

Contents at a Glance

| | |
|--|------------|
| <i>Introduction</i> | xxi |
| Part 1 • Basic Web Programming | 1 |
| Chapter 1 • Behind the Scenes: How Web Applications Work | 3 |
| Chapter 2 • HTML Basics | 19 |
| Chapter 3 • Brief Guide to Dynamic Web Applications | 77 |
| Part 2 • Server-Side Web Programming with Visual C# | 91 |
| Chapter 4 • Introduction to ASP.NET | 93 |
| Chapter 5 • Introduction to Web Forms | 113 |
| Chapter 6 • Introduction to the System.Web Namespace | 139 |
| Chapter 7 • The SessionState Object | 187 |
| Chapter 8 • The HttpServerUtility Object | 215 |
| Chapter 9 • Debugging ASP.NET and Error Handling | 235 |
| Chapter 10 • File and Event Log Access with ASP.NET | 267 |
| Chapter 11 • Sending and Receiving Messages with ASP.NET | 311 |
| Part 3 • Accessing Data with ASP.NET | 345 |
| Chapter 12 • Introduction to Relational Databases and SQL | 347 |
| Chapter 13 • Introduction to ADO.NET | 369 |
| Chapter 14 • Accessing Data | 399 |
| Chapter 15 • Using XML in Web Applications | 451 |

| | |
|---|------------|
| Part 4 • C# Web Applications | 483 |
| Chapter 16 • Introduction to C# Web Applications | 485 |
| Chapter 17 • State Maintenance and Cacheing | 513 |
| Chapter 18 • Controlling Access and Monitoring | 549 |
| Chapter 19 • Planning Applications | 579 |
| Part 5 • Advanced Visual C# Web Applications | 595 |
| Chapter 20 • Leveraging Browser Clients | 597 |
| Chapter 21 • Web Services | 635 |
| Chapter 22 • Web Services, COM Components, and the SOAP Toolkit | 667 |
| Chapter 23 • Build Your Own Web Controls | 683 |
| Chapter 24 • Efficiency and Scalability | 711 |
| Afterword | 723 |
| Part 6 • Appendices | 727 |
| Appendix A • Quick HTML Reference | 729 |
| Appendix B • JScript 5.5 Reference | 755 |
| Index | 785 |

Contents

| | |
|-------------------------------|-----|
| <i>Introduction</i> | xxi |
|-------------------------------|-----|

Part 1 • Basic Web Programming 1

Chapter 1 • Behind the Scenes: How Web Applications Work 3

| | |
|---|----|
| How Web Requests Work | 3 |
| How a Client Requests Content | 5 |
| How the Web Server Responds—Preparation | 6 |
| Parts of a URL | 6 |
| Server Translates the Path | 7 |
| Server Checks for the Resource | 7 |
| Server Checks Permissions | 7 |
| How the Web Server Responds—Fulfillment | 8 |
| What the Client Does with the Response | 9 |
| Introducing Dynamic Web Pages | 10 |
| How Does the Server Separate Code from Content? | 11 |
| How and When Does the Server Process Code? | 11 |
| How Do Clients Act with Dynamic Server Pages? | 12 |
| When Is HTML Not Enough? | 12 |
| What C# Can Do | 13 |
| Make If/Then Decisions | 13 |
| Process Information from Clients | 13 |
| Access Data and Files | 13 |
| Format Responses Using XML, CSS, XSLT, and HTML | 14 |
| Launch and Communicate with .NET and COM+ Objects | 14 |
| Advantages of C# in Web Applications | 15 |
| Web Services | 15 |
| Thin-Client Applications (Web Forms) | 16 |
| Rich-Client Applications (Windows Forms) | 17 |
| Summary | 17 |

Chapter 2 • HTML Basics 19

| | |
|---|----|
| What Is HTML? | 19 |
| What Does HTML Do? | 20 |
| Why Is HTML Important? | 21 |
| The Limitations of HTML | 21 |
| Syntax: Tags and Attributes | 22 |
| What Is a Tag? | 23 |
| What Is an End Tag? | 23 |
| Why Does HTML Look Like <THIS>? | 23 |
| Attribute Syntax | 24 |

