STARTING OUT WITH

Visual C#®

Fifth Edition

Tony Gaddis
Haywood Community College

Pearson

A01_GADD3519_05_SE_FM.indd 1 04/01/19 5:40 PM

Senior Vice President Courseware Portfolio Management: Marcia J. Horton
Director, Portfolio Management: Engineering, Computer Science & Global Editions: Julian Partridge
Portfolio Manager: Matt Goldstein
Portfolio Management Assistant: Meghan Jacoby
Managing Content Producer: Scott Disanno
Content Producer: Carole Snyder
Rights and Permissions Manager: Ben Ferrini
Manufacturing Buyer, Higher Ed, Lake Side Communications, Inc. (LSC): Maura Zaldivar-Garcia
Inventory Manager: Bruce Boundy
Product Marketing Manager: Yvonne Vannatta
Field Marketing Manager: Demetrius Hall
Marketing Assistant: Jon Bryant
Cover Designer: Integra
Cover Photo: Akepong Srichaichana/Shutterstock
Printer/Binder: LSC Communications, Inc.
Full-Service Project Management: Abhishan Sharma, Integra

Copyright © 2020 by Pearson Education, Inc. or its affiliates. All Rights Reserved. Printed in the United States of America. This publication is protected by copyright, and permission should be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise. For information regarding permissions, request forms and the appropriate contacts within the Pearson Education Global Rights & Permissions department, please visit www.pearsoned.com/permissions.

Acknowledgements of third party content appear the same page as the content, which constitutes an extension of this copyright page. Unless otherwise indicated herein, any third-party trademarks that may appear in this work are the property of their respective owners and any references to third-party trademarks, logos or other trade dress are for demonstrative or descriptive purposes only. Such references are not intended to imply any sponsorship, endorsement, authorization, or promotion of Pearson's products by the owners of such marks, or any relationship between the owner and Pearson Education, Inc. or its affiliates, authors, licensees or distributors.

The programs and applications presented in this book have been included for their instructional value. They have been tested with care, but are not guaranteed for any particular purpose. The publisher does not offer any warranties or representations, nor does it accept any liabilities with respect to the programs or applications.

Library of Congress Cataloging-in-Publication Data
Names: Gaddis, Tony, author.
Title: Starting out with visual C# / Tony Gaddis, Haywood Community College.
Description: Fifth edition. | Hoboken, NJ : Pearson, [2020] | Includes index.
Identifiers: LCCN 2018048218 | ISBN 9780135183519 | ISBN 0135183510
Subjects: LCSH: C# (Computer program language) | Visual programming languages (Computer science)
Classification: LCC QA76.73.C154 G33 2019 | DDC 005.1/18--dc23 LC record available at https://lccn.loc.gov/2018048218

10 9 8 7 6 5 4 3 2 1

Pearson

ISBN 10: 0-13-518351-0
ISBN 13: 978-0-13-518351-9

Locations of **VideoNotes**

www.pearson.com/cs-resources

VideoNote

Chapter 1

Tutorial 1-1: Starting Visual Studio and Setting Up the Environment 28
Tutorial 1-2: Starting a New Visual C# Project 30
Tutorial 1-3: Saving and Closing a Project 32
Tutorial 1-4: Opening an Existing Project 40
Tutorial 1-5: Getting Familiar with the Visual Studio Environment 41

Chapter 2

Tutorial 2-1: Creating the GUI for the *Hello World* Application . . 61
Tutorial 2-2: Writing Code for the *Hello World* Application 75
Tutorial 2-3: Creating the *Language Translator* Application 86
Tutorial 2-4: Creating the *Flags* Application 94
Tutorial 2-5: Creating the *Card Flip* Application 98
Solving the Clickable Number Images Problem 113

Chapter 3

Tutorial 3-1: The *Birth Date String* Application 126
Tutorial 3-2: Calculating Fuel Economy 147
Tutorial 3-3: Creating the *Sale Price Calculator* Application with Currency Formatting 153
Tutorial 3-4: Creating the *Test Average* Application with Exception Handling 161
Tutorial 3-5: Creating the *Change Counter* Application 170
Tutorial 3-6: Single-stepping through an Application's Code at Runtime 186
Solving the Tip, Tax, and Total Problem 196

Chapter 4

Tutorial 4-1: Completing the *Test Score Average* Application . . . 206
Tutorial 4-2: Completing the *Payroll with Overtime* Application 212
Tutorial 4-3: Completing the *Loan Qualifier* Application 218
Tutorial 4-4: Calculating Fuel Economy 241
Tutorial 4-5: Creating the *Color Theme* Application 250
Tutorial 4-6: Creating the *Time Zone* Application 257
Solving the Mass and Weight Problem 266

Chapter 5

Tutorial 5-1: Using a Loop to Calculate an Account Balance . . . 277
Tutorial 5-2: Enhancing the *Ending Balance* Application 280
Tutorial 5-3: Using the `for` Loop 290
Tutorial 5-4: Writing Data to a Text File 301
Tutorial 5-5: Appending Data to the Friend.txt File 306
Tutorial 5-6: Using a Loop to Read to the End of a File 314
Tutorial 5-7: Calculating a Running Total 318
Tutorial 5-8: Simulating Coin Tosses 328
Tutorial 5-9: Creating a Load Event Handler 333
Solving the Celsius to Fahrenheit Table Problem 340

