

Contents

<i>About the Author</i>	<i>xxi</i>
<i>Acknowledgments</i>	<i>xxiii</i>
<i>Introduction</i>	<i>xxv</i>
Part One Introducing C# and the .NET Platform.....	1
<i>Chapter 1 The Philosophy of .NET</i>	3
<i>Understanding the Previous State of Affairs</i>	3
<i>The .NET Solution</i>	6
<i>The Building Blocks of the .NET Platform</i> <i>(CLR, CTS, and CLS)</i>	7
<i>The Role of the .NET Base Class Libraries</i>	7
<i>What C# Brings to the Table</i>	8
<i>Additional .NET-Aware Programming Languages</i>	9
<i>An Overview of .NET Binaries (aka Assemblies)</i>	11
<i>The Role of the Common Intermediate Language</i>	13
<i>The Role of .NET Type Metadata</i>	16
<i>The Role of the Assembly Manifest</i>	17
<i>Compiling CIL to Platform-Specific Instructions</i>	18
<i>Understanding the Common Type System</i>	18
<i>Intrinsic CTS Data Types</i>	22
<i>Understanding the Common Language Specification</i>	23
<i>Understanding the Common Language Runtime</i>	25
<i>A Tour of the .NET Namespaces</i>	28
<i>Increasing Your Namespace Nomenclature</i>	33
<i>Deploying the .NET Runtime</i>	39
<i>Summary</i>	41

Chapter 2	Building C# Applications	43
	The Role of the Command Line Compiler (<i>csc.exe</i>)	43
	Building a C# Application Using <i>csc.exe</i>	45
	Working with <i>csc.exe</i> Response Files	50
	Generating Bug Reports	52
	Remaining C# Compiler Options	54
	The Command Line Debugger (<i>cordbg.exe</i>)	55
	Using the Visual Studio .NET IDE	57
	Building a VS .NET Test Application	61
	Debugging with the Visual Studio .NET IDE	70
	Other Key Aspects of the VS .NET IDE	71
	Documenting Your Source Code via XML	76
	C# “Preprocessor” Directives	82
	An Interesting Aside: The <i>System.Environment</i> Class	88
	Building .NET Applications with Other IDEs	89
	Summary	90
Part Two	The C# Programming Language	91
Chapter 3	C# Language Fundamentals	93
	The Anatomy of a Basic C# Class	93
	Creating Objects: Constructor Basics	97
	The Composition of a C# Application	100
	Default Assignments and Variable Scope	102
	The C# Member Variable Initialization Syntax	104
	Basic Input and Output with the <i>Console</i> Class	105
	Understanding Value Types and Reference Types	109
	The Master Node: <i>System.Object</i>	115
	The System Data Types (and C# Aliases)	122
	Converting Between Value Types and Reference Types:	
	Boxing and Unboxing	128
	Defining Program Constants	132
	C# Iteration Constructs	134
	C# Control Flow Constructs	137
	The Complete Set of C# Operators	139
	Defining Custom Class Methods	140
	Understanding Static Methods	143
	Method Parameter Modifiers	147
	Array Manipulation in C#	154
	String Manipulation in C#	160
	C# Enumerations	164

Defining Structures in C#	168
Defining Custom Namespaces	171
Summary	176
 Chapter 4 Object-Oriented Programming with C#	 179
Formal Definition of the C# Class	179
Defining the “Default Public Interface” of a Type	184
Recapping the Pillars of OOP	186
The First Pillar: C#’s Encapsulation Services	191
Pseudo-Encapsulation: Creating Read-Only Fields	199
The Second Pillar: C#’s Inheritance Support	201
Keeping Family Secrets: The “protected” Keyword	205
Nested Type Definitions	211
The Third Pillar: C#’s Polymorphic Support	213
Casting Between Types	222
Generating Class Definitions Using Visual Studio .NET	225
Summary	229
 Chapter 5 Exceptions and Object Lifetime	 231
Ode to Errors, Bugs, and Exceptions	231
The Role of .NET Exception Handling	232
The System.Exception Base Class	234
Throwing a Generic Exception	235
Catching Exceptions	236
CLR System-Level Exceptions (System.SystemException)	240
Custom Application-Level Exceptions (System.ApplicationException)	243
Handling Multiple Exceptions	247
The Finally Block	250
The Last Chance Exception	251
Dynamically Identifying Application- and System-Level Exceptions	252
Debugging System Exceptions Using VS .NET	253
Understanding Object Lifetime	257
The CIL of “new”	257
The Basics of Garbage Collection	259
Finalizing a Type	260
The Finalization Process	263
Building an Ad Hoc Destruction Method	264
Garbage Collection Optimizations	267
The System.GC Type	267
Summary	272

