

Contents

Preface **xxiii**

Before You Begin **xxxix**

I Introduction **I**

1.1	Introduction	2
1.2	Microsoft's Windows® Operating System	2
1.3	C, C++ and Java	3
1.4	Visual C#	4
1.5	Key Software Trend: Object Technology	5
1.6	The Internet and the World Wide Web	5
1.7	Extensible Markup Language (XML)	7
1.8	Introduction to Microsoft .NET	7
1.9	The .NET Framework and the Common Language Runtime	8
1.10	Test-Driving a C# Advanced Painter Application	9
1.11	(Only Required Section of the Case Study) Software Engineering Case Study: Introduction to Object Technology and the UML	11
1.12	Wrap-Up	16
1.13	Web Resources	16

2 Dive Into® Visual C# 2008 Express **18**

2.1	Introduction	19
2.2	Overview of the Visual Studio 2008 IDE	19
2.3	Menu Bar and Toolbar	25
2.4	Navigating the Visual Studio IDE	28
	2.4.1 Solution Explorer	30
	2.4.2 Toolbox	31
	2.4.3 Properties Window	32
2.5	Using Help	33
2.6	Using Visual Programming to Create a Simple Program that Displays Text and an Image	35
2.7	Wrap-Up	47
2.8	Web Resources	48

3 Introduction to C# Applications 49

3.1	Introduction	50
3.2	A Simple C# Application: Displaying a Line of Text	50
3.3	Creating a Simple Application in Visual C# Express	55
3.4	Modifying Your Simple C# Application	63
3.5	Formatting Text with <code>Console.Write</code> and <code>Console.WriteLine</code>	65
3.6	Another C# Application: Adding Integers	66
3.7	Arithmetic	70
3.8	Decision Making: Equality and Relational Operators	71
3.9	(Optional) Software Engineering Case Study: Examining the ATM Requirements Document	76
3.10	Wrap-Up	85
3.11	Web Resources	85

4 Introduction to Classes and Objects 86

4.1	Introduction	87
4.2	Classes, Objects, Methods, Properties and Instance Variables	87
4.3	Declaring a Class with a Method and Instantiating an Object of a Class	89
4.4	Declaring a Method with a Parameter	93
4.5	Instance Variables and Properties	96
4.6	UML Class Diagram with a Property	102
4.7	Software Engineering with Properties and <code>set</code> and <code>get</code> Accessors	102
4.8	Auto-Implemented Properties	104
4.9	Value Types vs. Reference Types	105
4.10	Initializing Objects with Constructors	107
4.11	Floating-Point Numbers and <code>decimal</code>	110
4.12	(Optional) Software Engineering Case Study: Identifying the Classes in the ATM Requirements Document	116
4.13	Wrap-Up	123

5 Control Statements: Part I 125

5.1	Introduction	126
5.2	Control Structures	126
5.3	<code>if</code> Single-Selection Statement	129
5.4	<code>if...else</code> Double-Selection Statement	130
5.5	<code>while</code> Repetition Statement	134
5.6	Counter-Controlled Repetition	135
5.7	Formulating Algorithms: Sentinel-Controlled Repetition	139
5.8	Formulating Algorithms: Nested Control Statements	144
5.9	Compound Assignment Operators	147
5.10	Increment and Decrement Operators	148
5.11	Simple Types	151

5.12	(Optional) Software Engineering Case Study: Identifying Class Attributes in the ATM System	151
5.13	Wrap-Up	156

6 Control Statements: Part 2 157

6.1	Introduction	158
6.2	Essentials of Counter-Controlled Repetition	158
6.3	for Repetition Statement	160
6.4	Examples Using the for Statement	164
6.5	do...while Repetition Statement	168
6.6	switch Multiple-Selection Statement	170
6.7	break and continue Statements	178
6.8	Logical Operators	180
6.9	(Optional) Software Engineering Case Study: Identifying Objects' States and Activities in the ATM System	186
6.10	Wrap-Up	190

