

USING DELEGATES

► WHAT YOU LEARNED IN THIS CHAPTER

7 Debugging and Error Handling

DEBUGGING IN VISUAL STUDIO

ERROR HANDLING

► WHAT YOU LEARNED IN THIS CHAPTER

8 Introduction to Object-Oriented Programming

WHAT IS OBJECT-ORIENTED
PROGRAMMING?

OOP TECHNIQUES

OOP IN DESKTOP APPLICATIONS

► WHAT YOU LEARNED IN THIS CHAPTER

9 Defining Classes

CLASS DEFINITIONS IN C#

SYSTEM.OBJECT

CONSTRUCTORS AND DESTRUCTORS

OOP TOOLS IN VISUAL STUDIO

CLASS LIBRARY PROJECTS

INTERFACES VERSUS ABSTRACT CLASSES

STRUCT TYPES

SHALLOW COPYING VERSUS DEEP COPYING

► WHAT YOU LEARNED IN THIS CHAPTER

10 Defining Class Members

MEMBER DEFINITIONS

ADDITIONAL CLASS MEMBER TOPICS

INTERFACE IMPLEMENTATION

PARTIAL CLASS DEFINITIONS

PARTIAL METHOD DEFINITIONS

EXAMPLE APPLICATION

THE CALL HIERARCHY WINDOW

► WHAT YOU LEARNED IN THIS CHAPTER

11 Collections, Comparisons, and Conversions

COLLECTIONS

COMPARISONS

CONVERSIONS

► WHAT YOU LEARNED IN THIS CHAPTER

12 Generics

WHAT ARE GENERICS?

USING GENERICS

DEFINING GENERIC TYPES

VARIANCE

EXERCISES

► WHAT YOU HAVE LEARNED IN THIS CHAPTER

13 Additional C# Techniques

THE :: OPERATOR AND THE GLOBAL NAMESPACE QUALIFIER

CUSTOM EXCEPTIONS

EVENTS

EXPANDING AND USING CARDLIB

ATTRIBUTES

INITIALIZERS

TYPE INFERENCE

ANONYMOUS TYPES

DYNAMIC LOOKUP

ADVANCED METHOD PARAMETERS

LAMBDA EXPRESSIONS

► WHAT YOU LEARNED IN THIS CHAPTER

PART II: Data Access

14 Files

FILE CLASSES FOR INPUT AND OUTPUT

STREAMS

MONITORING THE FILE SYSTEM

► WHAT YOU LEARNED IN THIS CHAPTER

15 XML and JSON

XML BASICS

JSON BASICS

XML SCHEMAS

XML DOCUMENT OBJECT MODEL

SEARCHING XML WITH XPATH

JSON SERIALIZATION AND
DESERIALIZATION

► WHAT YOU LEARNED IN THIS CHAPTER

16 LINQ

LINQ TO XML

LINQ PROVIDERS

LINQ QUERY SYNTAX

LINQ METHOD SYNTAX

ORDERING QUERY RESULTS

UNDERSTANDING THE ORDERBY CLAUSE

QUERYING A LARGE DATA SET

USING AGGREGATE OPERATORS

USING THE SELECT DISTINCT QUERY

ORDERING BY MULTIPLE LEVELS

USING GROUP QUERIES

USING JOINS

EXERCISES

► WHAT YOU LEARNED IN THIS CHAPTER

17 Databases

USING DATABASES

ENTITY FRAMEWORK

CODE-FIRST VERSUS DATABASE-FIRST

MIGRATIONS AND SCAFFOLDING
INSTALL SQL SERVER EXPRESS LOCALDB
A CODE-FIRST DATABASE
EXPLORING YOUR DATABASE
NAVIGATING DATABASE RELATIONSHIPS
CREATING AND QUERYING XML FROM AN
EXISTING DATABASE
► WHAT YOU LEARNED IN THIS CHAPTER

PART III: Additional Techniques

18 .NET and ASP.NET

CROSS-PLATFORM BASICS AND KEY “MUST
KNOW” TERMS
WHAT WAS .NET STANDARD?
REFERENCING AND TARGETING .NET
WHAT WAS .NET CORE?
PORTING FROM .NET FRAMEWORK TO .NET
OVERVIEW OF WEB APPLICATIONS
WHICH ASP.NET TO USE AND WHY
► WHAT YOU LEARNED IN THIS CHAPTER

