

Join the discussion @ [p2p.wrox.com](http://p2p.wrox.com)



COVERS VISUAL C# 2012



Beginning

# Visual C#<sup>®</sup> 2012 Programming

Karli Watson, Jacob Vibe Hammer, Jon D. Reid, Morgan Skinner, Daniel Kemper, Christian Nagel



# BEGINNING VISUAL C#® 2012 PROGRAMMING

---

INTRODUCTION .....xxxi

## ► PART I THE C# LANGUAGE

<b>CHAPTER 1</b>	Introducing C# .....	3
<b>CHAPTER 2</b>	Writing a C# Program .....	13
<b>CHAPTER 3</b>	Variables and Expressions .....	25
<b>CHAPTER 4</b>	Flow Control .....	49
<b>CHAPTER 5</b>	More About Variables .....	79
<b>CHAPTER 6</b>	Functions .....	105
<b>CHAPTER 7</b>	Debugging and Error Handling .....	131
<b>CHAPTER 8</b>	Introduction to Object-Oriented Programming .....	157
<b>CHAPTER 9</b>	Defining Classes .....	177
<b>CHAPTER 10</b>	Defining Class Members .....	203
<b>CHAPTER 11</b>	Collections, Comparisons, and Conversions .....	235
<b>CHAPTER 12</b>	Generics .....	279
<b>CHAPTER 13</b>	Additional OOP Techniques .....	313
<b>CHAPTER 14</b>	C# Language Enhancements .....	339

## ► PART II WINDOWS PROGRAMMING

<b>CHAPTER 15</b>	Basic Desktop Programming .....	381
<b>CHAPTER 16</b>	Advanced Desktop Programming .....	425
<b>CHAPTER 17</b>	Windows Store Apps .....	469
<b>CHAPTER 18</b>	Deploying Desktop Applications .....	513

## ► PART III WEB PROGRAMMING

<b>CHAPTER 19</b>	ASP.NET Web Programming .....	535
<b>CHAPTER 20</b>	Deploying Web Applications .....	591

*Continued*

► **PART IV DATA ACCESS**

**CHAPTER 21** File System Data..... 605  
**CHAPTER 22** XML..... 639  
**CHAPTER 23** Introduction to LINQ ..... 663  
**CHAPTER 24** Applying LINQ ..... 699

► **PART V ADDITIONAL TECHNIQUES**

**CHAPTER 25** Windows Communication Foundation..... 727  
**CHAPTER 26** Windows Workflow Foundation..... 759  
**APPENDIX A** Exercise Solutions ..... 779  
**INDEX**..... 827

BEGINNING

# **Visual C#<sup>®</sup> 2012 Programming**



BEGINNING

# Visual C#<sup>®</sup> 2012 Programming

Karli Watson  
Jacob Vibe Hammer  
John D. Reid  
Morgan Skinner  
Daniel Kemper  
Christian Nagel



WILEY

Wiley Publishing Inc

## Beginning Visual C#® 2012 Programming

Published by  
John Wiley & Sons, Inc.  
10475 Crosspoint Boulevard  
Indianapolis, IN 46256  
[www.wiley.com](http://www.wiley.com)

Copyright © 2013 by John Wiley & Sons, Inc., Indianapolis, Indiana

Published simultaneously in Canada

ISBN: 978-1-118-31441-8  
ISBN: 978-1-118-33194-1 (ebk)  
ISBN: 978-1-118-33540-6 (ebk)  
ISBN: 978-1-118-39637-7 (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://booksupport.wiley.com>. For more information about Wiley products, visit [www.wiley.com](http://www.wiley.com).

**Library of Congress Control Number:** 2012946841

**Trademarks:** Wiley, the Wiley logo, Wrox, the Wrox logo, Wrox 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.



*for donna*

— KARLI WATSON

*Love is as strong as death;  
Many waters cannot quench love,  
neither can the floods drown it.*

*From the inside of my Mum's hymnbook, of which I  
now have the privilege of being the custodian.*

— MORGAN SKINNER



# ABOUT THE AUTHORS

**KARLI WATSON** is an IT contractor and author currently working in London in the financial sector. For the most part, he immerses himself in .NET (in particular C#) and has written numerous books in the field for several publishers. He specializes in communicating complex ideas in a way that is accessible to anyone with a passion to learn, and spends much of his time playing with new technology to find new things to teach people about.