A01_GADD3519_05_SE_FM.indd 3 04/01/19 5:40 PM

Chapter 6
Tutorial 6-1: Creating and Calling Methods 349
Tutorial 6-2: Passing an Argument to a Method 356
Tutorial 6-3: Using an Output Parameter 367
Tutorial 6-4: Writing a Value-Returning Method 376
Tutorial 6-5: Modularizing Input Validation with a Boolean Method 379
Tutorial 6-6: Practicing the *Step Into* Command 385
Tutorial 6-7: Practicing the *Step Over* Command 386
Tutorial 6-8: Practicing the *Step Out* Command 387
Solving the Kinetic Energy Problem 392

Chapter 7
Tutorial 7-1: Using an Array to Hold a List of Random Lottery Numbers 407
Tutorial 7-2: Processing an Array 432
Tutorial 7-3: Completing the *Seating Chart* Application 447
Tutorial 7-4: Completing the *Test Score List* Application 460
Tutorial 7-5: Completing the Random Card Application 467
Solving the Total Sales Problem 474

Chapter 8
Tutorial 8-1: Completing the *Password Validation* Application 485
Tutorial 8-2: Completing the *Telephone Format* Application . . 498
Tutorial 8-3: Completing the *Telephone Unformat* Application 502
Tutorial 8-4: Completing the *CSV* Reader Application 509
Solving the Sum of Numbers in a String Problem 522

Chapter 9
Tutorial 9-1: Completing the *Phonebook* Application 535
Tutorial 9-2: Completing the *Color Spectrum* Application 552
Tutorial 9-3: Storing Names and Birthdays in a Dictionary: . . . 564
Tutorial 9-4: Using a Dictionary to Simulate a Deck of Cards . 566
Solving the Course Information Problem 578

Chapter 10
Tutorial 10-1: Creating and Using the `Coin` Class 588
Tutorial 10-2: Creating and Using the `CellPhone` Class 598
Tutorial 10-3: Creating and Using the `BankAccount` Class . . . 603
Tutorial 10-4: Completing the *Cell Phone Inventory* Application 612
Tutorial 10-5: Creating an Application with Two Forms 630
Tutorial 10-6: Accessing a Control on a Different Form 635
Tutorial 10-7: Creating a Simple Logging Utility with a Static Class: 643
Solving the Pet Class Problem 652

Chapter 11
Tutorial 11-1: Creating and Testing the `SavingsAccount` and `CDAccount` Classes 664
Tutorial 11-2: Completing the *Polymorphism* Application 677
Tutorial 11-3: Completing the *Computer Science Student* Application 684
Solving the Employee and Production–Worker Classes Problem 712

Chapter 12
Tutorial 12-1: Starting the *Phone Book* Application and Creating the Phonelist.mdf Database 721
Tutorial 12-2: Completing the *Phone Book* Application 729

Tutorial 12-3: Creating the *Products* Application and Using a Details View 738
Tutorial 12-4: Creating the *Product Lookup* Application 749
Tutorial 12-5: Creating the *Multiform Products* Application . . 752
Tutorial 12-6: Creating the *Product Queries* Application 764
Tutorial 12-7: Creating the *Product Queries* Application 773
Tutorial 12-8: Creating the *Product Search* Application 777
Solving the Personnel Database Problem 787

Chapter 13

Tutorial 13-1: Working with `List` Methods and Lambdas . . . 811
Solving the List Manipulator Problem. 818

Chapter 14

Tutorial 14-1: Completing the *Student Roster* Application 827
Tutorial 14-2: Modifying the *Student Roster* Application 835
Tutorial 14-3: Completing the *Team Player* Application 849
Tutorial 14-4: Completing the *Phonelist Names* Application . . 859
Tutorial 14-5: Completing the *Product Reports* Application . . 864
Tutorial 14-6: Completing the *Query Data Source* Application 871
Solving the Unique Words Problem 881

Brief Contents

Chapter 1	**Introduction to Computers and Programming**	**1**
Chapter 2	**Introduction to Visual C#**	**51**
Chapter 3	**Processing Data**	**117**
Chapter 4	**Making Decisions**	**201**
Chapter 5	**Loops, Files, and Random Numbers**	**271**
Chapter 6	**Modularizing Your Code with Methods**	**343**
Chapter 7	**Arrays and Lists**	**397**
Chapter 8	**Text Processing**	**479**
Chapter 9	**Structures, Enumerated Types, and Dictionaries**	**525**
Chapter 10	**Introduction to Classes**	**581**
Chapter 11	**Inheritance, Polymorphism, and Interfaces**	**655**
Chapter 12	**Databases**	**715**
Chapter 13	**Delegates, Anonymous Methods, and Lambda Expressions**	**789**
Chapter 14	**Language-Integrated Query (LINQ)**	**821**
Appendix A	**C# Primitive Data Types**	**883**
Appendix B	**Additional User Interface Controls**	**885**
Appendix C	**ASCII/Unicode Characters**	**905**
Appendix D	**Answers to Checkpoint Questions**	**907**
Appendix E	**Installing LINQ to SQL Classes in Visual Studio**	**929**
Index		**931**