7 Methods: A Deeper Look 192

7.1	Introduction	193
7.2	Packaging Code in C#	193
7.3	static Methods, static Variables and Class Math	194
7.4	Declaring Methods with Multiple Parameters	197
7.5	Notes on Declaring and Using Methods	201
7.6	Method-Call Stack and Activation Records	202
7.7	Argument Promotion and Casting	203
7.8	The .NET Framework Class Library	205
7.9	Case Study: Random-Number Generation	206
	7.9.1 Scaling and Shifting Random Numbers	210
	7.9.2 Random-Number Repeatability for Testing and Debugging	211
7.10	Case Study: A Game of Chance (Introducing Enumerations)	212
7.11	Scope of Declarations	217
7.12	Method Overloading	220
7.13	Recursion	223
7.14	Passing Arguments: Pass-by-Value vs. Pass-by-Reference	226
7.15	(Optional) Software Engineering Case Study: Identifying Class Operations in the ATM System	229
7.16	Wrap-Up	237

8 Arrays 238

8.1	Introduction	239
8.2	Arrays	239

8.3	Declaring and Creating Arrays	241
8.4	Examples Using Arrays	242
8.5	Case Study: Card Shuffling and Dealing Simulation	251
8.6	foreach Statement	255
8.7	Passing Arrays and Array Elements to Methods	257
8.8	Passing Arrays by Value and by Reference	259
8.9	Case Study: Class GradeBook Using an Array to Store Grades	263
8.10	Multidimensional Arrays	268
8.11	Case Study: Class GradeBook Using a Rectangular Array	273
8.12	Variable-Length Argument Lists	279
8.13	Using Command-Line Arguments	280
8.14	(Optional) Software Engineering Case Study: Collaboration Among Objects in the ATM System	282
8.15	Wrap-Up	290

9 Introduction to LINQ and Generic Collections 292

9.1	Introduction	293
9.2	Querying an Array Using LINQ	294
9.3	Introduction to Collections	303
9.4	Querying a Generic Collection Using LINQ	306
9.5	Wrap-Up	308
9.6	Deitel LINQ Resource Center	308

10 Classes and Objects: A Deeper Look 309

10.1	Introduction	310
10.2	Time Class Case Study	311
10.3	Controlling Access to Members	315
10.4	Referring to the Current Object's Members with the this Reference	316
10.5	Indexers	318
10.6	Time Class Case Study: Overloaded Constructors	321
10.7	Default and Parameterless Constructors	327
10.8	Composition	328
10.9	Garbage Collection and Destructors	331
10.10	static Class Members	332
10.11	readonly Instance Variables	336
10.12	Software Reusability	338
10.13	Data Abstraction and Encapsulation	339
10.14	Time Class Case Study: Creating Class Libraries	341
10.15	internal Access	345
10.16	Class View and Object Browser	347
10.17	Object Initializers	348
10.18	Time Class Case Study: Extension Methods	351
10.19	Delegates	354

10.20	Lambda Expressions	357
10.21	Anonymous Types	360
10.22	(Optional) Software Engineering Case Study: Starting to Program the Classes of the ATM System	362
10.23	Wrap-Up	368

11 Object-Oriented Programming: Inheritance 370

11.1	Introduction	371
11.2	Base Classes and Derived Classes	372
11.3	protected Members	374
11.4	Relationship between Base Classes and Derived Classes	375
11.4.1	Creating and Using a <code>CommissionEmployee</code> Class	376
11.4.2	Creating a <code>BasePlusCommissionEmployee</code> Class without Using Inheritance	381
11.4.3	Creating a <code>CommissionEmployee–BasePlusCommissionEmployee</code> Inheritance Hierarchy	385
11.4.4	<code>CommissionEmployee–BasePlusCommissionEmployee</code> Inheritance Hierarchy Using protected Instance Variables	388
11.4.5	<code>CommissionEmployee–BasePlusCommissionEmployee</code> Inheritance Hierarchy Using private Instance Variables	394
11.5	Constructors in Derived Classes	399
11.6	Software Engineering with Inheritance	405
11.7	Class object	406
11.8	Wrap-Up	407