During those (seemingly few) times where he isn't doing the above, Karli will probably be wishing he was hurtling down a mountain on a snowboard. Or possibly trying to get his novel published. Either way, you'll know him by his brightly colored clothes. You can also find him tweeting online as @karlequin, and maybe one day he'll get round to making himself a website.

**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 son.

**JON D. REID** is the Director of Software Technology for IFS Metrix Service Management ([www.IFSWORLD.com/Metrix](http://www.IFSWORLD.com/Metrix)). He has coauthored a number of .NET books, including *Beginning Visual C# 2010*, *Fast Track C#*, *Pro Visual Studio .NET*, and many others.

**MORGAN SKINNER** started messing with computers in 1980 when he first started programming at school in assembly language. Since then he's used many languages commercially, including Pascal, Modula-2, VAX Macro assembly language, Smalltalk, PowerBuilder, C, C++ and C# (to name the more well-known ones). He joined Microsoft in 2001 after getting his hands on .NET for the first time, and he spent nearly 10 years there as an Application Development Consultant, working with some of the smallest — and largest — companies in the UK. Morgan left Microsoft in 2011 and is now an independent contractor working on bespoke systems; see [www.morganskinner.com](http://www.morganskinner.com) for more details.

**DANIEL KEMPER** is a software architect with a couple of Microsoft certifications. He specializes in rich Internet application, desktop client, and reporting technologies.

**CHRISTIAN NAGEL** is a Microsoft Regional Director and Microsoft MVP, an associate of thinkecture, and founder of CN innovation. A software architect and developer, he offers training and consulting on how to develop solutions using the Microsoft platform. He draws on more than 25 years of software development experience. Christian started his computing career with PDP 11 and VAX/VMS systems, covering a variety of languages and platforms. Since 2000, when .NET was just a technology preview, he has been working with various .NET technologies to build .NET solutions. Currently, he mainly coaches the development of Windows 8 apps accessing Windows Azure services. With his profound knowledge of Microsoft technologies, he has written numerous books and is certified as a Microsoft Certified Trainer and Professional Developer. Christian speaks at international conferences such as TechEd, Basta!, and TechDays, and he founded INETA Europe to support .NET user groups. You can contact Christian via his websites, [www.cninnovation.com](http://www.cninnovation.com) and [www.thinkecture.com](http://www.thinkecture.com). You can also follow his tweets at @christiannagel.

# ABOUT THE TECHNICAL EDITORS

**DOUG HOLLAND** is an architect with Microsoft's Developer and Platform Evangelism team and works with Microsoft's strategic ISV partners to help bring new and exciting experiences to consumers on Windows 8 and Windows Phone 8.

**RICHARD HOPTON** has been developing business software systems for 10 years, currently focusing on designing and building highly scalable REST-based API solutions using C# for a digital media company in London, UK. Richard has been published in Microsoft's monthly developer newsletter, MSDN Flash, and has spoken at numerous developer community events throughout the UK.

**MARCEL MEIJER** has lived in the world of information and communications technologies for more than 15 years. Currently, he is mainly concerned with Windows Azure, the cloud, C#, software development, and architecture. He works as a senior architect at VX Company. In his spare time, he is a board member of the SDN (Software Development Network; [www.sdn.nl](http://www.sdn.nl)). At SDN, he is responsible for arranging speakers for the SDN Events (SDE); selecting content for SDN Conferences; and arranging and editing content for SDN Magazine.

# CREDITS

**ACQUISITIONS EDITOR**

Mary James

**PROJECT EDITOR**

Patrick Meader

**TECHNICAL EDITORS**

Doug Holland  
Richard Hopton  
Marcel Meijer

**PRODUCTION EDITOR**

Christine Mugnolo

**COPY EDITOR**

Kezia Endsley

**EDITORIAL MANAGER**

Mary Beth Wakefield

**FREELANCER EDITORIAL MANAGER**

Rosemarie Graham

**ASSOCIATE DIRECTOR OF MARKETING**

David Mayhew

**MARKETING MANAGER**

Ashley Zurcher

**BUSINESS MANAGER**

Amy Knies

**PRODUCTION MANAGER**

Tim Tate

**VICE PRESIDENT AND EXECUTIVE GROUP  
PUBLISHER**

Richard Swadley

**VICE PRESIDENT AND EXECUTIVE PUBLISHER**

Neil Edde

**ASSOCIATE PUBLISHER**

Jim Minatel

**PROJECT COORDINATOR, COVER**

Katie Crocker

**PROOFREADERS**

Sarah Kaikini, Word One New York  
Scott Klemp, Word One New York  
James Saturnio, Word One New York