19 Basic Cloud Programming

THE CLOUD, CLOUD COMPUTING, AND THE
CLOUD OPTIMIZED STACK
CLOUD PATTERNS AND BEST PRACTICES
USING MICROSOFT AZURE C# LIBRARIES TO
CREATE A STORAGE CONTAINER
CREATING AN ASP.NET CORE WEB
APPLICATION THAT USES THE STORAGE
CONTAINER
EXERCISES
► WHAT YOU LEARNED IN THIS CHAPTER

[20 Basic Web API and WCF Programming](#)
[CREATING AN ASP.NET CORE WEB API](#)
[CONSUMING AN ASP.NET CORE WEB API](#)
[WHAT IS REST?](#)
[WHAT IS WCF?](#)
[WCF CONCEPTS](#)
[WCF PROGRAMMING](#)
[► WHAT YOU LEARNED IN THIS CHAPTER](#)

[21 Basic Desktop Programming](#)
[XAML](#)
[THE PLAYGROUND](#)
[CONTROL LAYOUT](#)
[THE GAME CLIENT](#)
[CREATING AND STYLING CONTROLS](#)
[WPF USER CONTROLS](#)
[THE MAIN WINDOW](#)
[PUTTING IT ALL TOGETHER](#)
[► WHAT YOU LEARNED IN THIS CHAPTER](#)

[APPENDIX: Exercise Solutions](#)
[INDEX](#)
[COPYRIGHT](#)
[ABOUT THE AUTHORS](#)
[ABOUT THE TECHNICAL EDITOR](#)
[ACKNOWLEDGMENTS](#)
[END USER LICENSE AGREEMENT](#)

List of Tables

Chapter 3

[TABLE 3-1: Integer Types](#)

[TABLE 3-2: Floating-Point Types](#)

[TABLE 3-3: Text and Boolean Types](#)

[TABLE 3-4: Literal Values](#)

[TABLE 3-5: Escape Sequences for String Literals](#)

[TABLE 3-6: Simple Mathematical Operators](#)

[TABLE 3-7: The String Concatenation Operator](#)

[TABLE 3-8: Increment and Decrement Operators](#)

[TABLE 3-9: Assignment Operators](#)

[TABLE 3-10: Operator Precedence](#)

Chapter 4

[TABLE 4-1: Boolean Comparison Operators](#)

[TABLE 4-2: Conditional Boolean Operators](#)

[TABLE 4-3: Boolean Assignment Operators](#)

[TABLE 4-4: Operator Precedence \(Updated\)](#)

Chapter 5

[TABLE 5-1: Implicit Numeric Conversions](#)

Chapter 9

[TABLE 9-1: Access Modifiers for Class Definitions](#)

[TABLE 9-2: Methods of `System.Object`](#)

[TABLE 9-3: Class View Icons](#)

[TABLE 9-4: Additional Class View Icons](#)

Chapter 12

[TABLE 12-1: Generic Collection Type](#)

[TABLE 12-2: Sorting with Generic Types](#)

[TABLE 12-3: Generic Type Constraints](#)

Chapter 14

[TABLE 14-1: File System Access Classes](#)

[TABLE 14-2: Static Methods of the File Class](#)

[TABLE 14-3: Static Methods of the Directory Class](#)

[TABLE 14-4: FileSystemInfo Properties](#)

[TABLE 14-5: FileInfo Properties](#)

[TABLE 14-6: Properties Unique to the DirectoryInfo Class](#)

[TABLE 14-7: Stream Classes](#)

[TABLE 14-8: FileAccess Enumeration Members](#)

[TABLE 14-9: FileMode Enumeration Members](#)

[TABLE 14-10: FileSystemWatcher Properties](#)

Chapter 15

[TABLE 15-1: Common DOM Classes](#)

[TABLE 15-2: XmlElement Properties](#)

[TABLE 15-3: Three Ways to Get the Value of a Node](#)

[TABLE 15-4: Methods for Creating Nodes](#)

[TABLE 15-5: Methods for Inserting Nodes](#)

[TABLE 15-6: Methods for Removing Nodes](#)

[TABLE 15-7: Methods for Selecting Nodes](#)

[TABLE 15-8: Common XPath Operations](#)

Chapter 16

[TABLE 16-1: Aggregate Operators for Numeric Results](#)

Chapter 18

[TABLE 18-1: Cross-Platform Key Terms](#)

[TABLE 18-2: Summary of .NET Standard Supported Version](#)

[TABLE 18-3: .NET Core Performance Improvements versus .NET Framework](#)

[TABLE 18-4: Examples of ASP.NET Server Controls](#)