- More HTML Syntax 25
- Creating a Simple Page 26
- Formatting Text 30
 - List Styles 30
 - Heading Styles 34
 - Fonts and Colors 35
 - Paragraph Tags, Div Tags, and Spans 37
 - The <body> Tag 38
 - Creating a Formatted Page with Fonts and Colors 39
- Including Images on Your Web Site 41
 - The Image Tag 41
 - Placing Images on a Page 43
 - Wrapping Text Around Images 45
 - Background Images 45
- Introduction to Hyperlinking 46
 - The Anchor Tag 47
 - Anchor Tags and Images 54
- Formatting Tables 54
 - Table, Table Row, Table Data Tags 54
 - Table Header, Table Footer Tags 58
 - Using Tables for Layout 58
- Using Image Maps 59
 - How Image Maps Work 60
 - Creating Client-Side Image Maps 61
 - Tools for Creating Image Maps 62
- Understanding Frames 62
 - Advantages and Disadvantages of Frames 65
 - How to Avoid Frames 66
- Controlling Element Position 66
 - Controlling Element Position with CSS Styles 66
 - Controlling the Z-Order Position 67
 - Controlling Element Position with Tables 68
 - Controlling Element Position with Transparent Images 70
- Cascading Style Sheets 71
 - What Cascading Style Sheets Do 71
 - Controlling Appearance with Styles 71
 - Types of CSS Styles 73
 - Overriding CSS Styles 73
 - Referencing External Style Sheets 74
 - Using Classes in Style Sheets 75
 - Control Hyperlink Appearance 75
 - Final Words about CSS Styles 75
- Summary 75

| | |
|--|-----------|
| Chapter 3 • Brief Guide to Dynamic Web Applications | 77 |
| What Is a Dynamic Web Application? | 78 |
| What Are Data-Driven Applications? | 79 |
| Why Should I Use HTML Templates and CSS? | 80 |
| Client Interaction | 81 |
| Web Applications versus Web Sites | 82 |
| What Is the Data and Where Is It? | 82 |
| Databases | 82 |
| HTML Files | 83 |
| Flat (ASCII) Files | 83 |
| XML Files | 83 |
| How Can You Retrieve the Data? | 84 |
| ADO.NET | 84 |
| XML Parsers | 84 |
| C# File/Stream Processing | 84 |
| In-Memory Storage | 85 |
| How Do You Package the Data? | 86 |
| HTML to Client | 86 |
| XML Data to Client | 87 |
| SOAP Requests/Responses | 88 |
| Custom Data Streams | 88 |
| What Is the User Interface? | 88 |
| Browser-Based Interfaces | 89 |
| .NET Windows Forms Interfaces | 89 |
| Client Capabilities | 89 |
| Summary | 90 |
| | |
| Part 2 • Server-Side Web Programming with Visual C# | 91 |
| | |
| Chapter 4 • Introduction to ASP.NET | 93 |
| What Is ASP.NET? | 93 |
| Framework for Processing HTTP Requests | 95 |
| Classic ASP versus ASP.NET | 96 |
| Why Do You Need ASP.NET? | 97 |
| Client Changes | 98 |
| Centralized Web-Based Applications | 98 |
| Distributed Web-Based Applications | 98 |
| Functional Interoperability | 99 |
| Standardization, Verifiability, and HTTP Affinity | 99 |
| Web Services | 99 |
| What Does ASP.NET Do? | 100 |
| Accepts Requests | 100 |
| Builds Responses | 100 |
| Assists with State Maintenance | 100 |
| Why Is ASP.NET in a C# Web Book? | 101 |
| C# Provides Code-Behind | 101 |
| .NET's Unified Platform for Design, Coding, and Debugging | 102 |

| | |
|---|------------|
| Creating Your First Web Form | 104 |
| Step 1: Creating a Project | 104 |
| Step 2: Laying Out the Page | 106 |
| Step 3: Writing the Code Behind the Page | 107 |
| Step 4: Viewing the Results in a Browser | 109 |
| Summary | 111 |
| Chapter 5 • Introduction to Web Forms | 113 |
| Web Forms Are Server-Side Objects | 113 |
| Creating Web Forms with Notepad | 114 |
| Code Render Blocks—Mix Code and HTML | 115 |
| Web Forms' Compatibility with Existing ASP Pages | 115 |
| Examples of ASP.NET Pages | 116 |
| ASP.NET Ships with 45 Web Controls | 118 |
| runat="server" | 120 |
| Form Validation | 122 |
| Client-Side Validation | 123 |
| Server-Side Validation | 130 |
| Code-Behind Programming | 132 |
| Use of Any .NET Language | 134 |
| Only One Language per Page | 134 |
| Display Separately from Data Processing | 134 |
| Page/Form Layout | 135 |
| Designing HTML Pages | 135 |
| Inline versus GridLayout | 135 |
| Generated HTML Code | 135 |
| Summary | 137 |
| Chapter 6 • Introduction to the System.Web Namespace | 139 |
| Obtaining Information from the Client | 139 |
| Accepting Requests | 140 |
| A Brief Introduction to the HttpResponse Object | 140 |
| The HttpRequest Object | 141 |
| Sending Information to the Client | 165 |
| HttpResponse Object Properties | 165 |
| HttpResponse Object Methods | 170 |
| The HttpApplication and HttpApplicationState Objects | 173 |
| Web Applications Are Multiuser Programs | 173 |
| Introduction to Threads—How IIS Handles Simultaneous Requests | 174 |
| Application Variables—Sharing Data between ASP.NET Instances | 174 |
| The global.asax File | 178 |
| Other HttpApplication Events | 179 |
| The HttpApplication.Error Event | 182 |
| Error Settings in the web.config File | 183 |
| When to Use Application Variables | 185 |
| Summary | 186 |

