

# Contents

<b>Preface</b>	<b>xxiii</b>
<b>Before You Begin</b>	<b>xli</b>
<b>1 Introduction to Computers, the Internet and Visual C#</b>	<b>1</b>
1.1 Introduction	2
1.2 What Is a Computer?	3
1.3 Computer Organization	3
1.4 Personal Computing, Distributed Computing and Client/Server Computing	4
1.5 Hardware Trends	5
1.6 Microsoft's Windows® Operating System	6
1.7 Machine Languages, Assembly Languages and High-Level Languages	6
1.8 Visual Basic	8
1.9 C, C++ and Java	8
1.10 Visual C#	9
1.11 Other High-Level Languages	10
1.12 Structured Programming	10
1.13 Key Software Trend: Object Technology	11
1.14 The Internet and the World Wide Web	12
1.15 Extensible Markup Language (XML)	13
1.16 Introduction to Microsoft .NET	13
1.17 The .NET Framework and the Common Language Runtime	14
1.18 Test-Driving a C# <b>Advanced Painter</b> Application	15
1.19 (Only Required Section of the Case Study) Software Engineering Case Study: Introduction to Object Technology and the UML	18
1.20 Wrap-Up	23
1.21 Web Resources	23
<b>2 Dive Into® Visual C# 2008 Express</b>	<b>34</b>
2.1 Introduction	35
2.2 Overview of the Visual Studio 2008 IDE	35
2.3 Menu Bar and Toolbar	41
2.4 Navigating the Visual Studio IDE	44
2.4.1 <b>Solution Explorer</b>	46
2.4.2 <b>Toolbox</b>	47
2.4.3 <b>Properties</b> Window	48

**x** Contents

2.5	Using Help	49
2.6	Using Visual Programming to Create a Simple Program that Displays Text and an Image	51
2.7	Wrap-Up	63
2.8	Web Resources	64

**3 Introduction to C# Applications 73**

3.1	Introduction	74
3.2	A Simple C# Application: Displaying a Line of Text	74
3.3	Creating a Simple Application in Visual C# Express	80
3.4	Modifying Your Simple C# Application	88
3.5	Formatting Text with <code>Console.Write</code> and <code>Console.WriteLine</code>	90
3.6	Another C# Application: Adding Integers	91
3.7	Memory Concepts	95
3.8	Arithmetic	96
3.9	Decision Making: Equality and Relational Operators	99
3.10	(Optional) Software Engineering Case Study: Examining the ATM Requirements Document	104
3.11	Wrap-Up	113
3.12	Web Resources	113

**4 Introduction to Classes and Objects 124**

4.1	Introduction	125
4.2	Classes, Objects, Methods, Properties and Instance Variables	125
4.3	Declaring a Class with a Method and Instantiating an Object of a Class	127
4.4	Declaring a Method with a Parameter	131
4.5	Instance Variables and Properties	134
4.6	UML Class Diagram with a Property	140
4.7	Software Engineering with Properties and set and get Accessors	140
4.8	Auto-Implemented Properties	142
4.9	Value Types vs. Reference Types	143
4.10	Initializing Objects with Constructors	145
4.11	Floating-Point Numbers and Type <code>decimal</code>	148
4.12	(Optional) Software Engineering Case Study: Identifying the Classes in the ATM Requirements Document	154
4.13	Wrap-Up	161

**5 Control Statements: Part I 169**

5.1	Introduction	170
5.2	Algorithms	170
5.3	Pseudocode	171
5.4	Control Structures	171
5.5	<code>if</code> Single-Selection Statement	174

5.6	<code>if...else</code> Double-Selection Statement	175
5.7	<code>while</code> Repetition Statement	180
5.8	Formulating Algorithms: Counter-Controlled Repetition	181
5.9	Formulating Algorithms: Sentinel-Controlled Repetition	186
5.10	Formulating Algorithms: Nested Control Statements	194
5.11	Compound Assignment Operators	198
5.12	Increment and Decrement Operators	199
5.13	Simple Types	202
5.14	(Optional) Software Engineering Case Study: Identifying Class Attributes in the ATM System	203
5.15	Wrap-Up	207