[TABLE 18-5: Examples of ASP.NET Validation Controls](#)

[TABLE 18-6: ASP.NET Web Forms State Management Techniques](#)

[TABLE 18-7: Differences between Projects and Web Sites](#)

[TABLE 18-8: Baseline ASP.NET Core on Kestrel performance](#)

[TABLE 18-9: Examples of ASP.NET Core Validation Attributes](#)

[TABLE 18-10: ASP.NET Core State Management Techniques](#)

Chapter 20

[TABLE 20-1: Binding Types](#)

[TABLE 20-2: NET Default Bindings](#)

[TABLE 20-3: DataContractAttribute Properties](#)

[TABLE 20-4: DataMemberAttribute Properties](#)

[TABLE 20-5: ServiceContractAttribute Properties](#)

[TABLE 20-6: OperationContractAttribute Properties](#)

Chapter 21

[TABLE 21-1: Common Control Events](#)

[TABLE 21-2: Common Layout Panels](#)

[TABLE 21-3: Image Control Properties](#)

[TABLE 21-4: TextBox Properties](#)

[TABLE 21-5: CheckBox Properties](#)

[TABLE 21-6: RadioButton Properties](#)

[TABLE 21-7: IsReadOnly and IsEditable Combinations](#)

[TABLE 21-8: Other ComboBox Properties](#)

[TABLE 21-9: Interesting ListBox Properties](#)

[TABLE 21-10: The Name Property](#)

[TABLE 21-11: The Register\(.\) Method's Parameters](#)

[TABLE 21-12: Overloads for the FrameworkPropertyMetadata Constructor](#)

[TABLE 21-13: Displaying MenuItem Properties](#)

[TABLE 21-14: Property and Field Names](#)

[TABLE 21-15: Cards in Hand Dependency Properties](#)

[TABLE 21-16: Game Decks Dependency Properties](#)

List of Illustrations

Chapter 1

[FIGURE 1-1](#)

[FIGURE 1-2](#)

[FIGURE 1-3](#)

[FIGURE 1-4](#)

Chapter 2

[FIGURE 2-1](#)

[FIGURE 2-2](#)

[FIGURE 2-3](#)

[FIGURE 2-4](#)

[FIGURE 2-5](#)

[FIGURE 2-6](#)

[FIGURE 2-7](#)

[FIGURE 2-8](#)

[FIGURE 2-9](#)

[FIGURE 2-10](#)

[FIGURE 2-11](#)

[FIGURE 2-12](#)

[FIGURE 2-13](#)

[FIGURE 2-14](#)

Chapter 3

[FIGURE 3-1](#)

[FIGURE 3-2](#)

[FIGURE 3-3](#)

[FIGURE 3-4](#)

Chapter 4

[FIGURE 4-1](#)

[FIGURE 4-2](#)

[FIGURE 4-3](#)

[FIGURE 4-4](#)

[FIGURE 4-5](#)

[FIGURE 4-6](#)

[FIGURE 4-7](#)

Chapter 5

[FIGURE 5-1](#)

[FIGURE 5-2](#)

[FIGURE 5-3](#)

[FIGURE 5-4](#)

[FIGURE 5-5](#)

[FIGURE 5-6](#)

[FIGURE 5-7](#)

[FIGURE 5-8](#)

[FIGURE 5-9](#)

[FIGURE 5-10](#)

[FIGURE 5-11](#)

[FIGURE 5-12](#)

[FIGURE 5-13](#)

[FIGURE 5-14](#)

[FIGURE 5-15](#)

Chapter 6

[FIGURE 6-1](#)

[FIGURE 6-2](#)

[FIGURE 6-3](#)

[FIGURE 6-4](#)

[FIGURE 6-5](#)

[FIGURE 6-6](#)

[FIGURE 6-7](#)

[FIGURE 6-8](#)

[FIGURE 6-9](#)

[FIGURE 6-10](#)

[FIGURE 6-11](#)

[FIGURE 6-12](#)

Chapter 7

[FIGURE 7-1](#)

[FIGURE 7-2](#)

[FIGURE 7-3](#)

[FIGURE 7-4](#)

[FIGURE 7-5](#)

[FIGURE 7-6](#)

[FIGURE 7-7](#)

[FIGURE 7-8](#)

[FIGURE 7-9](#)

[FIGURE 7-10](#)

[FIGURE 7-11](#)

[FIGURE 7-12](#)

[FIGURE 7-13](#)