12 Polymorphism, Interfaces and Operator Overloading 408

12.1	Introduction	409
12.2	Polymorphism Examples	411
12.3	Demonstrating Polymorphic Behavior	412
12.4	Abstract Classes and Methods	415
12.5	Case Study: Payroll System Using Polymorphism	417
12.5.1	Creating Abstract Base Class <code>Employee</code>	418
12.5.2	Creating Concrete Derived Class <code>SalariedEmployee</code>	420
12.5.3	Creating Concrete Derived Class <code>HourlyEmployee</code>	422
12.5.4	Creating Concrete Derived Class <code>CommissionEmployee</code>	423
12.5.5	Creating Indirect Concrete Derived Class <code>BasePlusCommissionEmployee</code>	425
12.5.6	Polymorphic Processing, Operator <code>is</code> and Downcasting	426
12.5.7	Summary of the Allowed Assignments Between Base-Class and Derived-Class Variables	431
12.6	sealed Methods and Classes	432
12.7	Case Study: Creating and Using Interfaces	433
12.7.1	Developing an <code>IPayable</code> Hierarchy	434

12.7.2	Declaring Interface <code>IPayable</code>	435
12.7.3	Creating Class <code>Invoice</code>	435
12.7.4	Modifying Class <code>Employee</code> to Implement Interface <code>IPayable</code>	437
12.7.5	Modifying Class <code>SalariedEmployee</code> for Use with <code>IPayable</code>	438
12.7.6	Using Interface <code>IPayable</code> to Process Invoices and Employees Polymorphically	440
12.7.7	Common Interfaces of the .NET Framework Class Library	442
12.8	Operator Overloading	443
12.9	(Optional) Software Engineering Case Study: Incorporating Inheritance and Polymorphism into the ATM System	446
12.10	Wrap-Up	455

13 Exception Handling 456

13.1	Introduction	457
13.2	Exception-Handling Overview	458
13.3	Example: Divide by Zero without Exception Handling	458
13.4	Example: Handling <code>DivideByZeroExceptions</code> and <code>FormatExceptions</code>	461
13.4.1	Enclosing Code in a try Block	464
13.4.2	Catching Exceptions	464
13.4.3	Uncaught Exceptions	464
13.4.4	Termination Model of Exception Handling	465
13.4.5	Flow of Control When Exceptions Occur	466
13.5	.NET Exception Hierarchy	466
13.5.1	Class <code>SystemException</code>	467
13.5.2	Determining Which Exceptions a Method Throws	467
13.6	<code>finally</code> Block	468
13.7	Exception Properties	476
13.8	User-Defined Exception Classes	481
13.9	Wrap-Up	484

14 Graphical User Interfaces with Windows Forms: Part I 485

14.1	Introduction	486
14.2	Windows Forms	487
14.3	Event Handling	490
14.3.1	A Simple Event-Driven GUI	490
14.3.2	Another Look at the Visual Studio Generated Code	492
14.3.3	Delegates and the Event-Handling Mechanism	493
14.3.4	Other Ways to Create Event Handlers	494
14.3.5	Locating Event Information	495
14.4	Control Properties and Layout	497
14.5	Labels, TextBoxes and Buttons	500
14.6	GroupBoxes and Panels	503

14.7	CheckBoxes and RadioButtons	507
14.8	PictureBoxes	515
14.9	ToolTips	518
14.10	NumericUpDown Control	520
14.11	Mouse-Event Handling	522
14.12	Keyboard-Event Handling	525
14.13	Wrap-Up	528

15 Graphical User Interfaces with Windows Forms: Part 2 **530**

15.1	Introduction	531
15.2	Menus	531
15.3	MonthCalendar Control	541
15.4	DateTimePicker Control	542
15.5	LinkLabel Control	545
15.6	ListBox Control	549
15.7	CheckedListBox Control	553
15.8	ComboBox Control	556
15.9	TreeView Control	560
15.10	ListView Control	565
15.11	TabControl Control	571
15.12	Multiple Document Interface (MDI) Windows	576
15.13	Visual Inheritance	584
15.14	User-Defined Controls	587
15.15	Wrap-Up	592