**INDEXER**

Robert Swanson

**COVER DESIGNER**

LeAndra Young

**COVER IMAGE**

© Lisa Loyd / iStockphoto



# ACKNOWLEDGMENTS

**THANKS ONCE AGAIN TO EVERYONE AT WILEY** for help, encouragement, and understanding. Striking the balance between getting the book done quickly and ensuring it's accurate in the face of numerous product and naming changes is never easy, but between us I think we've just about managed it. Special thanks to Patrick Meader for remaining (mostly) calm throughout the project — or at least calmer than me... As ever, no acknowledgements would be complete without thanks to my wife, donna, who very nearly succeeds in keeping me sane during writing periods. And, of course, thanks to you for (hopefully) buying this book, and the very best of luck in your coding adventures!

— KARLI WATSON





# CONTENTS

*INTRODUCTION*

*xxxi*

## **PART I: THE C# LANGUAGE**

<b>CHAPTER 1: INTRODUCING C#</b>	<b>3</b>
<b>What Is the .NET Framework?</b>	<b>3</b>
What's in the .NET Framework?	4
Writing Applications Using the .NET Framework	4
CIL and JIT	4
Assemblies	5
Managed Code	5
Garbage Collection	6
Fitting It Together	6
Linking	7
<b>What Is C#?</b>	<b>7</b>
Applications You Can Write with C#	8
C# in this Book	8
<b>Visual Studio 2012</b>	<b>8</b>
Visual Studio Express 2012 Products	9
Solutions	9
<b>Summary</b>	<b>10</b>
<b>CHAPTER 2: WRITING A C# PROGRAM</b>	<b>13</b>
<b>The Visual Studio 2012 Development Environment</b>	<b>14</b>
<b>Console Applications</b>	<b>16</b>
The Solution Explorer	18
The Properties Window	19
The Error List Window	19
<b>Desktop Applications</b>	<b>20</b>
<b>Summary</b>	<b>23</b>
<b>CHAPTER 3: VARIABLES AND EXPRESSIONS</b>	<b>25</b>
<b>Basic C# Syntax</b>	<b>26</b>
<b>Basic C# Console Application Structure</b>	<b>28</b>
<b>Variables</b>	<b>29</b>
Simple Types	29

---

Variable Naming	33
Naming Conventions	34
Literal Values	34
Variable Declaration and Assignment	36
<b>Expressions</b>	<b>37</b>
Mathematical Operators	37
Assignment Operators	41
Operator Precedence	42
Namespaces	42
<b>Summary</b>	<b>45</b>
<b>CHAPTER 4: FLOW CONTROL</b>	<b>49</b>
<hr/>	
<b>Boolean Logic</b>	<b>49</b>
Boolean Assignment Operators	52
Bitwise Operators	53
Operator Precedence Updated	57
<b>The goto Statement</b>	<b>58</b>
<b>Branching</b>	<b>59</b>
The Ternary Operator	59
The if Statement	59
Checking More Conditions Using if Statements	62
The switch Statement	63
<b>Looping</b>	<b>65</b>
do Loops	66
while Loops	68
for Loops	70
Interrupting Loops	74
Infinite Loops	75
<b>Summary</b>	<b>75</b>
<b>CHAPTER 5: MORE ABOUT VARIABLES</b>	<b>79</b>
<hr/>	
<b>Type Conversion</b>	<b>80</b>
Implicit Conversions	80
Explicit Conversions	81
Explicit Conversions Using the Convert Commands	84
<b>Complex Variable Types</b>	<b>86</b>
Enumerations	87
Defining Enumerations	87
Structs	90
Defining Structs	90