## **6 Control Statements: Part 2 221**

6.1	Introduction	222
6.2	Essentials of Counter-Controlled Repetition	222
6.3	<code>for</code> Repetition Statement	224
6.4	Examples Using the <code>for</code> Statement	228
6.5	<code>do...while</code> Repetition Statement	233
6.6	<code>switch</code> Multiple-Selection Statement	235
6.7	<code>break</code> and <code>continue</code> Statements	242
6.8	Logical Operators	244
6.9	Structured-Programming Summary	250
6.10	(Optional) Software Engineering Case Study: Identifying Objects’ States and Activities in the ATM System	255
6.11	Wrap-Up	260

## **7 Methods: A Deeper Look 270**

7.1	Introduction	271
7.2	Packaging Code in <i>C#</i>	272
7.3	<code>static</code> Methods, <code>static</code> Variables and Class <code>Math</code>	273
7.4	Declaring Methods with Multiple Parameters	276
7.5	Notes on Declaring and Using Methods	280
7.6	Method-Call Stack and Activation Records	281
7.7	Argument Promotion and Casting	282
7.8	The .NET Framework Class Library	284
7.9	Case Study: Random-Number Generation	285
	7.9.1 Scaling and Shifting Random Numbers	289
	7.9.2 Random-Number Repeatability for Testing and Debugging	290
7.10	Case Study: A Game of Chance (Introducing Enumerations)	291
7.11	Scope of Declarations	296
7.12	Method Overloading	299
7.13	Recursion	302
7.14	Passing Arguments: Pass-by-Value vs. Pass-by-Reference	305

7.15	(Optional) Software Engineering Case Study: Identifying Class Operations in the ATM System	308
7.16	Wrap-Up	316

## **8 Arrays 332**

8.1	Introduction	333
8.2	Arrays	333
8.3	Declaring and Creating Arrays	335
8.4	Examples Using Arrays	336
8.5	Case Study: Card Shuffling and Dealing Simulation	345
8.6	foreach Statement	349
8.7	Passing Arrays and Array Elements to Methods	351
8.8	Passing Arrays by Value and by Reference	353
8.9	Case Study: Class GradeBook Using an Array to Store Grades	357
8.10	Multidimensional Arrays	363
8.11	Case Study: Class GradeBook Using a Rectangular Array	368
8.12	Variable-Length Argument Lists	373
8.13	Using Command-Line Arguments	375
8.14	(Optional) Software Engineering Case Study: Collaboration Among Objects in the ATM System	377
8.15	Wrap-Up	384

## **9 Introduction to LINQ and Generic Collections 405**

9.1	Introduction	406
9.2	Querying an Array Using LINQ	407
9.3	Introduction to Collections	416
9.4	Querying a Generic Collection Using LINQ	419
9.5	Wrap-Up	421
9.6	Deitel LINQ Resource Center	421

## **10 Classes and Objects: A Deeper Look 426**

10.1	Introduction	427
10.2	Time Class Case Study	428
10.3	Controlling Access to Members	432
10.4	Referring to the Current Object's Members with the this Reference	433
10.5	Indexers	435
10.6	Time Class Case Study: Overloaded Constructors	438
10.7	Default and Parameterless Constructors	444
10.8	Composition	445
10.9	Garbage Collection and Destructors	448
10.10	static Class Members	449
10.11	readonly Instance Variables	453
10.12	Software Reusability	455

10.13	Data Abstraction and Encapsulation	456
10.14	Time Class Case Study: Creating Class Libraries	458
10.15	internal Access	462
10.16	<b>Class View</b> and <b>Object Browser</b>	464
10.17	Object Initializers	465
10.18	Time Class Case Study: Extension Methods	468
10.19	Delegates	471
10.20	Lambda Expressions	474
10.21	Anonymous Types	477
10.22	(Optional) Software Engineering Case Study: Starting to Program the Classes of the ATM System	479
10.23	Wrap-Up	485

## **11 Object-Oriented Programming: Inheritance 496**

11.1	Introduction	497
11.2	Base Classes and Derived Classes	498
11.3	protected Members	500
11.4	Relationship between Base Classes and Derived Classes	501
11.4.1	Creating and Using a <code>CommissionEmployee</code> Class	502
11.4.2	Creating a <code>BasePlusCommissionEmployee</code> Class without Using Inheritance	507
11.4.3	Creating a <code>CommissionEmployee–BasePlusCommissionEmployee</code> Inheritance Hierarchy	511
11.4.4	<code>CommissionEmployee–BasePlusCommissionEmployee</code> Inheritance Hierarchy Using protected Instance Variables	514
11.4.5	<code>CommissionEmployee–BasePlusCommissionEmployee</code> Inheritance Hierarchy Using private Instance Variables	520
11.5	Constructors in Derived Classes	525
11.6	Software Engineering with Inheritance	531
11.7	Class object	532
11.8	Wrap-Up	533