16 GUI with Windows Presentation Foundation **593**

16.1	Introduction	594
16.2	Windows Presentation Foundation (WPF)	595
16.3	XML Basics	596
16.4	Structuring Data	599
16.5	XML Namespaces	604
16.6	Declarative GUI Programming Using XAML	608
16.7	Creating a WPF Application in Visual C# Express	610
16.8	Laying Out Controls	612
	16.8.1 General Layout Principles	612
	16.8.2 Layout in Action	613
16.9	Event Handling	618
16.10	Commands and Common Application Tasks	625
16.11	WPF GUI Customization	630
16.12	Using Styles to Change a Control's Appearance	631
16.13	Customizing Windows	636
16.14	Defining a Control's Appearance with Control Templates	639

16.15	Data-Driven GUIs with Data Binding	644
16.16	Wrap-Up	650
16.17	Web Resources	650

17 WPF Graphics and Multimedia 651

17.1	Introduction	652
17.2	Controlling Fonts	652
17.3	Basic Shapes	654
17.4	Polygons and PolyLines	656
17.5	Brushes	659
17.6	Transforms	665
17.7	WPF Customization: A Television GUI	668
17.8	Animations	677
17.9	(Optional) 3-D Objects and Transforms	680
17.10	Wrap-Up	687

18 Strings, Characters and Regular Expressions 688

18.1	Introduction	689
18.2	Fundamentals of Characters and Strings	690
18.3	string Constructors	691
18.4	string Indexer, Length Property and CopyTo Method	692
18.5	Comparing strings	693
18.6	Locating Characters and Substrings in strings	697
18.7	Extracting Substrings from strings	699
18.8	Concatenating strings	700
18.9	Miscellaneous string Methods	701
18.10	Class StringBuilder	702
18.11	Length and Capacity Properties, EnsureCapacity Method and Indexer of Class StringBuilder	704
18.12	Append and AppendFormat Methods of Class StringBuilder	705
18.13	Insert, Remove and Replace Methods of Class StringBuilder	708
18.14	Char Methods	710
18.15	Card Shuffling and Dealing Simulation	713
18.16	Introduction to Regular-Expression Processing	717
	18.16.1 Simple Regular Expressions and Class Regex	718
	18.16.2 Complex Regular Expressions	723
	18.16.3 Validating User Input with Regular Expressions and LINQ	724
	18.16.4 Regex Methods Replace and Split	729
18.17	Wrap-Up	731

19 Files and Streams 732

19.1	Introduction	733
19.2	Data Hierarchy	733

19.3	Files and Streams	735
19.4	Classes File and Directory	736
19.5	Creating a Sequential-Access Text File	745
19.6	Reading Data from a Sequential-Access Text File	755
19.7	Case Study: Credit Inquiry Program Using LINQ	760
19.8	Serialization	766
19.9	Creating a Sequential-Access File Using Object Serialization	766
19.10	Reading and Deserializing Data from a Binary File	773
19.11	Wrap-Up	777

20 XML and LINQ to XML 778

20.1	Introduction	779
20.2	Document Type Definitions (DTDs)	779
20.3	W3C XML Schema Documents	783
20.4	Extensible Stylesheet Language and XSL Transformations	790
20.5	LINQ to XML: Document Object Model (DOM)	799
20.6	LINQ to XML Class Hierarchy	803
20.7	LINQ to XML: Namespaces and Creating Documents	812
20.8	XSLT with Class XsltCompiledTransform	815
20.9	Wrap-Up	817
20.10	Web Resources	818

21 Databases and LINQ to SQL 819

21.1	Introduction	820
21.2	Relational Databases	821
21.3	Relational Database Overview: Books Database	822
21.4	SQL	826
21.4.1	Basic SELECT Query	826
21.4.2	WHERE Clause	827
21.4.3	ORDER BY Clause	829
21.4.4	Retrieving Data from Multiple Tables: INNER JOIN	831
21.4.5	INSERT Statement	832
21.4.6	UPDATE Statement	833
21.4.7	DELETE Statement	834
21.5	LINQ to SQL	835
21.6	LINQ to SQL: Extracting Information from a Database	836
21.6.1	Creating LINQ to SQL Classes	836
21.6.2	Creating Data Bindings	837
21.7	More Complex LINQ Queries and Data Binding	840
21.8	Retrieving Data from Multiple Tables with LINQ	845
21.9	Creating a Master/Detail View Application	848
21.10	Programming with LINQ to SQL: Address-Book Case Study	853
21.11	Wrap-Up	859
21.12	Tools and Web Resources	859