---

Arrays	92
Declaring Arrays	93
foreach Loops	95
Multidimensional Arrays	96
Arrays of Arrays	97
<b>String Manipulation</b>	<b>98</b>
<b>Summary</b>	<b>102</b>
<b>CHAPTER 6: FUNCTIONS</b>	<b>105</b>
<hr/>	
<b>Defining and Using Functions</b>	<b>106</b>
Return Values	108
Parameters	109
Parameter Matching	111
Parameter Arrays	111
Reference and Value Parameters	113
Out Parameters	115
<b>Variable Scope</b>	<b>116</b>
Variable Scope in Other Structures	118
Parameters and Return Values versus Global Data	120
<b>The Main( ) Function</b>	<b>121</b>
<b>Struct Functions</b>	<b>123</b>
<b>Overloading Functions</b>	<b>124</b>
<b>Using Delegates</b>	<b>125</b>
<b>Summary</b>	<b>128</b>
<b>CHAPTER 7: DEBUGGING AND ERROR HANDLING</b>	<b>131</b>
<hr/>	
<b>Debugging in Visual Studio</b>	<b>132</b>
Debugging in Nonbreak (Normal) Mode	132
Outputting Debugging Information	133
Tracepoints	137
Diagnostics Output Versus Tracepoints	139
Debugging in Break Mode	140
Entering Break Mode	140
Monitoring Variable Content	142
Stepping Through Code	144
Immediate and Command Windows	146
The Call Stack Window	146
<b>Error Handling</b>	<b>147</b>
try...catch...finally	147
Listing and Configuring Exceptions	152

---

Notes on Exception Handling	152
<b>Summary</b>	<b>153</b>
<b>CHAPTER 8: INTRODUCTION TO OBJECT-ORIENTED PROGRAMMING</b>	<b>157</b>
<hr/>	
<b>What Is Object-Oriented Programming?</b>	<b>158</b>
What Is an Object?	158
Properties and Fields	159
Methods	160
Everything's an Object	161
The Life Cycle of an Object	161
Constructors	161
Destructors	162
Static and Instance Class Members	162
Static Constructors	162
Static Classes	163
<b>OOP Techniques</b>	<b>163</b>
Interfaces	163
Disposable Objects	164
Inheritance	165
Polymorphism	167
Interface Polymorphism	168
Relationships Between Objects	168
Containment	168
Collections	169
Operator Overloading	169
Events	170
Reference Types Versus Value Types	170
<b>OOP in Desktop Applications</b>	<b>171</b>
<b>Summary</b>	<b>174</b>
<b>CHAPTER 9: DEFINING CLASSES</b>	<b>177</b>
<hr/>	
<b>Class Definitions in C#</b>	<b>177</b>
Interface Definitions	180
<b>System.Object</b>	<b>182</b>
<b>Constructors and Destructors</b>	<b>184</b>
Constructor Execution Sequence	185
<b>OOP Tools in Visual Studio</b>	<b>188</b>
The Class View Window	188
The Object Browser	190
Adding Classes	191

---

Class Diagrams	192
<b>Class Library Projects</b>	<b>193</b>
<b>Interfaces Versus Abstract Classes</b>	<b>196</b>
<b>Struct Types</b>	<b>198</b>
<b>Shallow Copying Versus Deep Copying</b>	<b>200</b>
<b>Summary</b>	<b>200</b>
<b>CHAPTER 10: DEFINING CLASS MEMBERS</b>	<b>203</b>
<hr/>	
<b>Member Definitions</b>	<b>203</b>
Defining Fields	204
Defining Methods	204
Defining Properties	205
Adding Members from a Class Diagram	210
Adding Methods	210
Adding Properties	212
Adding Fields	212
Refactoring Members	212
Automatic Properties	213
<b>Additional Class Member Topics</b>	<b>213</b>
Hiding Base Class Methods	213
Calling Overridden or Hidden Base Class Methods	215
The this Keyword	216
Using Nested Type Definitions	216
<b>Interface Implementation</b>	<b>218</b>
Implementing Interfaces in Classes	219
Explicit Interface Member Implementation	220
Additional Property Accessors	220
<b>Partial Class Definitions</b>	<b>221</b>
<b>Partial Method Definitions</b>	<b>222</b>
<b>Example Application</b>	<b>224</b>
Planning the Application	224
The Card Class	224
The Deck Class	224
Writing the Class Library	224
Adding the Suit and Rank Enumerations	225
Adding the Card Class	226
Adding the Deck Class	227
A Client Application for the Class Library	230
<b>The Call Hierarchy Window</b>	<b>231</b>
<b>Summary</b>	<b>232</b>