A01_GADD3519_05_SE_FM.indd 6 04/01/19 5:40 PM

Contents

Preface xiii

Attention Students xx

Chapter 1 **Introduction to Computers and Programming 1**

1.1 Introduction . . . 1
1.2 Hardware and Software . . . 2
1.3 How Computers Store Data . . . 7
1.4 How a Program Works . . . 11
1.5 Graphical User Interfaces . . . 18
1.6 Objects . . . 21
1.7 The Program Development Process . . . 23
1.8 Getting Started with the Visual Studio Environment . . . 27
TUTORIAL 1-1: Starting Visual Studio and Setting Up the Environment . . . 28
TUTORIAL 1-2: Starting a New Visual C# Project . . . 30
TUTORIAL 1-3: Saving and Closing a Project . . . 32
TUTORIAL 1-4: Opening an Existing Project . . . 40
TUTORIAL 1-5: Getting Familiar with the Visual Studio Environment . . . 41
Key Terms 43 • Review Questions 44 • Programming Problems 49

Chapter 2 **Introduction to Visual C# 51**

2.1 Getting Started with Forms and Controls . . . 51
2.2 Creating the GUI for Your First Visual C# Application: The *Hello World* Application . . . 60
TUTORIAL 2-1: Creating the GUI for the *Hello World* Application . . . 61
2.3 Introduction to C# Code . . . 65
2.4 Writing Code for the *Hello World* Application . . . 75
TUTORIAL 2-2: Writing Code for the *Hello World* Application . . . 75
2.5 Label Controls . . . 78
TUTORIAL 2-3: Creating the *Language Translator* Application . . . 86
2.6 Making Sense of IntelliSense . . . 89
2.7 PictureBox Controls . . . 90
TUTORIAL 2-4: Creating the *Flags* Application . . . 94
TUTORIAL 2-5: Creating the *Card Flip* Application . . . 98
2.8 Comments, Blank Lines, and Indentation . . . 102
2.9 Writing the Code to Close an Application's Form . . . 104
2.10 Dealing with Syntax Errors . . . 105
Key Terms 107 • Review Questions 107 • Programming Problems 112

A01_GADD3519_05_SE_FM.indd 7 04/01/19 5:40 PM

Chapter 3 Processing Data 117

3.1 Reading Input with TextBox Controls 117
3.2 A First Look at Variables 120
TUTORIAL 3-1: The *Birth Date String* Application 126
3.3 Numeric Data Types and Variables 131
3.4 Performing Calculations 137
3.5 Inputting and Outputting Numeric Values 142
TUTORIAL 3-2: Calculating Fuel Economy 147
3.6 Formatting Numbers with the `ToString` Method 150
TUTORIAL 3-3: Creating the *Sale Price Calculator* Application with Currency Formatting 153
3.7 Simple Exception Handling 157
TUTORIAL 3-4: Creating the *Test Average* Application with Exception Handling 161
3.8 Using Named Constants 165
3.9 Declaring Variables as Fields 166
TUTORIAL 3-5: Creating the *Change Counter* Application 170
3.10 Using the `Math` Class 174
3.11 More GUI Details 176
3.12 Using the Debugger to Locate Logic Errors 185
TUTORIAL 3-6: Single-stepping through an Application's Code at Runtime 185
Key Terms 191 • Review Questions 191 • Programming Problems 196

Chapter 4 Making Decisions 201

4.1 Decision Structures and the `if` Statement 201
TUTORIAL 4-1: Completing the *Test Score Average* Application 206
4.2 The `if-else` Statement 210
TUTORIAL 4-2: Completing the *Payroll with Overtime* Application 211
4.3 Nested Decision Structures 216
TUTORIAL 4-3: Completing the *Loan Qualifier* Application 218
4.4 Logical Operators 227
4.5 `bool` Variables and Flags 232
4.6 Comparing Strings 232
4.7 Preventing Data Conversion Exceptions with the `TryParse` Methods 236
TUTORIAL 4-4: Calculating Fuel Economy 241
4.8 Input Validation 244
4.9 Radio Buttons and Check Boxes 245
TUTORIAL 4-5: Creating the *Color Theme* Application 249
4.10 The `switch` Statement 252
4.11 Introduction to List Boxes 255
TUTORIAL 4-6: Creating the *Time Zone* Application 257
Key Terms 261 • Review Questions 261 • Programming Problems 265

Chapter 5 Loops, Files, and Random Numbers 271

5.1 More about ListBoxes 271
5.2 The `while` Loop 273
TUTORIAL 5-1: Using a Loop to Calculate an Account Balance 277

TUTORIAL 5-2: Enhancing the *Ending Balance* Application 280
5.3 The ++ and -- operators 284
5.4 The `for` Loop 285
TUTORIAL 5-3: Using the for Loop 290
5.5 The `do-while` Loop 293
5.6 Using Files for Data Storage 295
TUTORIAL 5-4: Writing Data to a Text File 301
TUTORIAL 5-5: Appending Data to the Friend.txt File 306
TUTORIAL 5-6: Using a Loop to Read to the End of a File 314
TUTORIAL 5-7: Calculating a Running Total 318
5.7 The OpenFileDialog and SaveFileDialog Controls 321
5.8 Random Numbers 326
TUTORIAL 5-8: Simulating Coin Tosses 328
5.9 The Load Event 332
TUTORIAL 5-9: Creating a Load Event Handler 333
Key Terms 336 • Review Questions 336 • Programming Problems 339