Chapter 6	Interfaces and Collections	273
Defining Interfaces Using C#		273
Invoking Interface Members at the Object Level		277
Exercising the Shapes Hierarchy		279
Understanding Explicit Interface Implementation		281
Interfaces As Polymorphic Agents		284
Building Interface Hierarchies		285
Implementing Interfaces Using VS .NET		288
Understanding the IConvertible Interface		290
Building a Custom Enumerator		
(IEnumerable and IEnumerator)		293
Building Cloneable Objects (ICloneable)		297
Building Comparable Objects (IComparable)		302
Exploring the System.Collections Namespace		306
Building a Custom Container		
(Retrofitting the Cars Type)		315
Summary		318
 Chapter 7	 Callback Interfaces, Delegates, and Events	 321
Understanding Callback Interfaces		321
Understanding the .NET Delegate Type		325
Members of System.MulticastDelegate		328
The Simplest Possible Delegate Example		329
Building a More Elaborate Delegate Example		333
Understanding Asynchronous Delegates		341
Understanding (and Using) Events		347
Summary		354
 Chapter 8	 Advanced C# Type Construction Techniques	 355
The Advanced Keywords of C#		355
A Catalog of C# Keywords		367
Building a Custom Indexer		370
A Variation of the Cars Indexer		372
Internal Representation of Type Indexers		373
Using the C# Indexer from VB .NET		374
Overloading Operators		375
The Internal Representation of Overloaded Operators		380

<i>Interacting with Overloaded Operators from Overloaded-Operator-Challenged Languages</i>	<i>383</i>
<i>Final Thoughts Regarding Operator Overloading</i>	<i>385</i>
<i>Understanding Custom Type Conversions</i>	<i>386</i>
<i>Creating Custom Conversion Routines</i>	<i>387</i>
<i>Defining Implicit Conversion Routines</i>	<i>390</i>
<i>The Internal Representation of Custom Conversion Routines</i>	<i>391</i>
<i>Summary</i>	<i>392</i>
 Part Three Programming with .NET Assemblies.....	393
 Chapter 9 Understanding .NET Assemblies	395
<i>Problems with Classic COM Binaries</i>	<i>395</i>
<i>An Overview of .NET Assemblies</i>	<i>397</i>
<i>Building a Single File Test Assembly</i>	<i>404</i>
<i>A C# Client Application</i>	<i>408</i>
<i>A Visual Basic .NET Client Application</i>	<i>409</i>
<i>Cross-Language Inheritance</i>	<i>411</i>
<i>Exploring the Carlibrary's Manifest</i>	<i>415</i>
<i>Exploring the Carlibrary's Types</i>	<i>418</i>
<i>Building a Multifile Assembly</i>	<i>419</i>
<i>Using the Multifile Assembly</i>	<i>422</i>
<i>Understanding Private Assemblies</i>	<i>425</i>
<i>Probing for Private Assemblies (The Basics)</i>	<i>425</i>
<i>Private Assemblies and XML Configuration Files</i>	<i>426</i>
<i>Probing for Private Assemblies (The Details)</i>	<i>429</i>
<i>Understanding Shared Assemblies</i>	<i>430</i>
<i>Understanding Strong Names</i>	<i>431</i>
<i>Building a Shared Assembly</i>	<i>432</i>
<i>Understanding Delayed Signing</i>	<i>434</i>
<i>Installing/Removing Shared Assemblies</i>	<i>435</i>
<i>Using a Shared Assembly</i>	<i>436</i>
<i>Versioning Shared Assemblies</i>	<i>437</i>
<i>Building SharedAssembly Version 2.0.0.0</i>	<i>438</i>
<i>Specifying Custom Version Policies</i>	<i>440</i>
<i>GAC Internals</i>	<i>440</i>
<i>Assembly-Centric Odds and Ends</i>	<i>442</i>
<i>Regarding the VS .NET Add References Dialog Box</i>	<i>449</i>
<i>Summary</i>	<i>450</i>