| | |
|---|------------|
| Chapter 7 • The SessionState Object | 187 |
| Introduction to Session State | 187 |
| What Is a Session? | 188 |
| Why Do You Need State? | 189 |
| Cookies, URL Munging, and the ASP.NET Session Object | 190 |
| Associating Data with a Browser Instance | 190 |
| Associating Data with Cookies | 191 |
| Associating Data with Munged URLs | 193 |
| Associating Data with Hidden Form Variables | 196 |
| Session Variables | 197 |
| Adding/Modifying Session Variables | 197 |
| C# Is Free of Threading Issues | 201 |
| Clearing Session Variables | 201 |
| Determining Whether a Session Variable Exists | 201 |
| ASP.NET Session State | 202 |
| Associating Data with Individuals | 202 |
| Identifying Users via Username/Password or Another Recognition Mechanism | 203 |
| ASP.NET Session State Security | 203 |
| ASP.NET Session State and Scalability | 204 |
| Comparing Options for Storing State | 205 |
| Summary | 214 |
| | |
| Chapter 8 • The HttpServerUtility Object | 215 |
| CreateObject and CreateObjectFromClsID Methods | 215 |
| Executing External Code with Server Object Methods and the #INCLUDE Command | 217 |
| Using the #INCLUDE Command in C# Web Applications | 217 |
| The Server.MapPath Method | 218 |
| The Server.Execute and Transfer Methods | 220 |
| Advanced Use of the Server.Execute Method | 223 |
| The Server.HtmlEncode, HtmlDecode, UrlEncode, and UrlDecode Methods | 226 |
| Using the Server.HtmlEncode Method | 227 |
| Using the Server.UrlEncode Method | 230 |
| The Server.GetLastError and ClearError Methods | 232 |
| Summary | 233 |
| | |
| Chapter 9 • Debugging ASP.NET and Error Handling | 235 |
| Setting Up a Site for Debugging | 235 |
| Check web.config | 236 |
| Check Debug Mode versus Release Mode | 236 |
| Enable Project Debugging | 237 |
| Keyboard Mapping | 238 |
| Using the Debugger | 238 |
| Setting Breakpoints | 238 |
| Breakpoint Properties | 239 |
| Stepping Through Code | 241 |

ASP.NET Event Sequence 242
 Using Trace Output 248
 Tracing from Code 253
 Defensive Coding: How to Avoid Errors 255
 Approaching Error Handling 256
 Writing Custom Error Pages 265
 Summary 266

Chapter 10 • File and Event Log Access with ASP.NET 267

Accessing Files 267
 Working with Files 268
 Checking Whether a File Exists 271
 Reading Files 271
 Using the FileInfo Class 273
 Using the StreamWriter Class 278
 Using the Directory and DirectoryInfo Classes 280
 Creating Files and Directories 283
 Page Counter Example 283
 Writing to Files 286
 Writing Text to Files 287
 Reading and Writing Binary Files 287
 Writing Data to Memory 288
 Writing HTML 289
 Sending and Receiving Files 294
 Sending (Downloading) Files 295
 Receiving Files 299
 Accessing the Windows Registry 302
 Summary 310

Chapter 11 • Sending and Receiving Messages with ASP.NET 311

Message Types 311
 Filesystem Messages 312
 Sending E-mail Messages 327
 Calling Methods in Another Web Form 338
 Retrieving Data from Another Web Page 342
 Summary 343