Chapter 6 **Modularizing Your Code with Methods 343**

6.1 Introduction to Methods 343
6.2 `void` Methods 345
TUTORIAL 6-1: Creating and Calling Methods 349
6.3 Passing Arguments to Methods 353
TUTORIAL 6-2: Passing an Argument to a Method 356
6.4 Passing Arguments by Reference 364
TUTORIAL 6-3: Using an Output Parameter 367
6.5 Value-Returning Methods 371
TUTORIAL 6-4: Writing a Value-Returning Method 376
TUTORIAL 6-5: Modularizing Input Validation with a Boolean Method 379
6.6 Debugging Methods 384
TUTORIAL 6-6: Practicing the *Step Into* Command 385
TUTORIAL 6-7: Practicing the *Step Over* Command 386
TUTORIAL 6-8: Practicing the *Step Out* Command 387
Key Terms 389 • Review Questions 389 • Programming Problems 385

Chapter 7 **Arrays and Lists 397**

7.1 Value Types and Reference Types 397
7.2 Array Basics 400
TUTORIAL 7-1: Using an Array to Hold a List of Random Lottery Numbers 407
7.3 Working with Files and Arrays 412
7.4 Passing Arrays as Arguments to Methods 415
7.5 Some Useful Array Algorithms 422
TUTORIAL 7-2: Processing an Array 432
7.6 Advanced Algorithms for Sorting and Searching Arrays 437
7.7 Two-Dimensional Arrays 444
TUTORIAL 7-3: Completing the *Seating Chart* Application 447
7.8 Jagged Arrays 453

7.9 The `List` Collection 454
TUTORIAL 7-4: Completing the *Test Score List* Application 460
7.10 The ImageList Control 465
TUTORIAL 7-5: Completing the *Random Card* Application 467
Key Terms 470 • Review Questions 470 • Programming Problems 474

Chapter 8 **Text Processing 479**

8.1 Introduction 479
8.2 Working with Characters 479
TUTORIAL 8-1: Completing the *Password Validation* Application 485
8.3 Working with Substrings 489
TUTORIAL 8-2: Completing the *Telephone Format* Application 498
TUTORIAL 8-3: Completing the *Telephone Unformat* Application 502
8.4 Tokenizing Strings 506
TUTORIAL 8-4: Completing the *CSV Reader* Application 509
8.5 The `String.Format` Method 514
Key Terms 519 • Review Questions 519 • Programming Problems 521

Chapter 9 **Structures, Enumerated Types, and Dictionaries 525**

9.1 Introduction 525
9.2 Structures 525
TUTORIAL 9-1: Completing the *Phonebook* Application 535
9.3 The `DateTime` and `TimeSpan` Structures 541
9.4 Enumerated Types 549
TUTORIAL 9-2: Completing the *Color Spectrum* Application 552
9.5 Dictionaries 556
TUTORIAL 9-3 Storing Names and Birthdays in a Dictionary: 564
TUTORIAL 9-4 Using a Dictionary to Simulate a Deck of Cards 566
Key Terms 572 • Review Questions 572 • Programming Problems 575

Chapter 10 **Introduction to Classes 581**

10.1 Introduction to Classes 581
TUTORIAL 10-1: Creating and Using the `Coin` Class 588
10.2 Properties 593
TUTORIAL 10-2: Creating and Using the `CellPhone` Class 598
10.3 Parameterized Constructors and Overloading 602
TUTORIAL 10-3: Creating and Using the `BankAccount` Class 603
10.4 Storing Class Type Objects in Arrays and `List`s 611
TUTORIAL 10-4: Completing the *Cell Phone Inventory* Application 612
10.5 Finding the Classes and Their Responsibilities in a Problem 616
10.6 Application of Classes: Creating Multiple Forms in a Project 625
TUTORIAL 10-5: Creating an Application with Two Forms 630
TUTORIAL 10-6: Accessing a Control on a Different Form 635
10.7 Static Class Members 639
TUTORIAL 10-7: Creating a Simple Logging Utility with a Static Class: 643
Key Terms 648 • Review Questions 648 • Programming Problems 652

Chapter 11 **Inheritance, Polymorphism, and Interfaces 655**

11.1 Inheritance 655
TUTORIAL 11-1: Creating and Testing the `SavingsAccount` and `CDAccount` Classes 664
11.2 Polymorphism 672
TUTORIAL 11-2: Completing the *Polymorphism* Application 677
11.3 Abstract Classes 682
TUTORIAL 11-3: Completing the *Computer Science Student* Application 684
11.4 Extension Methods. 688
TUTORIAL 11-4: Creating Extension Methods 693
11.5 Interfaces 696
TUTORIAL 11-5: Writing and Implementing an Interface 699
Key Terms 708 • Review Questions 708 • Programming Problems 712

Chapter 12 **Databases 715**

12.1 Introduction to Database Management Systems 715
12.2 Tables, Rows, and Columns. 717
12.3 Creating a Database in Visual Studio. 720
TUTORIAL 12-1: Starting the *Phone Book* Application and Creating the Phonelist.mdf Database. 721
12.4 The DataGridView Control. 729
TUTORIAL 12-2: Completing the *Phone Book* Application 729
12.5 Connecting to an Existing Database and Using Details View Controls. . . . 736
TUTORIAL 12-3: Creating the *Products* Application and Using a Details View . . . 738
12.6 More About Data-Bound Controls. 746
TUTORIAL 12-4: Creating the *Product Lookup* Application 749
TUTORIAL 12-5: Creating the *Multiform Products* Application 752
12.7 Selecting Data with the SQL `Select` Statement 758
TUTORIAL 12-6: Creating the *Product Queries* Application 764
TUTORIAL 12-7: Creating the *Product Queries* Application 773
TUTORIAL 12-8: Creating the *Product Search* Application 777
Key Terms 783 • Review Questions 783 • Programming Problems 787