## **12 Polymorphism, Interfaces and Operator Overloading 539**

12.1	Introduction	540
12.2	Polymorphism Examples	542
12.3	Demonstrating Polymorphic Behavior	543
12.4	Abstract Classes and Methods	546
12.5	Case Study: Payroll System Using Polymorphism	548
12.5.1	Creating Abstract Base Class <code>Employee</code>	549
12.5.2	Creating Concrete Derived Class <code>SalariedEmployee</code>	551
12.5.3	Creating Concrete Derived Class <code>HourlyEmployee</code>	553
12.5.4	Creating Concrete Derived Class <code>CommissionEmployee</code>	554

12.5.5	Creating Indirect Concrete Derived Class BasePlusCommissionEmployee	556
12.5.6	Polymorphic Processing, Operator <code>is</code> and Downcasting	557
12.5.7	Summary of the Allowed Assignments Between Base-Class and Derived-Class Variables	562
12.6	sealed Methods and Classes	563
12.7	Case Study: Creating and Using Interfaces	564
12.7.1	Developing an IPayable Hierarchy	565
12.7.2	Declaring Interface IPayable	566
12.7.3	Creating Class Invoice	566
12.7.4	Modifying Class Employee to Implement Interface IPayable	568
12.7.5	Modifying Class SalariedEmployee for Use with IPayable	569
12.7.6	Using Interface IPayable to Process Invoices and Employees Polymorphically	571
12.7.7	Common Interfaces of the .NET Framework Class Library	573
12.8	Operator Overloading	574
12.9	(Optional) Software Engineering Case Study: Incorporating Inheritance and Polymorphism into the ATM System	577
12.10	Wrap-Up	586

## **13 Exception Handling** **591**

13.1	Introduction	592
13.2	Exception-Handling Overview	593
13.3	Example: Divide by Zero without Exception Handling	593
13.4	Example: Handling DivideByZeroExceptions and FormatExceptions	596
13.4.1	Enclosing Code in a try Block	599
13.4.2	Catching Exceptions	599
13.4.3	Uncaught Exceptions	599
13.4.4	Termination Model of Exception Handling	600
13.4.5	Flow of Control When Exceptions Occur	601
13.5	.NET Exception Hierarchy	601
13.5.1	Class SystemException	602
13.5.2	Determining Which Exceptions a Method Throws	602
13.6	finally Block	603
13.7	Exception Properties	611
13.8	User-Defined Exception Classes	616
13.9	Wrap-Up	619

## **14 Graphical User Interfaces with Windows Forms: Part I** **624**

14.1	Introduction	625
14.2	Windows Forms	626
14.3	Event Handling	629
14.3.1	A Simple Event-Driven GUI	629

14.3.2	Another Look at the Visual Studio Generated Code	631
14.3.3	Delegates and the Event-Handling Mechanism	632
14.3.4	Other Ways to Create Event Handlers	633
14.3.5	Locating Event Information	634
14.4	Control Properties and Layout	636
14.5	Labels, TextBoxes and Buttons	639
14.6	GroupBoxes and Panels	642
14.7	CheckBoxes and RadioButtons	646
14.8	PictureBoxes	654
14.9	ToolTips	657
14.10	NumericUpDown Control	659
14.11	Mouse-Event Handling	661
14.12	Keyboard-Event Handling	664
14.13	Wrap-Up	667

## **15 Graphical User Interfaces with Windows Forms: Part 2** **677**

15.1	Introduction	678
15.2	Menus	678
15.3	MonthCalendar Control	688
15.4	DateTimePicker Control	689
15.5	LinkLabel Control	692
15.6	ListBox Control	696
15.7	CheckedListBox Control	700
15.8	ComboBox Control	703
15.9	TreeView Control	707
15.10	Listview Control	712
15.11	TabControl Control	718
15.12	Multiple Document Interface (MDI) Windows	723
15.13	Visual Inheritance	731
15.14	User-Defined Controls	734
15.15	Wrap-Up	739

## **16 GUI with Windows Presentation Foundation** **747**

16.1	Introduction	748
16.2	Windows Presentation Foundation (WPF)	749
16.3	XML Basics	750
16.4	Structuring Data	753
16.5	XML Namespaces	758
16.6	Declarative GUI Programming Using XAML	762
16.7	Creating a WPF Application in Visual C# Express	764
16.8	Laying Out Controls	766
16.8.1	General Layout Principles	766
16.8.2	Layout in Action	767