Part 3 • Accessing Data with ASP.NET 345

Chapter 12 • Introduction to Relational Databases and SQL 347

Databases versus Files 347
 Transactions and Logging 352
 Tables, Indexes, Primary Keys, and Foreign Keys 353
 Set-Based Data Retrieval 357
 Creating the ClassRecords Database 359

| | |
|--|------------|
| Introduction to SQL | 359 |
| The SQL SELECT Statement | 360 |
| INNER and OUTER JOIN Statements | 361 |
| Calculated Values and the GROUP BY Clause | 363 |
| The INSERT Statement | 365 |
| The UPDATE Statement | 366 |
| The DELETE Statement | 367 |
| Generate Unique IDs | 368 |
| Summary | 368 |
| Chapter 13 • Introduction to ADO.NET | 369 |
| The System.Data Namespace | 369 |
| Working with Connection Objects | 369 |
| The ADO.NET Approach to Data Access | 387 |
| From Recordsets to XML | 388 |
| The Command Object | 391 |
| The DataReader Object | 392 |
| The DataSet Object | 397 |
| Summary | 398 |
| Chapter 14 • Accessing Data | 399 |
| Deterministic Finalization | 399 |
| Call Dispose | 401 |
| Retrieve Only the Data You Need | 403 |
| Introduction to Stored Procedures | 405 |
| Creating Stored Procedures | 405 |
| Retrieving Data Using Stored Procedures | 411 |
| Creating and Using Parameterized Stored Procedures | 414 |
| Improving Your Data Access Code | 420 |
| The DataList and Repeater Web Controls | 424 |
| Creating a Web Server Control Formatting Template | 425 |
| Deleting, Updating, and Inserting Data | 431 |
| How Not to Write an Application | 447 |
| Summary | 449 |
| Chapter 15 • Using XML in Web Applications | 451 |
| Introduction to the System.Xml Namespace | 451 |
| What Can You Do with XML? | 452 |
| Reading XML Documents | 460 |
| DOM versus SAX | 460 |
| Using XmlReader Classes | 461 |
| Querying XML Documents for Data | 469 |
| Using the XmlDataDocument Class | 470 |
| The XmlException Classes | 473 |
| Performing XSLT Transforms Programmatically | 474 |
| Moving to Applications | 481 |
| Summary | 482 |

| | |
|---|------------|
| Part 4 • C# Web Applications | 483 |
| Chapter 16 • Introduction to C# Web Applications | 485 |
| Applications versus Web Sites | 485 |
| The web.config and machine.config Files Revisited | 492 |
| Is the web.config File Required? | 492 |
| The .config File Hierarchy | 493 |
| Site Hierarchy versus Directory Hierarchy | 500 |
| Inside Configuration Files | 500 |
| Creating Custom Configuration Sections | 502 |
| Creating a Custom Configuration Handler | 506 |
| Configuration File Location and Lock Settings | 508 |
| Location-Specific Tags Override Higher-Level Settings | 509 |
| Summary | 511 |
| Chapter 17 • State Maintenance and Cacheing | 513 |
| State Maintenance Options | 513 |
| Maintaining State in Cookies | 515 |
| Maintaining State in Session Variables | 522 |
| Maintaining State in Files | 524 |
| Maintaining State in Database Tables | 531 |
| Using the Cache Object | 535 |
| Cacheing ASP.NET Pages | 541 |
| Cache Location Attribute | 543 |
| Cache VaryByCustom Attribute | 544 |
| Cache VaryByControl Attribute | 544 |
| Cache VaryByHeader Attribute | 544 |
| Cacheing Data Programmatically | 544 |
| Partial Page Cacheing | 545 |
| A Multitude of Choices | 546 |
| Summary | 547 |
| Chapter 18 • Controlling Access and Monitoring | 549 |
| Implementing Username/Password Security | 550 |
| Using Basic Authentication | 552 |
| Using Custom Authentication | 556 |
| Using ASP.NET Forms Authentication | 560 |
| Securing Your Server with SSL | 566 |
| Monitoring Your Site—Beyond Page Counting | 570 |
| Tracking Usage | 571 |
| Advanced Page Counting | 571 |
| Knowing Who's in Your Application | 577 |
| Disabling an Application for Maintenance | 577 |
| Summary | 578 |