Chapter 13 **Delegates, Anonymous Methods, and Lambda Expressions 789**

13.1 Delegates 789
13.2 Anonymous Methods 795
13.3 Lambda Expressions. 797
13.4 Built-In Delegates: `Action`, `Func` and `Predicate` 803
TUTORIAL 13-1: Working with `List` Methods and Lambdas:. 811
Key Terms 815 • Review Questions 815 • Programming Challenges 818

Chapter 14 **Language-Integrated Query (LINQ) 821**

14.1 Introduction to LINQ 821
TUTORIAL 14-1: Completing the *Student Roster* Application 827
14.2 LINQ Method Syntax 831
TUTORIAL 14-2: Modifying the *Student Roster* Application 835

14.3 More LINQ Extension Methods 837
TUTORIAL 14-3: Completing the *Team Player* Application 849
14.4 Using LINQ to Query a Database 854
TUTORIAL 14-4: Completing the *Phonelist Names* Application 859
TUTORIAL 14-5: Completing the *Product Reports* Application 864
TUTORIAL 14-6: Completing the *QueryDataSource* Application 871
Key Terms 878 • Review Questions 878 • Programming Problems 881

Appendix A **C# Primitive Data Types** 883
Appendix B **Additional User Interface Controls** 885
Appendix C **ASCII/Unicode Characters** 905
Appendix D **Answers to Checkpoint Questions** 907
Appendix E **Installing LINQ to SQL Classes in Visual Studio** 929
Index 931

Preface

Welcome to *Starting Out with Visual C#*, Fifth Edition. This book is intended for an introductory programming course and is ideal for students with no prior experience. Students who are new to programming will appreciate the clear, down-to-earth explanations and the detailed walk-throughs that are provided by the hands-on tutorials. More experienced students will appreciate the depth of detail as they learn about the .NET Framework, databases, Language-Integrated Query, and other topics.

As with all the books in the *Starting Out With* series, the hallmark of this text is its clear, friendly, and easy-to-understand writing. In addition, it is rich in example programs that are concise and practical. The programs in this book include short examples that highlight specific programming topics, as well as more involved examples that focus on problem solving. Each chapter provides numerous hands-on tutorials that guide the student through each step of the development of an application. In addition to detailed, step-by-step instructions, the tutorials also provide the application's completed code and screen captures of the completed forms.

New to This Edition

The biggest changes in this edition appear in the last half of the book. Some of the chapters have been reorganized, and two new chapters have been added. Here is a summary of the reorganization:

- **The ImageList control is now covered along with Arrays and Lists in Chapter 7.** Since an ImageList is a collection of images, it is fitting that this control be covered in the same chapter as Arrays and Lists.
- **Chapter 8 has been split into two chapters, with new material added to both of the resulting chapters.** In this edition, Chapter 8 is titled *Text Processing* and Chapter 9 is titled *Structures, Enumerated Types, and Dictionaries*.
- **Two new chapters have been added.** In this edition, Chapter 13 is titled *Delegates and Lambda Expressions* and Chapter 14 is titled *Language-Integrated Query (LINQ)*.

An abundance of new topics and improvements has been added to this edition:

- **`var` Keyword:** This edition introduces the `var` keyword for variable declaration.
- **Advanced String Formatting:** Chapter 8 includes a new section on formatting strings with the `String.Format` method.
- **Working with Dates and Times:** Chapter 9 includes a new section about working with dates and **times**. The section covers the `DateTime` and `TimeSpan` data types, as well as the DateTimePicker control.
- **Dictionaries:** Chapter 9 includes a new section on the `Dictionary` collection. The student learns to store and work with data as key–value pairs in dictionaries.
- **More Use of Auto-Properties:** In the previous editions, property declarations were almost always written with explicitly declared backing fields, even when the properties were simply used to set and get a value. Where possible, those property declarations have been rewritten as auto-properties.
- **Object Initializer Syntax:** Object initializer syntax is introduced in Chapters 8 and 10 as an alternative way to **declare** and initialize structures and class instances.
- **Static Members:** Chapter 10 has a new section on static fields, properties, methods, and classes.
- **Sealed Classes and Methods:** Chapter 11 has a new discussion on using the `sealed` keyword to prevent classes from being inherited from, and to prevent methods from being overridden.

A01_GADD3519_05_SE_FM.indd 13 04/01/19 5:40 PM

- **Extension Methods:** Chapter 11 covers extension methods, which allow you to extend classes that you cannot inherit from, as well as the primitive data types.
- **Interfaces**: Chapter 11 now includes a section on writing and using interfaces.
- **Delegates:** Chapter 13 introduces the student to delegates.
- **Anonymous Methods:** Chapter 13 also introduces anonymous methods, which go hand-in-hand with delegates.
- **Lambda Expressions:** Chapter 13 concludes by showing the student how to use a lambda expression to concisely create a delegate and anonymous method.
- **Language-Integrated Query (LINQ):** Chapter 14 introduces LINQ, and shows how to query simple data structures such as arrays and Lists using **LINQ to Objects**, and how to query databases using **LINQ to SQL**.