16.9	Event Handling	772
16.10	Commands and Common Application Tasks	779
16.11	WPF GUI Customization	784
16.12	Using Styles to Change a Control's Appearance	785
16.13	Customizing Windows	790
16.14	Defining a Control's Appearance with Control Templates	793
16.15	Data-Driven GUIs with Data Binding	798
16.16	Wrap-Up	804
16.17	Web Resources	804

## **17 WPF Graphics and Multimedia 818**

17.1	Introduction	819
17.2	Controlling Fonts	819
17.3	Basic Shapes	821
17.4	Polygons and PolyLines	823
17.5	Brushes	826
17.6	Transforms	832
17.7	WPF Customization: A Television GUI	835
17.8	Animations	844
17.9	(Optional) 3-D Objects and Transforms	847
17.10	Wrap-Up	854

## **18 Strings, Characters and Regular Expressions 862**

18.1	Introduction	863
18.2	Fundamentals of Characters and Strings	864
18.3	string Constructors	865
18.4	string Indexer, Length Property and CopyTo Method	866
18.5	Comparing strings	867
18.6	Locating Characters and Substrings in strings	871
18.7	Extracting Substrings from strings	873
18.8	Concatenating strings	874
18.9	Miscellaneous string Methods	875
18.10	Class StringBuilder	876
18.11	Length and Capacity Properties, EnsureCapacity Method and Indexer of Class StringBuilder	878
18.12	Append and AppendFormat Methods of Class StringBuilder	879
18.13	Insert, Remove and Replace Methods of Class StringBuilder	882
18.14	Char Methods	884
18.15	Card Shuffling and Dealing Simulation	887
18.16	Introduction to Regular-Expression Processing	891
18.16.1	Simple Regular Expressions and Class Regex	892
18.16.2	Complex Regular Expressions	897
18.16.3	Validating User Input with Regular Expressions and LINQ	898
18.16.4	Regex Methods Replace and Split	903
18.17	Wrap-Up	905



<b>19</b>	<b>Files and Streams</b>	<b>912</b>
19.1	Introduction	913
19.2	Data Hierarchy	913
19.3	Files and Streams	915
19.4	Classes File and Directory	916
19.5	Creating a Sequential-Access Text File	925
19.6	Reading Data from a Sequential-Access Text File	935
19.7	Case Study: Credit Inquiry Program Using LINQ	940
19.8	Serialization	946
19.9	Creating a Sequential-Access File Using Object Serialization	946
19.10	Reading and Deserializing Data from a Binary File	953
19.11	Wrap-Up	957
<b>20</b>	<b>XML and LINQ to XML</b>	<b>964</b>
20.1	Introduction	965
20.2	Document Type Definitions (DTDs)	965
20.3	W3C XML Schema Documents	969
20.4	Extensible Stylesheet Language and XSL Transformations	976
20.5	LINQ to XML: Document Object Model (DOM)	985
20.6	LINQ to XML Class Hierarchy	989
20.7	LINQ to XML: Namespaces and Creating Documents	998
20.8	XSLT with Class XsltCompiledTransform	1001
20.9	Wrap-Up	1003
20.10	Web Resources	1004
<b>21</b>	<b>Databases and LINQ to SQL</b>	<b>1011</b>
21.1	Introduction	1012
21.2	Relational Databases	1013
21.3	Relational Database Overview: Books Database	1014
21.4	SQL	1018
21.4.1	Basic SELECT Query	1018
21.4.2	WHERE Clause	1019
21.4.3	ORDER BY Clause	1021
21.4.4	Retrieving Data from Multiple Tables: INNER JOIN	1023
21.4.5	INSERT Statement	1024
21.4.6	UPDATE Statement	1025
21.4.7	DELETE Statement	1026
21.5	LINQ to SQL	1027
21.6	LINQ to SQL: Extracting Information from a Database	1028
21.6.1	Creating LINQ to SQL Classes	1028
21.6.2	Creating Data Bindings	1029
21.7	More Complex LINQ Queries and Data Binding	1032
21.8	Retrieving Data from Multiple Tables with LINQ	1037

21.9	Creating a Master/Detail View Application	1040
21.10	Programming with LINQ to SQL: Address-Book Case Study	1045
21.11	Wrap-Up	1051
21.12	Tools and Web Resources	1051