Chapter 10 Processes, AppDomains, Contexts, and Threads	451
Reviewing Processes and Threads	
Under Traditional Win32	451
Interacting with Processes Under the .NET Platform	453
Understanding the System.AppDomain Type	462
Understanding Context (or How Low Can You Go?)	468
Summarizing Processes, AppDomains, and Context	474
The Process/AppDomain/Context/Thread Relationship	474
Multithreaded Programming via Delegates	476
The System.Threading Namespace	476
Spawning Secondary Threads	481
A More Elaborate Threading Example	484
Concurrency Revisited	487
Synchronization Using the C# “lock” Keyword	490
Synchronization Using the	
System.Threading.Interlocked Type	492
Synchronization Using the [Synchronized] Attribute	493
Thread Safety and the .NET Base Class Libraries	494
Programming with Timer Callbacks	494
Summary	496
 Chapter 11 Type Reflection, Late Binding, and Attribute-Based Programming	 499
The Necessity of Type Metadata	499
Understanding Reflection	504
The System.Type Class	504
Investigating the System.Reflection Namespace	511
Reflecting on a Private Assembly	511
Reflecting on Shared Assemblies	516
Understanding Dynamic Invocation (Late Binding)	517
Understanding Attributed Programming	520
Building Custom Attributes	523
Restricting Attribute Usage	524
Assembly- (and Module-) Level Attributes	526
Reflecting on Attributes at Runtime	528
Putting Reflection, Late Binding, and	
Custom Attributes in Perspective	528
Summary	535

Part Four	Leveraging the .NET Libraries	537
Chapter 12	Object Serialization	
	and the .NET Remoting Layer	539
Object Persistence in the .NET Framework		540
The Role of Object Graphs		540
Configuring Objects for Serialization		541
Serialization Using a Binary Formatter		545
Serialization Using a SOAP Formatter		547
Serialization Using an XML Formatter		548
Customizing the Serialization Process		549
Defining .NET Remoting		554
The .NET Remoting Namespaces		554
Understanding the .NET Remoting Framework		556
All Together Now!		559
Terms of the .NET Remoting Trade		560
Object Marshaling Choices (MBR or MBV?)		560
Final Notes Regarding MBR/MBV Objects		563
Activation Choices for MBR Types (WKO or CAO?)		563
Basic Deployment of a .NET Remoting Project		566
Building Our First Distributed Application		568
Testing the Remoting Application		571
Understanding the ChannelServices Type		572
Understanding the RemotingConfiguration Type		573
Revisiting the Activation Mode of WKO Types		576
Deploying the Server to a Remote Machine		577
Leveraging the TCP Channel		578
Remoting Configuration Files		579
Working with MBV Objects		582
Understanding Client-Activated Objects (CAO)		587
The Lease-Based Lifetime of CAO/WKO-Singleton Objects		589
Server-side (and Client-side) Lease Sponsorship		595
Alternative Hosts for Remote Objects		597
Asynchronous Remoting		603
The Role of the [OneWayAttribute] Type		604
Final Thoughts... ..		605
Summary		606