---

<b>CHAPTER 11: COLLECTIONS, COMPARISONS, AND CONVERSIONS</b>	<b>235</b>
<b>Collections</b>	<b>236</b>
Using Collections	236
Defining Collections	241
Indexers	242
Adding a Cards Collection to CardLib	244
Keyed Collections and IDictionary	247
Iterators	248
Iterators and Collections	252
Deep Copying	253
Adding Deep Copying to CardLib	255
<b>Comparisons</b>	<b>256</b>
Type Comparisons	256
Boxing and Unboxing	257
The is Operator	258
Value Comparisons	260
Operator Overloading	261
Adding Operator Overloads to CardLib	265
The IComparable and IComparer Interfaces	269
Sorting Collections	271
<b>Conversions</b>	<b>274</b>
Overloading Conversion Operators	274
The as Operator	275
<b>Summary</b>	<b>276</b>
<b>CHAPTER 12: GENERICS</b>	<b>279</b>
<b>What Are Generics?</b>	<b>280</b>
<b>Using Generics</b>	<b>281</b>
<b>Nullable Types</b>	<b>281</b>
Operators and Nullable Types	282
The ?? Operator	283
<b>The System.Collections.Generic Namespace</b>	<b>287</b>
List<T>	287
Sorting and Searching Generic Lists	289
Dictionary<K, V>	294
Modifying CardLib to Use a Generic Collection Class	295
<b>Defining Generic Types</b>	<b>296</b>
<b>Defining Generic Classes</b>	<b>296</b>
The default Keyword	298

---

Constraining Types	298
Inheriting from Generic Classes	304
Generic Operators	305
Generic Structs	306
Defining Generic Interfaces	306
Defining Generic Methods	306
Defining Generic Delegates	308
<b>Variance</b>	<b>308</b>
Covariance	309
Contravariance	310
<b>Summary</b>	<b>310</b>
<b>CHAPTER 13: ADDITIONAL OOP TECHNIQUES</b>	<b>313</b>
<hr/>	
<b>The :: Operator and the Global Namespace Qualifier</b>	<b>313</b>
<b>Custom Exceptions</b>	<b>315</b>
Adding Custom Exceptions to CardLib	315
<b>Events</b>	<b>316</b>
What Is an Event?	316
Handling Events	317
Defining Events	319
Multipurpose Event Handlers	323
The EventHandler and Generic EventHandler<T> Types	325
Return Values and Event Handlers	326
Anonymous Methods	326
<b>Expanding and Using CardLib</b>	<b>327</b>
A Card Game Client for CardLib	327
<b>Attributes</b>	<b>334</b>
Reading Attributes	334
Creating Attributes	335
<b>Summary</b>	<b>336</b>
<b>CHAPTER 14: C# LANGUAGE ENHANCEMENTS</b>	<b>339</b>
<hr/>	
<b>Initializers</b>	<b>340</b>
Object Initializers	340
Collection Initializers	342
<b>Type Inference</b>	<b>344</b>
<b>Anonymous Types</b>	<b>346</b>
<b>Dynamic Lookup</b>	<b>350</b>
The dynamic Type	350
IDynamicMetaObjectProvider	354

<b>Advanced Method Parameters</b>	<b>354</b>
Optional Parameters	354
Optional Parameter Values	355
The OptionalAttribute Attribute	356
Optional Parameter Order	356
Named Parameters	356
Named and Optional Parameter Guidelines	360
<b>Extension Methods</b>	<b>360</b>
<b>Lambda Expressions</b>	<b>364</b>
Anonymous Methods Recap	364
Lambda Expressions for Anonymous Methods	365
Lambda Expression Parameters	368
Lambda Expression Statement Bodies	368
Lambda Expressions as Delegates and Expression Trees	369
Lambda Expressions and Collections	370
<b>Caller Information Attributes</b>	<b>372</b>
<b>Summary</b>	<b>375</b>

## PART II: WINDOWS PROGRAMMING

### CHAPTER 15: BASIC DESKTOP PROGRAMMING 381

<b>XAML</b>	<b>382</b>
Separation of Concerns	382
XAML in Action	383
Namespaces	383
Code-Behind Files	384
<b>The Playground</b>	<b>384</b>
WPF Controls	385
Properties	386
Dependency Properties	387
Attached Properties	388
Events	388
Handling Events	389
Routed Events	390
Routed Commands	390
Control Types	392
<b>Control Layout</b>	<b>393</b>
Stack Order	393
Alignment, Margins, Padding, and Dimensions	393
Border	394
Canvas	394