## **22 ASP.NET 3.5 and ASP.NET AJAX 1060**

22.1	Introduction	1061
22.2	Simple HTTP Transactions	1062
22.3	Multitier Application Architecture	1066
22.4	Creating and Running a Simple Web-Form Example	1067
22.4.1	Examining an ASPX File	1067
22.4.2	Examining a Code-Behind File	1069
22.4.3	Relationship Between an ASPX File and a Code-Behind File	1070
22.4.4	How the Code in an ASP.NET Web Page Executes	1071
22.4.5	Examining the XHTML Generated by an ASP.NET Application	1071
22.4.6	Building an ASP.NET Web Application	1073
22.5	Web Controls	1081
22.5.1	Text and Graphics Controls	1082
22.5.2	AdRotator Control	1088
22.5.3	Validation Controls	1091
22.6	Session Tracking	1098
22.6.1	Cookies	1099
22.6.2	Session Tracking with <code>HttpSessionState</code>	1107
22.7	Case Study: Connecting to a Database in ASP.NET	1114
22.7.1	Building a Web Form That Displays Data from a Database	1115
22.7.2	Modifying the Code-Behind File for the Guestbook Application	1122
22.8	Case Study: Secure Books Database Application	1123
22.8.1	Examining the Completed Secure Books Database Application	1124
22.8.2	Creating the Secure Books Database Application	1127
22.9	ASP.NET AJAX	1151
22.9.1	Traditional Web Applications	1151
22.9.2	Ajax Web Applications	1152
22.9.3	Examining an ASP.NET AJAX Application	1152
22.10	New ASP.NET 3.5 Data Controls	1159
22.11	Wrap-Up	1160
22.12	Web Resources	1161

## **23 Windows Communication Foundation (WCF) Web Services 1172**

23.1	Introduction	1173
23.2	WCF Services Basics	1174
23.3	Simple Object Access Protocol (SOAP)	1174
23.4	Representational State Transfer (REST)	1175

23.5	JavaScript Object Notation (JSON)	1175
23.6	Publishing and Consuming SOAP-Based Web Services	1176
23.6.1	Creating a WCF Web Service	1176
23.6.2	Code for the <code>WelcomeSOAPXMLService</code>	1176
23.6.3	Building a SOAP-Based Web Service	1177
23.6.4	Deploying the <code>WelcomeSOAPXMLService</code>	1180
23.6.5	Creating a Client to Consume the <code>WelcomeSOAPXMLService</code>	1182
23.6.6	Consuming the <code>WelcomeSOAPXMLService</code>	1184
23.7	Publishing and Consuming REST-Based XML Web Services	1185
23.7.1	Creating a REST-Based XML Web Service	1185
23.7.2	Consuming a REST-Based XML Web Service	1188
23.8	Publishing and Consuming REST-Based JSON Web Services	1189
23.8.1	Creating a REST-Based JSON Web Service	1189
23.8.2	Consuming a REST-Based JSON Web Service	1191
23.9	Blackjack Web Service: Using Session Tracking in a SOAP-Based Web Service	1193
23.9.1	Creating a Blackjack Web Service	1193
23.9.2	Consuming the Blackjack Web Service	1197
23.10	Airline Reservation Web Service: Database Access and Invoking a Service from ASP.NET	1206
23.11	Equation Generator: Returning User-Defined Types	1211
23.11.1	Creating the REST-Based XML <b>EquationGenerator</b> Web Service	1214
23.11.2	Consuming the REST-Based XML <b>EquationGenerator</b> Web Service	1215
23.11.3	Creating the REST-Based JSON <code>EquationGenerator</code> Web Service	1219
23.11.4	Consuming the REST-Based JSON <code>EquationGenerator</code> Web Service	1220
23.12	Wrap-Up	1223
23.13	Deitel Web Services Resource Centers	1224

## **24 Silverlight, Rich Internet Applications and Multimedia** **1232**

24.1	Introduction	1233
24.2	Platform Overview	1234
24.3	Silverlight Runtime and Tools Installation	1235
24.4	Building a Silverlight <b>WeatherViewer</b> Application	1235
24.4.1	GUI Layout	1238
24.4.2	Obtaining and Displaying Weather Forecast Data	1240
24.4.3	Custom Controls	1244
24.5	Animations and the <b>FlickrViewer</b>	1247
24.6	Images and Deep Zoom	1253
24.6.1	Getting Started With Deep Zoom Composer	1256
24.6.2	Creating a Silverlight Deep Zoom Application	1258