22	ASP.NET 3.5 and ASP.NET AJAX	861
22.1	Introduction	862
22.2	Simple HTTP Transactions	863
22.3	Multitier Application Architecture	867
22.4	Creating and Running a Simple Web-Form Example	868
	22.4.1 Examining an ASPX File	868
	22.4.2 Examining a Code-Behind File	870
	22.4.3 Relationship Between an ASPX File and a Code-Behind File	871
	22.4.4 How the Code in an ASP.NET Web Page Executes	872
	22.4.5 Examining the XHTML Generated by an ASP.NET Application	872
	22.4.6 Building an ASP.NET Web Application	874
22.5	Web Controls	882
	22.5.1 Text and Graphics Controls	883
	22.5.2 AdRotator Control	889
	22.5.3 Validation Controls	892
22.6	Session Tracking	899
	22.6.1 Cookies	900
	22.6.2 Session Tracking with HttpSessionState	908
22.7	Case Study: Connecting to a Database in ASP.NET	915
	22.7.1 Building a Web Form That Displays Data from a Database	916
	22.7.2 Modifying the Code-Behind File for the Guestbook Application	923
22.8	Case Study: Secure Books Database Application	924
	22.8.1 Examining the Completed Secure Books Database Application	925
	22.8.2 Creating the Secure Books Database Application	928
22.9	ASP.NET AJAX	952
	22.9.1 Traditional Web Applications	952
	22.9.2 Ajax Web Applications	953
	22.9.3 Examining an ASP.NET AJAX Application	953
22.10	New ASP.NET 3.5 Data Controls	960
22.11	Wrap-Up	961
22.12	Web Resources	962

23	Windows Communication Foundation (WCF) Web Services	963
23.1	Introduction	964
23.2	WCF Services Basics	965
23.3	Simple Object Access Protocol (SOAP)	965
23.4	Representational State Transfer (REST)	966
23.5	JavaScript Object Notation (JSON)	966
23.6	Publishing and Consuming SOAP-Based Web Services	967
	23.6.1 Creating a WCF Web Service	967
	23.6.2 Code for the WelcomeSOAPXMLService	967
	23.6.3 Building a SOAP-Based Web Service	968
	23.6.4 Deploying the WelcomeSOAPXMLService	971

23.6.5	Creating a Client to Consume the <code>WelcomeSOAPXMLService</code>	973
23.6.6	Consuming the <code>WelcomeSOAPXMLService</code>	975
23.7	Publishing and Consuming REST-Based XML Web Services	976
23.7.1	Creating a REST-Based XML Web Service	976
23.7.2	Consuming a REST-Based XML Web Service	979
23.8	Publishing and Consuming REST-Based JSON Web Services	980
23.8.1	Creating a REST-Based JSON Web Service	980
23.8.2	Consuming a REST-Based JSON Web Service	982
23.9	Blackjack Web Service: Using Session Tracking in a SOAP-Based Web Service	984
23.9.1	Creating a Blackjack Web Service	984
23.9.2	Consuming the Blackjack Web Service	988
23.10	Airline Reservation Web Service: Database Access and Invoking a Service from ASP.NET	997
23.11	Equation Generator: Returning User-Defined Types	1002
23.11.1	Creating the REST-Based XML <code>EquationGenerator</code> Web Service	1005
23.11.2	Consuming the REST-Based XML <code>EquationGenerator</code> Web Service	1006
23.11.3	Creating the REST-Based JSON <code>EquationGenerator</code> Web Service	1010
23.11.4	Consuming the REST-Based JSON <code>EquationGenerator</code> Web Service	1011
23.12	Wrap-Up	1014
23.13	Deitel Web Services Resource Centers	1015