A GUI-Based Approach

Beginning students are more motivated to learn programming when their applications have some sort of graphical element, such as a graphical user interface (GUI). Students using this book will learn to create GUI-based, event-driven, Visual C# applications. The Visual Studio environment is used to create forms that are rich with user interface controls and graphical images.

Learn to Use Objects Early, Learn to Write Classes Later

This book explains what an object is very early and shows the student how to create objects from classes that are provided by the .NET Framework. It then introduces the student to the fundamentals of input and output, control structures, methods, arrays and lists, and file operations. Then, the student learns to write his or her own classes and explores the topics of inheritance and polymorphism.

Brief Overview of Each Chapter

Chapter 1: Introduction to Computers and Programming. This chapter begins by giving a very concrete and easy-to-understand explanation of how computers work, how data is stored and manipulated, and why we write programs in high-level languages. In this chapter, the student learns what an object is and sees several examples by studying the objects that make up a program's GUI. The chapter discusses steps in the programming development cycle. It also gives an introduction to the Visual Studio environment.

Chapter 2: Introduction to Visual C#. In this chapter, the student learns to create forms with labels, buttons, and picture boxes and learns to modify control properties. The student is introduced to C# code and learns the organizational structure of namespaces, classes, and methods. The student learns to write simple event-driven applications that respond to button clicks or provide interaction through clickable images. The importance of commenting code is also discussed.

Chapter 3: Processing Data. This chapter introduces variables and data types. It discusses the use of local variables and variables declared as fields within a form class. The student learns to create applications that read input from TextBox controls, perform mathematical operations, and produce formatted output. The student learns about the exceptions that can occur when the user enters invalid data into a TextBox and learns to write simple exception-handling code to deal with those problems. Named constants are introduced as a way of representing unchanging values and creating self-documenting, maintainable code. The student also learns more intricacies of creating graphical user interfaces. The chapter concludes with a discussion and tutorial on using the Visual Studio debugger to locate logic errors by single-stepping through an application's code.

Chapter 4: Making Decisions. In this chapter, the student learns about relational operators and Boolean expressions and is shown how to control the flow of a program

with decision structures. The `if`, `if-else`, and `if-else-if` statements are covered. Nested decision structures, logical operators, and the `switch` statement are also discussed. The student learns to use the `TryParse` family of methods to validate input and prevent exceptions. Radio buttons, check boxes, and list boxes are introduced as ways to let the user select items in a GUI.

Chapter 5: Loops, Files, and Random Numbers. This chapter shows the student how to use loops to create repetition structures. The `while` loop, the `for` loop, and the `do-while` loop are presented. Counters, accumulators, and running totals are also discussed. This chapter also introduces sequential file input and output and using text files. The student learns various programming techniques for writing data to text files and reading the contents of test files. The chapter concludes with a discussion of pseudorandom numbers, their applications, and how to generate them.

Chapter 6: Modularizing Your Code with Methods. In this chapter, the student first learns how to write and call `void` methods as well as value-returning methods. The chapter shows the benefits of using methods to modularize programs and discusses the top-down design approach. Then, the student learns to pass arguments to methods. Passing by value, by reference, and output parameters are discussed. The chapter concludes with a discussion and tutorial on debugging methods with the Visual Studio step-into, step-over, and step-out, commands.

Chapter 7: Arrays and Lists. Arrays and lists are reference-type objects in C# so this chapter begins by discussing the difference between value type and reference type objects in the C# language. Then, the student learns to create and work with single-dimensional and two-dimensional arrays. The student learns to pass arrays as arguments to methods, transfer data between arrays and files, work with partially filled arrays, and create jagged arrays. Many examples of array processing are provided, including examples of finding the sum, average, highest, and lowest values in an array. Finally, the student learns to create `List` objects and store data in them, and use the ImageList control, a data structure for storing and retrieving images.

Chapter 8: Text Processing. This chapter discusses various techniques for working with text. The topics include working with characters, working with substrings, tokenizing strings, and formatting strings.

Chapter 9: Structures, Enumerated Types, and Dictionaries. In this chapter, the student learns to use structures to encapsulate several variables into a single item. The student next learns to create and use enumerated types. Last, the student learns to use the Dictionary collection from the .NET Framework to store and work with data as key–value pairs.

Chapter 10: Introduction to Classes. Up to this point, the student has extensively used objects that are instances of .NET Framework classes. In this chapter, the student learns to write classes to create his or her own objects with fields, methods, and constructors. The student learns how to implement various types of properties within a class, including auto-properties and read-only auto-properties. Creating arrays of objects and storing objects in a `List` are also discussed. A primer on finding the classes in a problem as well as their responsibilities is provided. The chapter shows the student how to create multiple form classes in a project, instantiate those classes, and display them. A tutorial is given where the student creates a multiform application in which the code in one form accesses controls on another form. The chapter concludes by discussing static members and static classes.

Chapter 11: Inheritance, Polymorphism, and Interfaces. The study of classes continues in this chapter with the subjects of inheritance and polymorphism. The topics covered include base classes, derived classes, how constructors functions work in inheritance, method overriding, and polymorphism. Abstract classes and abstract methods are also discussed. The chapter discusses extension methods, and concludes with a section on interfaces.