Chapter 13 Building a Better Window (Introducing Windows Forms)	607
A Tale of Three GUI Namespaces	607
Overview of the System.Windows.Forms Namespace	608
Interacting with the Windows Forms Types	609
Building a VS .NET Windows Forms Project Workspace	612
The System.Windows.Forms.Application Class	615
The Anatomy of a Form	621
The Component Class	622
The Control Class	623
Control Events	627
Responding to Keyboard Events	631
The Control Class Revisited	633
The ScrollableControl Class	636
ContainerControl Class	637
The Form Class	637
The Life-Cycle of a Windows Form Type	640
Handling Form Level Events a la VS .NET	643
Building Menus with Windows Forms	644
Building Your Menu System	646
Creating a Pop-Up Menu	649
Adorning Your Menu System	651
Building a Menu Using Visual Studio .NET	654
Understanding Status Bars	655
Building a Tool Bar	660
Building ToolBars at Design Time	664
Building an MDI Application	666
Summary	669
Chapter 14 A Better Painting Framework (GDI+)	671
Survey of the GDI+ Namespaces	671
Overview of the System.Drawing Namespace	673
Examining the System.Drawing Utility Types	675
Regarding the Disposal of System.Drawing Types	680
Understanding Paint Sessions	680
Understanding the Graphics Class	685
The GDI+ Coordinate Systems	687
Establishing an Active Color	692
Manipulating Fonts	695
Building a Font Application	698
The FontDialog Class	703
Survey of the System.Drawing.Drawing2D Namespace	705

Rendering Images	720
Dragging, Hit Testing, and the PictureBox Control	723
Understanding the .NET Resource Format	732
System.Resources Namespace	733
Working with ResourceWriters	737
Working with ResourceManagers	737
Automatic Resource Configuration a la Visual Studio .NET	739
Summary	742

Chapter 15 Programming with Windows

Forms Controls.....	743
Understanding the Windows Forms Control Hierarchy	743
Adding Controls to Forms (IDE-Free)	744
Adding Controls to Forms (via VS .NET)	747
The TextBox Control	749
The Mighty Button Type	752
Working with CheckBoxes	755
Working with RadioButtons and GroupBoxes	756
ListBoxes and ComboBoxes	761
The MonthCalendar Control	763
More on the DateTime Type	766
Setting the Form's Default Input Button	767
Configuring the Tab Order	768
Assigning ToolTips to Controls	769
The TrackBar Control	770
Working with Panel Controls	773
The UpDown Controls: DomainUpDown and NumericUpDown	774
Working with the ErrorProvider	776
Configuring a Control's Anchoring Behavior	779
Configuring a Control's Docking Behavior	780
Building Custom Dialog Boxes	781
Understanding Form Inheritance	787
Building Custom Windows Forms Controls	789
Building a Custom UserControl	790
Creating the Images	791
Building the Design Time GUI	792
Implementing the CarControl	792
Controlling the Animation	796
Rendering the Pet Name	796
Testing the CarControl Type	797
Select Members of the System.ComponentModel Namespace	798
Enhancing the Design Time Appearance of CarControl	800
Summary	803

Chapter 16 The System.IO Namespace	805
Exploring the System.IO Namespace	805
The Directory(Info) and File(Info) Types	806
Enumerating Files with the DirectoryInfo Type	810
Creating Subdirectories with the DirectoryInfo Type	811
The Static Members of the Directory Class	812
The FileInfo Class	814
The Abstract Stream Class	818
Working with StreamWriters and StreamReaders	822
Working with StringWriters	826
Working with StringReaders	828
Working with Binary Data (BinaryReaders and BinaryWriters)	829
“Watching” Files and Directories	831
A Brief Word Regarding Asynchronous IO	833
A Windows Forms Car Logger Application	834
Summary	841
 Chapter 17 Data Access with ADO.NET	 843
The Need for ADO.NET	843
The Two Faces of ADO.NET	844
The Role of ADO.NET Data Providers	845
Understanding the ADO.NET Namespaces	850
The Types of System.Data	851
Examining the DataColumn Type	852
Examining the DataRow Type	858
Details of the DataTable	861
Building a Complete DataTable	863
Understanding the DataView Type	871
Understanding the Role of the DataSet	874
Expressing Relations Using the DataRelation Type	879
Reading and Writing XML-Based DataSets	883
Building a Simple Test Database	886
Selecting a Data Provider	887
The Types of the System.Data.OleDb Namespace	888
Working with the Connected Layer of ADO.NET	888
Working with the OleDbDataReader	894
Inserting, Updating, and Deleting Records Using OleDbCommand	897
Executing a Stored Procedure Using OleDbCommand	899
The Disconnected Layer and the OleDbDataAdapter Type	901
Working with the SQL Data Provider	906