---

DockPanel	395
StackPanel	397
WrapPanel	398
Grid	398
<b>The Game Client</b>	<b>401</b>
The About Window	401
Designing the User Interface	401
The Image Control	401
The Label Control	402
The TextBlock Control	402
The Button Control	402
The Options Window	405
The TextBox Control	406
The CheckBox Control	406
The RadioButton Control	407
The ComboBox Control	408
The TabControl	409
Handling Events in the Options Window	411
Data Binding	413
The DataContext	414
Binding to Local Objects	414
Static Binding to External Objects	414
Dynamic Binding to External Objects	415
Starting a Game	418
The ListBox Control	418
<b>Summary</b>	<b>421</b>
<b>CHAPTER 16: ADVANCED DESKTOP PROGRAMMING</b>	<b>425</b>
<b>The Main Window</b>	<b>425</b>
The Menu Control	426
Routed Commands with Menus	426
<b>Creating and Styling Controls</b>	<b>429</b>
Styles	430
Templates	430
Value Converters	434
The IValueConverter Interface	434
ValueConversionAttribute	434
Triggers	435
Animations	436
Timelines without Key Frames	436
Timelines with Key Frames	437

<b>WPF User Controls</b>	<b>438</b>
Implementing Dependency Properties	439
<b>Putting It All Together</b>	<b>447</b>
Refactoring the Domain Model	447
The View Models	451
Completing the Game	459
<b>Summary</b>	<b>466</b>
<b>CHAPTER 17: WINDOWS STORE APPS</b>	<b>469</b>
<hr/>	
<b>Getting Started</b>	<b>469</b>
Windows 8 Developer License	470
How to Get the License	470
<b>Windows Store Apps versus Desktop Applications</b>	<b>471</b>
<b>Developing Windows Store Apps</b>	<b>472</b>
View Modes	472
Full-Screen Mode	473
Snapped Mode	473
Fill Mode	473
Screen Orientation	473
The Missing Caption Bar	473
Menus and Toolbars	473
The App Bar	474
Charms	475
Tiles and Badges	476
App Lifetime	476
Lock Screen Apps	476
<b>App Development</b>	<b>476</b>
WPF and Windows Store App XAML Differences	476
Namespace Changes	477
Effects	477
Opacity Mask	477
Style Triggers	477
Commands	478
Templates and Pages	478
Blank Page	478
Basic Page	478
Grouped Items and Group Details Pages	478
Item Details Page	479
Items and Split Pages	479
Sandboxed Apps	479
Disk Access	479
Serialization, Streams, and Async Programming	480

---

Navigation Between Pages	483
Managing State	485
<b>Converting KarliCards, Part 1</b>	<b>485</b>
Creating the CardLib Project	486
Converting the View Models	489
Visual Changes	493
Drop Shadow and Opacity Masks	493
Style Triggers	493
Converting User Controls	493
<b>Common Elements of Windows Store Apps</b>	<b>499</b>
The AppBar Control	500
The Settings Panel	501
Tiles, Badges, and Splash Screens	503
<b>Converting KarliCards, Part 2</b>	<b>504</b>
<b>The Windows Store</b>	<b>509</b>
Checking the Store Requirements	509
<b>Summary</b>	<b>510</b>
<b>CHAPTER 18: DEPLOYING DESKTOP APPLICATIONS</b>	<b>513</b>
<hr/>	
<b>Deployment Overview</b>	<b>514</b>
<b>ClickOnce Deployment</b>	<b>514</b>
Implementing ClickOnce Deployment	515
Installing the Application with ClickOnce	522
Creating and Using Updates of the Application	524
<b>InstallShield Limited Edition</b>	<b>524</b>
The Project Assistant	524
Step 1: Application Information	525
Step 2: Installation Requirements	525
Step 3: Installation Architecture	526
Step 4: Application Files	526
Step 5: Application Shortcuts	527
Step 6: Application Registry	528
Step 7: Installation Interview	528
<b>Summary</b>	<b>530</b>
<b>PART III: WEB PROGRAMMING</b>	
<hr/>	
<b>CHAPTER 19: ASP.NET WEB PROGRAMMING</b>	<b>535</b>
<hr/>	
Overview of Web Applications	536
ASP.NET Runtime	536
Creating a Simple Page	536