Chapter 12: Databases. This chapter introduces the student to basic database concepts. The student first learns about tables, rows, and columns and how to create an SQL Server database in Visual Studio. The student then learns how to connect a database to a Visual C# application and display a table in a DataGridView control, a Details view, and other data-bound controls. Finally, the student learns how to write SQL Select statements to retrieve data from a table.

Chapter 13: Delegates and Lambda Expressions. Lambda expressions have become commonplace in C# programming. Before the student can really understand lambda expressions, he or she must first understand delegates and anonymous methods. The goal of this chapter is to build a foundation of knowledge about how delegates and anonymous methods work, and then build an understanding of lambda expressions.

Chapter 14: Language-Integrated Query (LINQ). This chapter begins by introducing the student to LINQ as a tool for querying the data in common data structures such as arrays and Lists, using LINQ to Objects. It discusses both query syntax and method syntax. Several useful LINQ extension methods are also discussed. The chapter concludes with a section on LINQ to SQL and discusses how to use LINQ to query a database.

Appendix A: C# Primitive Data Types. This appendix gives an overview of the primitive data types available in C#.

Appendix B: Additional User Interface Controls. This appendix shows how to create a variety of controls such as ToolTips, combo boxes, scroll bars, TabControls, WebBrowser controls, ErrorProvider components, and menu systems.

Appendix C: ASCII/Unicode Characters. This appendix lists the ASCII (American Standard Code for Information Interchange) character set, which is also the Latin Subset of Unicode.

Appendix D: Answers to Checkpoint Questions. This appendix provides the answers to the Checkpoint questions that appear throughout each chapter in the book.

Appendix E: Installing LINQ to SQL. This appendix shows how to use the Visual Studio Installer to download and install LINQ to SQL.

Organization of the Text

The text teaches Visual C# step by step. Each chapter covers a major set of programming topics, introduces controls and GUI elements, and builds knowledge as the student progresses through the book. Although the chapters can be easily taught in their existing sequence, there is some flexibility. Figure P-1 shows the chapter dependencies. As shown in the figure, Chapters 1–7 present the fundamentals of Visual C# programming and should be covered in sequence. Then, you can move directly to Chapter 8, 9, 10, or 12. Chapter 11 should be covered after Chapter 10. Then Chapter 13 and 14 can be covered in order.

Features of the Text

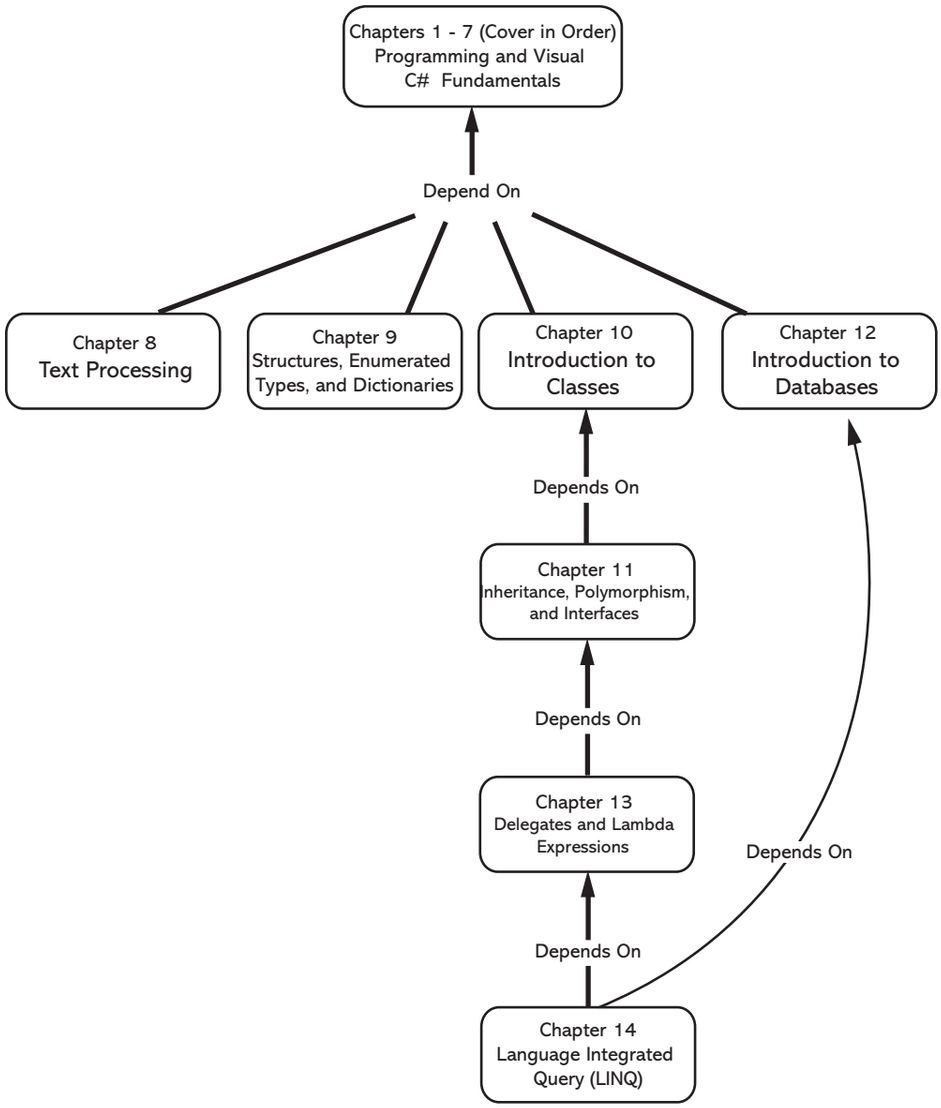
Concept Statements. Each major section of the text starts with a concept statement. This statement concisely summarizes the main point of the section.