Auto-Generating SQL Commands Using CommandBuilder Types	912
A Complete ADO.NET Windows Forms Example	914
Filling a Multitabled DataSet (and Adding DataRelations)	916
Bring In the Wizards!	919
Working with DataSets at Design Time	927
Summary	931
 Part Five Web Applications and XML Web Services	 933
 Chapter 18 ASP.NET Web Pages and Web Controls	 935
The Role of HTTP	935
Understanding Web Applications and Web Servers	936
The Role of HTML	939
HTML Form Development	940
The Role of Client-Side Scripting	943
Submitting the Form Data (GET and POST)	947
Building a Classic Active Server Page	947
The Problem(s) with Classic ASP	950
Some Benefits of ASP.NET	950
The ASP.NET Namespaces	952
Creating an ASP.NET Web Application by Hand	953
The ASP.NET Compilation Cycle	958
Creating an ASP.NET Web Application via VS .NET	961
Coding the BetterAspNetCarApp Web Application	966
The Composition of an ASP.NET Page	968
The Derivation of an ASP.NET Page	970
Interacting with the Incoming HTTP Request	971
Interacting with the Outgoing HTTP Response	975
The Life Cycle of an ASP.NET Web Page	978
Understanding the ASP.NET Web Controls	982
Key Members of the System.Web.UI.Control Type	985
Key Members of the System.Web.UI.WebControl Type	990
Select Examples of ASP.NET WebForm Controls	991
The Role of the Validation Controls	1005
Understanding the Role of ASP.NET HTML Controls	1014
And Now for Something Completely Different: GDI+ on the Web Server	 1016
Debugging and Tracing ASP.NET Pages	1020
Summary	1021

Chapter 19 ASP.NET Web Applications	1023
The Issue of State	1023
ASP.NET State Management Techniques	1026
Understanding the Role of ASP.NET View State	1026
The Role of the Global.asax File	1030
Understanding the Application/Session Distinction	1033
Working with the Application Cache	1037
Maintaining Session Data	1043
Understanding Cookies	1047
Configuring Your ASP.NET Web Application	
Using web.config	1050
Configuration Inheritance	1057
Summary	1058
 Chapter 20 XML Web Services	 1059
Understanding the Role of XML Web Services	1059
The Building Blocks of an XML Web Service	1062
The .NET XML Web Service Namespaces	1064
Examining the System.Web.Services Namespace	1064
Building an XML Web Service in the Raw	1065
Testing Your XML Web Service	1066
Building an XML Web Service Using Visual Studio .NET	1069
Understanding the System.Web.Services.WebService	
Base Class	1072
Understanding the [WebMethod] Attribute	1073
Understanding the [WebService] Attribute	1077
Exploring the Web Service Description Language (WSDL)	1078
The Basic Format of a WSDL Document	1079
Viewing the HelloWS WSDL Document	1082
The wsdl.exe Command Line Utility	1084
Revisiting the XML Web Service Wire Protocols	1086
Transforming WSDL into C# Code (Generating a Proxy)	1089
Leveraging the Proxy (Synchronous Invocations)	1092
Leveraging the Proxy (Asynchronous Invocations)	1093
Avoiding Hard-Coded Proxy Logic	1094
Generating a Proxy with VS .NET	1095
Exposing Arrays of Types from Web Methods	1096
A Windows Forms Client	1097

<i>Exposing Custom Types: The Details</i>	1098
<i>Consuming Custom Types: The Details</i>	1099
<i>Updating the CarsSalesInfowS Project</i>	1100
<i>Understanding the Discovery Service Protocol (UDDI)</i>	1102
<i>Summary</i>	1105
 <i>Index</i>	 1107