[FIGURE 7-14](#)

[FIGURE 7-15](#)

[FIGURE 7-16](#)

[FIGURE 7-17](#)

[FIGURE 7-18](#)

[FIGURE 7-19](#)

[FIGURE 7-20](#)

Chapter 8

[FIGURE 8-1](#)

[FIGURE 8-2](#)

[FIGURE 8-3](#)

[FIGURE 8-4](#)

[FIGURE 8-5](#)

[FIGURE 8-6](#)

[FIGURE 8-7](#)

[FIGURE 8-8](#)

[FIGURE 8-9](#)

[FIGURE 8-10](#)

[FIGURE 8-11](#)

[FIGURE 8-12](#)

[FIGURE 8-13](#)

[FIGURE 8-14](#)

[FIGURE 8-15](#)

Chapter 9

[FIGURE 9-1](#)

[FIGURE 9-2](#)

[FIGURE 9-3](#)

[FIGURE 9-4](#)

[FIGURE 9-5](#)

[FIGURE 9-6](#)

[FIGURE 9-7](#)

[FIGURE 9-8](#)

[FIGURE 9-9](#)

[FIGURE 9-10](#)

[FIGURE 9-11](#)

[FIGURE 9-12](#)

[FIGURE 9-13](#)

[FIGURE 9-14](#)

Chapter 10

[FIGURE 10-1](#)

[FIGURE 10-2](#)

[FIGURE 10-3](#)

[FIGURE 10-4](#)

[FIGURE 10-5](#)

[FIGURE 10-6](#)

[FIGURE 10-7](#)

[FIGURE 10-8](#)

[FIGURE 10-9](#)

[FIGURE 10-10](#)

[FIGURE 10-11](#)

[FIGURE 10-12](#)

[FIGURE 10-13](#)

[FIGURE 10-14](#)

Chapter 11

[FIGURE 11-1](#)

[FIGURE 11-2](#)

[FIGURE 11-3](#)

[FIGURE 11-4](#)

[FIGURE 11-5](#)

[FIGURE 11-6](#)

[FIGURE 11-7](#)

[FIGURE 11-8](#)

Chapter 12

[FIGURE 12-1](#)

[FIGURE 12-2](#)

[FIGURE 12-3](#)

[FIGURE 12-4](#)

[FIGURE 12-5](#)

Chapter 13

[FIGURE 13-1](#)

[FIGURE 13-2](#)

[FIGURE 13-3](#)

[FIGURE 13-4](#)

[FIGURE 13-5](#)

[FIGURE 13-6](#)

[FIGURE 13-7](#)

[FIGURE 13-8](#)

[FIGURE 13-9](#)

[FIGURE 13-10](#)

[FIGURE 13-11](#)

[FIGURE 13-12](#)

[FIGURE 13-13](#)

[FIGURE 13-14](#)

[FIGURE 13-15](#)

[FIGURE 13-16](#)

[FIGURE 13-17](#)

[FIGURE 13-18](#)

[FIGURE 13-19](#)

[FIGURE 13-20](#)

[FIGURE 13-21](#)

Chapter 14

[FIGURE 14-1](#)

[FIGURE 14-2](#)

[FIGURE 14-3](#)

[FIGURE 14-4](#)

[FIGURE 14-5](#)

[FIGURE 14-6](#)

[FIGURE 14-7](#)

[FIGURE 14-8](#)

Chapter 15

[FIGURE 15-1](#)

[FIGURE 15-2](#)

[FIGURE 15-3](#)

[FIGURE 15-4](#)

[FIGURE 15-5](#)

[FIGURE 15-6](#)

[FIGURE 15-7](#)

Chapter 17

[FIGURE 17-1](#)

[FIGURE 17-2](#)

[FIGURE 17-3](#)

[FIGURE 17-4](#)

[FIGURE 17-5](#)

[FIGURE 17-6](#)

[FIGURE 17-7](#)

[FIGURE 17-8](#)

[FIGURE 17-9](#)

[FIGURE 17-10](#)

[FIGURE 17-11](#)

[FIGURE 17-12](#)

[FIGURE 17-13](#)

[FIGURE 17-14](#)

[FIGURE 17-15](#)

[FIGURE 17-16](#)

[FIGURE 17-17](#)

[FIGURE 17-18](#)

[FIGURE 17-19](#)

Chapter 18

[FIGURE 18-1](#)

[FIGURE 18-2](#)

[FIGURE 18-3](#)

[FIGURE 18-4](#)