24.7	Audio and Video	1266
24.8	Isolated Storage	1271
24.9	Silverlight Demos and Web Resources	1272
24.10	Wrap-Up	1273

## **25 Searching and Sorting 1281**

25.1	Introduction	1282
25.2	Searching Algorithms	1283
25.2.1	Linear Search	1283
25.2.2	Binary Search	1287
25.3	Sorting Algorithms	1292
25.3.1	Selection Sort	1292
25.3.2	Insertion Sort	1297
25.3.3	Merge Sort	1301
25.4	Wrap-Up	1307

## **26 Data Structures 1313**

26.1	Introduction	1314
26.2	Simple-Type structs, Boxing and Unboxing	1314
26.3	Self-Referential Classes	1315
26.4	Linked Lists	1316
26.5	Stacks	1329
26.6	Queues	1333
26.7	Trees	1336
26.7.1	Binary Search Tree of Integer Values	1337
26.7.2	Binary Search Tree of IComparable Objects	1344
26.8	Wrap-Up	1349

## **27 Generics 1356**

27.1	Introduction	1357
27.2	Motivation for Generic Methods	1358
27.3	Generic-Method Implementation	1360
27.4	Type Constraints	1362
27.5	Overloading Generic Methods	1365
27.6	Generic Classes	1366
27.7	Wrap-Up	1375

## **28 Collections 1381**

28.1	Introduction	1382
28.2	Collections Overview	1383
28.3	Class Array and Enumerators	1385

28.4	Nongeneric Collections	1388
28.4.1	Class ArrayList	1389
28.4.2	Class Stack	1393
28.4.3	Class Hashtable	1395
28.5	Generic Collections	1400
28.5.1	Generic Class SortedDictionary	1400
28.5.2	Generic Class LinkedList	1403
28.6	Wrap-Up	1407

## **A Operator Precedence Chart 1413**

## **B Simple Types 1415**

## **C Number Systems 1417**

C.1	Introduction	1418
C.2	Abbreviating Binary Numbers as Octal and Hexadecimal Numbers	1421
C.3	Converting Octal and Hexadecimal Numbers to Binary Numbers	1422
C.4	Converting from Binary, Octal or Hexadecimal to Decimal	1422
C.5	Converting from Decimal to Binary, Octal or Hexadecimal	1423
C.6	Negative Binary Numbers: Two's Complement Notation	1425

## **D ATM Case Study Code 1430**

D.1	ATM Case Study Implementation	1430
D.2	Class ATM	1431
D.3	Class Screen	1437
D.4	Class Keypad	1437
D.5	Class CashDispenser	1438
D.6	Class DepositSlot	1439
D.7	Class Account	1440
D.8	Class BankDatabase	1442
D.9	Class Transaction	1445
D.10	Class BalanceInquiry	1447
D.11	Class Withdrawal	1448
D.12	Class Deposit	1452
D.13	Class ATMCASEStudy	1454
D.14	Wrap-Up	1454

## **E UML 2: Additional Diagram Types 1456**

E.1	Introduction	1456
E.2	Additional Diagram Types	1456

<b>F</b>	<b>ASCII Character Set</b>	<b>1458</b>
<b>G</b>	<b>Unicode®</b>	<b>1459</b>
G.1	Introduction	1460
G.2	Unicode Transformation Formats	1461
G.3	Characters and Glyphs	1462
G.4	Advantages/Disadvantages of Unicode	1462
G.5	Using Unicode	1463
G.6	Character Ranges	1465
<b>H</b>	<b>Using the Visual C# 2008 Debugger</b>	<b>1470</b>
H.1	Introduction	1471
H.2	Breakpoints and the <b>Continue</b> Command	1471
H.3	<i>DataTips</i> and Visualizers	1477
H.4	The <b>Locals</b> and <b>Watch</b> Windows	1478
H.5	Controlling Execution Using the <b>Step Into</b> , <b>Step Over</b> , <b>Step Out</b> and <b>Continue</b> Commands	1481
H.6	Other Debugging Features	1484
H.6.1	<b>Edit</b> and <b>Continue</b>	1484
H.6.2	<b>Exception Assistant</b>	1486
H.6.3	<b>Just My Code™</b> Debugging	1487
H.6.4	Other Debugger Features	1487
H.7	Wrap-Up	1487
	<b>Index</b>	<b>1491</b>