24 Silverlight, Rich Internet Applications and Multimedia **1016**

24.1	Introduction	1017
24.2	Platform Overview	1018
24.3	Silverlight Runtime and Tools Installation	1019
24.4	Building a Silverlight <code>WeatherViewer</code> Application	1019
24.4.1	GUI Layout	1022
24.4.2	Obtaining and Displaying Weather Forecast Data	1024
24.4.3	Custom Controls	1028
24.5	Animations and the <code>FlickerViewer</code>	1031
24.6	Images and Deep Zoom	1037
24.6.1	Getting Started With Deep Zoom Composer	1040
24.6.2	Creating a Silverlight Deep Zoom Application	1042
24.7	Audio and Video	1050
24.8	Isolated Storage	1055
24.9	Silverlight Demos and Web Resources	1056
24.10	Wrap-Up	1057

25	Data Structures	1059
25.1	Introduction	1060
25.2	Simple-Type structs, Boxing and Unboxing	1060
25.3	Self-Referential Classes	1061
25.4	Linked Lists	1062
25.5	Stacks	1075
25.6	Queues	1079
25.7	Trees	1082
	25.7.1 Binary Search Tree of Integer Values	1083
	25.7.2 Binary Search Tree of Comparable Objects	1090
25.8	Wrap-Up	1095
26	Generics	1097
26.1	Introduction	1098
26.2	Motivation for Generic Methods	1099
26.3	Generic-Method Implementation	1101
26.4	Type Constraints	1103
26.5	Overloading Generic Methods	1106
26.6	Generic Classes	1107
26.7	Wrap-Up	1116
27	Collections	1118
27.1	Introduction	1119
27.2	Collections Overview	1120
27.3	Class Array and Enumerators	1122
27.4	Nongeneric Collections	1125
	27.4.1 Class ArrayList	1126
	27.4.2 Class Stack	1130
	27.4.3 Class Hashtable	1132
27.5	Generic Collections	1137
	27.5.1 Generic Class SortedDictionary	1137
	27.5.2 Generic Class LinkedList	1140
27.6	Wrap-Up	1144
A	Operator Precedence Chart	1145
B	Simple Types	1147
C	Number Systems	1149

C.1	Introduction	1150
C.2	Abbreviating Binary Numbers as Octal and Hexadecimal Numbers	1153
C.3	Converting Octal and Hexadecimal Numbers to Binary Numbers	1154
C.4	Converting from Binary, Octal or Hexadecimal to Decimal	1154
C.5	Converting from Decimal to Binary, Octal or Hexadecimal	1155
C.6	Negative Binary Numbers: Two's Complement Notation	1157

D ATM Case Study Code 1159

D.1	ATM Case Study Implementation	1159
D.2	Class ATM	1160
D.3	Class Screen	1166
D.4	Class Keypad	1166
D.5	Class CashDispenser	1167
D.6	Class DepositSlot	1168
D.7	Class Account	1169
D.8	Class BankDatabase	1171
D.9	Class Transaction	1174
D.10	Class BalanceInquiry	1176
D.11	Class Withdrawal	1177
D.12	Class Deposit	1181
D.13	Class ATMCASEStudy	1183
D.14	Wrap-Up	1183

E UML 2: Additional Diagram Types 1185

E.1	Introduction	1185
E.2	Additional Diagram Types	1185

F ASCII Character Set 1187

G Unicode® 1188

G.1	Introduction	1189
G.2	Unicode Transformation Formats	1190
G.3	Characters and Glyphs	1191
G.4	Advantages/Disadvantages of Unicode	1191
G.5	Using Unicode	1192
G.6	Character Ranges	1194

H Using the Visual C# 2008 Debugger 1196

H.1	Introduction	1197
-----	--------------	------

H.2	Breakpoints and the Continue Command	1197
H.3	<i>DataTips</i> and Visualizers	1203
H.4	The Locals and Watch Windows	1204
H.5	Controlling Execution Using the Step Into , Step Over , Step Out and Continue Commands	1207
H.6	Other Debugging Features	1210
H.6.1	Edit and Continue	1210
H.6.2	Exception Assistant	1212
H.6.3	Just My Code™ Debugging	1213
H.6.4	Other Debugger Features	1213
H.7	Wrap-Up	1213

Index	1215
--------------	-------------