Tutorials. Each chapter has several hands-on tutorials that guide the student through the development of an application. Each tutorial provides detailed, step-by-step instructions, as well as the application's completed code and screen captures of the completed forms.

Example Programs. Each chapter has an abundant number of code examples designed to highlight the current topic.

Notes. Notes appear at several places throughout the text. They are short explanations of interesting or often misunderstood points relevant to the topic at hand.

Figure P-1 Chapter dependencies



Tips. Tips advise the student on the best techniques for approaching different programming or animation problems.

Warnings. Warnings caution students about programming techniques or practices that can lead to malfunctioning programs or lost data.

Checkpoints. Checkpoints are questions placed at intervals throughout each chapter. They are designed to query the student's knowledge quickly after learning a new topic. The answers to the Checkpoint questions can be found in Appendix D.

Review Questions. Each chapter presents a thorough and diverse set of Review Questions. They include Multiple Choice, True/False, Algorithm Workbench, and Short Answer.

Programming Problems. Each chapter offers a pool of Programming Problems designed to solidify the student's knowledge of the topics currently being studied.

VideoNote

VideoNotes. Each tutorial in the book has an accompanying online VideoNote that can be accessed on the book's companion Web site www.pearson.com/gaddis. Students can

follow along with the author as he or she works through each tutorial in the videos. Also, one programming problem at the end of each chapter has an accompanying VideoNote that shows the student how to create the solution.

Supplements

Student The following supplementary material is available with the book:

- Source code and files required for the chapter tutorials are available at www.pearson.com/cs-resources
- A complete set of online VideoNotes that take the student through each tutorial in the book. Also, one programming problem from each chapter has an accompanying VideoNote that helps the student create a solution. You may access the VideoNotes by going to www.pearson.com/cs-resources.

Instructor The following supplements are available to qualified instructors:

- Answers to all Review Questions in the text
- Solutions for all Programming Problems in the text
- Completed versions of all tutorials
- PowerPoint presentation slides for every chapter
- Test bank

For information on how to access these supplements, visit the Pearson Education Instructor Resource Center at www.pearsonhighered.com/irc.

Acknowledgments

There were many helping hands in the development and publication of this text. I would like to thank the following faculty reviewers for their helpful suggestions and expertise:

Matthew Alimagham
Spartanburg Community College

Carolyn Borne
Louisiana State University

Arthur E. Carter
Radford University

Sallie B. Dodson
Radford University

Elizabeth Freije
Indiana University—Purdue University, Indianapolis

Bettye J. Parham
Daytona State College

Wendy L. Payne
Gulf Coast State College

Jason Sharp
Tarleton State University

John Van Assen
York Technical College

Reginald White
Black Hawk College

Dawn R. Wick
Southwestern Community College

I also want to thank everyone at Pearson for making the *Starting Out With ...* series so successful. I have worked so closely with the team at Pearson that I consider them among my closest friends. I am extremely fortunate to have Matt Goldstein as my editor, and Meghan Jacoby as Editorial Assistant. They have guided me through the process of revising this book, as well as many others. I am also fortunate to have Demetrius Hall and Yvonne Vannatta as my marketing team. Their hard work is truly inspiring, and they do a great job of getting this book out to the academic community. The production team, led by Carole Snyder, worked tirelessly to make this book a reality. Thanks to you all!

About the Author

Tony Gaddis is the principal author of the *Starting Out With* series of textbooks. Tony has nearly 20 years experience teaching computer science courses at Haywood Community College in North Carolina. He is a highly acclaimed instructor who was previously selected as the North Carolina Community College Teacher of the Year and has received the Teaching Excellence Award from the National Institute for Staff and Organizational Development.

The *Starting Out With* series includes introductory books using the C++ programming language, the Java™ programming language, Microsoft® Visual Basic®, Microsoft® C#®, Python, Programming Logic and Design, MIT App Inventor, and Alice, all published by Pearson Education.

Attention Students

Installing Visual Studio

To complete the tutorials and programming problems in this book, you need to install Visual Studio 2017, or a later version, on your computer. We recommend that you download Visual Studio Community 2017 from the following website, and install it on your system:

www.visualstudio.com

Visual Studio Community 2017 is a free, full-featured development environment, and is a perfect companion for this textbook.

NOTE: If you are working in your school's computer lab, there is a good chance that **Microsoft Visual Studio** has already been installed. If this is the case, your instructor will show you how to start Visual Studio.

Installing the Student Sample Program Files

The Student Sample Program files that accompany this book are available for download from the book's companion Web site at:

www.pearson.com/cs-resources

These files are required for many of the book's tutorials. Simply download the Student Sample Program files to a location on your hard drive where you can easily access them.

A01_GADD3519_05_SE_FM.indd 20 04/01/19 5:40 PM