[FIGURE 18-5](#)

[FIGURE 18-6](#)

[FIGURE 18-7](#)

[FIGURE 18-8](#)

[FIGURE 18-9](#)

[FIGURE 18-10](#)

[FIGURE 18-11](#)

[FIGURE 18-12](#)

[FIGURE 18-13](#)

[FIGURE 18-14](#)

[FIGURE 18-15](#)

[FIGURE 18-16](#)

Chapter 19

[FIGURE 19-1](#)

[FIGURE 19-2](#)

[FIGURE 19-3](#)

[FIGURE 19-4](#)

[FIGURE 19-5](#)

[FIGURE 19-6](#)

[FIGURE 19-7](#)

[FIGURE 19-8](#)

[FIGURE 19-9](#)

[FIGURE 19-10](#)

[FIGURE 19-11](#)

[FIGURE 19-12](#)

[FIGURE 19-13](#)

[FIGURE 19-14](#)

Chapter 20

[FIGURE 20-1](#)

[FIGURE 20-2](#)

[FIGURE 20-3](#)

[FIGURE 20-4](#)

[FIGURE 20-5](#)

[FIGURE 20-6](#)

[FIGURE 20-7](#)

[FIGURE 20-8](#)

[FIGURE 20-9](#)

[FIGURE 20-10](#)

[FIGURE 20-11](#)

[FIGURE 20-12](#)

[FIGURE 20-13](#)

[FIGURE 20-14](#)

[FIGURE 20-15](#)

[FIGURE 20-16](#)

[FIGURE 20-17](#)

[FIGURE 20-18](#)

Chapter 21

[FIGURE 21-1](#)

[FIGURE 21-2](#)

[FIGURE 21-3](#)

[FIGURE 21-4](#)

[FIGURE 21-5](#)

[FIGURE 21-6](#)

[FIGURE 21-7](#)

[FIGURE 21-8](#)

[FIGURE 21-9](#)

[FIGURE 21-10](#)

[FIGURE 21-11](#)

[FIGURE 21-12](#)

[FIGURE 21-13](#)

[FIGURE 21-14](#)

[FIGURE 21-15](#)

[FIGURE 21-16](#)

[FIGURE 21-17](#)

[FIGURE 21-18](#)

BEGINNING C# and .NET

2021 EDITION

Benjamin Perkins

Jon D. Reid



INTRODUCTION

THE C# LANGUAGE WAS UNVEILED TO THE WORLD when Microsoft announced the first version of its .NET Framework in 2002. Since then, its popularity has rocketed, and it has arguably become the language of choice for desktop, web, cloud, and cross-platform developers who use .NET. 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 no reason now not to move into C#. The language is not difficult, and it is a great one with which to learn elementary programming techniques. 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# is C# 9 (included with .NET 5.0 and .NET Framework 4.8), which builds on the existing successes and adds even more attractive features. The latest releases of both Visual Studio and Visual Studio Code 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, cloud, and cross-platform programming, making use of data sources, and some new and advanced techniques. You will also learn about the capabilities of Visual Studio 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 both experts in their field and are enthusiastic in their passion for the C# language and .NET. Nowhere will you find two 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 on this book's web page at www.wiley.com and at github.com/benperk/BeginningCSharpAndDotNET) 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 .NET. 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 by 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 .NET 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 have not 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 and C# language developments, specifically the chapters on collections, generics, and C# language enhancements ([Chapters 11](#) and [12](#)).

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 for reference material.

In addition to the core material, starting with [Chapter 3](#) most chapters also include 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 the Appendix. You can also find these exercises as part of the wiley.com code downloads on this book's page at www.wiley.com.

Every chapter receives an overhaul with every new release of C# and .NET, the less relevant material is removed, and new material added. All the code has been tested against the latest version of the development tools used, and all the screenshots have been retaken in the most current version of the Windows OS 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
- Examples of programming ASP.NET Core for running cross-platform
- Examples of programming cloud applications, using Azure SDK to create and access cloud resources

HOW THIS BOOK IS STRUCTURED

This book is divided into four 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
- **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 .NET, including cloud and cross-platform development, ASP.NET Web API, Windows Presentation Foundation (WPF), Windows Communication Foundation (WCF), and Universal Windows Applications

The following sections describe the chapters in the three 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 will learn the fundamentals of programming in this environment and how Visual Studio (VS) fits in.

[Chapter 2](#) starts you off with writing C# applications. You will 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 will be introduced to the Visual Studio development environment