---

<b>Server Controls</b>	<b>544</b>
<b>ASP.NET Postback</b>	<b>545</b>
<b>ASP.NET AJAX Postback</b>	<b>550</b>
<b>Input Validation</b>	<b>553</b>
<b>State Management</b>	<b>557</b>
Client-Side State Management	558
View State	558
Cookie	559
Server-Side State Management	560
Session	560
Application	561
Cache	562
<b>Styles</b>	<b>562</b>
<b>Master Pages</b>	<b>566</b>
<b>Site Navigation</b>	<b>571</b>
<b>Authentication and Authorization</b>	<b>573</b>
Authentication Configuration	574
Using Security Controls	577
<b>Reading from and Writing to a SQL Server Database</b>	<b>580</b>
<b>Summary</b>	<b>589</b>
<b>CHAPTER 20: DEPLOYING WEB APPLICATIONS</b>	<b>591</b>
<hr/>	
<b>Internet Information Services</b>	<b>591</b>
<b>IIS Configuration</b>	<b>592</b>
<b>Copying a Website</b>	<b>594</b>
<b>Publishing a Web Application</b>	<b>597</b>
<b>Summary</b>	<b>600</b>
<hr/>	
<b>PART IV: DATA ACCESS</b>	
<hr/>	
<b>CHAPTER 21: FILE SYSTEM DATA</b>	<b>605</b>
<hr/>	
<b>Streams</b>	<b>605</b>
<b>The Classes for Input and Output</b>	<b>606</b>
The File and Directory Classes	607
The FileInfo Class	608
The DirectoryInfo Class	610
Path Names and Relative Paths	611
The FileStream Object	611
File Position	612
Reading Data	613
Writing Data	615

---

The StreamWriter Object	617
The StreamReader Object	618
Reading Data	620
Delimited Files	621
Asynchronous File Access	625
Reading and Writing Compressed Files	625
<b>Serialized Objects</b>	<b>628</b>
<b>Monitoring the File System</b>	<b>632</b>
<b>Summary</b>	<b>636</b>
<b>CHAPTER 22: XML</b>	<b>639</b>
<b>XML Documents</b>	<b>640</b>
XML Elements	640
Attributes	641
The XML Declaration	641
The Structure of an XML Document	642
XML Namespaces	642
Well-Formed and Valid XML	643
Validating XML Documents	644
Schemas	644
The XSD dialog box shown in the XmlDocument Class	645
<b>Using XML in Your Application</b>	<b>647</b>
XML Document Object Model	647
The XmlDocument Class	648
The XmlElement Class	648
Changing the Values of Nodes	652
Selecting Nodes	656
XPath	657
<b>Summary</b>	<b>661</b>
<b>CHAPTER 23: INTRODUCTION TO LINQ</b>	<b>663</b>
<b>First LINQ Query</b>	<b>664</b>
Declaring a Variable for Results Using the var Keyword	665
Specifying the Data Source: from Clause	666
Specify Condition: where Clause	666
Selecting Items: select Clause	666
Finishing Up: Using the foreach Loop	667
Deferred Query Execution	667
<b>Using the LINQ Method Syntax</b>	<b>667</b>
LINQ Extension Methods	667
Query Syntax versus Method Syntax	668

Ordering Query Results	669
Understanding the orderby Clause	670
Ordering Using Method Syntax	671
Querying a Large Data Set	672
Using Aggregate Operators	674
Querying Complex Objects	678
Projection: Creating New Objects in Queries	681
Projection: Method Syntax	682
Using the Select Distinct Query	683
Using the Any and All Methods	684
Ordering by Multiple Levels	685
Multi-Level Ordering Method Syntax: ThenBy	687
Using Group Queries	687
Using Take and Skip	689
Using First and FirstOrDefault	691
Using the LINQ Set Operators	692
Using Joins	694
Summary	696
<b>CHAPTER 24: APPLYING LINQ</b>	<b>699</b>
<b>LINQ Varieties</b>	<b>699</b>
Using LINQ with Databases	700
Installing SQL Server and the Northwind Sample Data	700
Installing SQL Server Express	701
Installing the Northwind Sample Database	701
First LINQ to Database Query	701
Navigating Database Relationships	704
Using LINQ with XML	706
LINQ to XML Functional Constructors	707
Constructing XML Element Text with Strings	710
Saving and Loading an XML Document	710
Loading XML from a String	712
Contents of a Saved XML Document	713
Working with XML Fragments	713
Generating XML from Databases	715
How to Query an XML Document	717
Using LINQ to XML Query Members	718
Elements()	718
Descendants()	719
Attributes()	721
Summary	723