| | |
|---|------------|
| Chapter 19 • Planning Applications | 579 |
| Imagine Something | 580 |
| Determine the Audience | 580 |
| Determine the Application Requirements | 581 |
| Choose a Design Methodology | 581 |
| Use Terminology Appropriately | 582 |
| Test Often | 582 |
| Consider All Facets of the Application | 582 |
| Create and Sell a Vision | 584 |
| Plan the User Interface | 585 |
| Goals for User Interface Planning | 586 |
| Building Prototypes | 586 |
| Plan Data Storage and Retrieval | 587 |
| Plan to Archive or Delete Unwanted Data | 587 |
| Plan Object Responsibilities and Interfaces | 588 |
| Plan Administrative Functions | 588 |
| Create the Database | 589 |
| Plan Data Relationships Carefully | 589 |
| Plan a Data Programming Interface | 590 |
| Consider Database Administration | 590 |
| Data Size Matters | 590 |
| Create Data-Access Components | 591 |
| Create Business Components | 591 |
| Build the User Interface | 593 |
| Summary | 594 |
| | |
| Part 5 • Advanced Visual C# Web Applications | 595 |
| | |
| Chapter 20 • Leveraging Browser Clients | 597 |
| Which Client-Side Script Makes Sense? | 597 |
| Why Use Client Script? | 599 |
| Sending Script to the Browser | 599 |
| Writing Client Script Directly into an ASPX File | 600 |
| Writing Client Script with Server Script | 601 |
| Writing Client Script from a Code-Behind Class | 604 |
| Passing Values from Server Code to Client Code | 605 |
| Referencing Script Files on the Server | 609 |
| Simplifying the Special Offers Example | 615 |
| The Document Object Model | 622 |
| Accessing the DOM from Script | 624 |
| Using ActiveX Controls | 627 |
| Sending ActiveX Objects to the Client | 629 |
| Accessing ActiveX Objects from Client-Side Script | 630 |
| Summary | 633 |

| | |
|--|------------|
| Chapter 21 • Web Services | 635 |
| Introduction to Web Services | 635 |
| Remote Method Calls and XML | 635 |
| Interoperate Across Applications/Platforms | 637 |
| When Should You Use Web Services? | 637 |
| How Does VS.NET Help? | 638 |
| Build a Web Service | 640 |
| Web Service Description Language (WSDL) | 645 |
| Consume a Web Service | 651 |
| Consuming Web Services from IE | 651 |
| Consuming Web Services from a .NET Windows Forms Application | 655 |
| SOAP (Simple Object Access Protocol) | 660 |
| What Is SOAP? | 661 |
| Finding Web Services (UDDI) | 665 |
| Summary | 666 |
| | |
| Chapter 22 • Web Services, COM Components, and the SOAP Toolkit | 667 |
| Why Worry about Older Technology? | 668 |
| The Microsoft SOAP Toolkit | 668 |
| Create a Client Application | 669 |
| Initializing the SoapClient | 670 |
| Create a Server Application | 676 |
| Build a VB6 ActiveX DLL | 676 |
| Build a .NET Web Service Wrapper | 678 |
| Testing the COM-Based Web Service | 680 |
| Summary | 681 |
| | |
| Chapter 23 • Build Your Own Web Controls | 683 |
| Build a User Control | 683 |
| Loading User Controls at Runtime | 691 |
| Partial Page Caching and User Controls | 691 |
| Build a Composite Control | 692 |
| Customize the Toolbox | 698 |
| Test the LabeledTextBox Composite Control | 699 |
| Build a Custom Server Control | 700 |
| Create the DataTable Custom Control | 700 |
| Transferring Data Between ASP.NET Web Forms and Components | 710 |
| Summary | 710 |
| | |
| Chapter 24 • Efficiency and Scalability | 711 |
| Always Consider Efficiency | 712 |
| Am I Using the Framework Appropriately? | 712 |
| Response Time versus Development Cost | 717 |
| HTML | 717 |
| Database Access | 717 |
| Web Forms | 718 |
| Session Data | 719 |

| | |
|---|------------|
| Hardware versus Software | 719 |
| Scaling Up | 719 |
| Scaling Out | 720 |
| Tips for Efficiency and Scalability | 720 |
| Move Processing into Components | 721 |
| Avoid Large Pages | 721 |
| Remove Dead Code | 721 |
| Avoid Extra Trips to the Server | 721 |
| Summary | 722 |
| Afterword | 723 |
| Web Development Is Time Intensive | 723 |
| Browser Incompatibilities Cause Problems | 723 |
| Browsers Are Great—But Not Always | 724 |
| XML Is Important | 724 |
| What Should You Study Now? | 725 |
| Part 6 • Appendices | 727 |
| Appendix A • Quick HTML Reference | 729 |
| Quick Tag Reference | 730 |
| HTML 4 Attribute Reference | 735 |
| Appendix B • JScript 5.5 Reference | 755 |
| JScript Intrinsic Objects | 755 |
| JScript Properties | 756 |
| JScript Methods | 758 |
| JScript Functions | 764 |
| JScript Statements | 764 |
| JScript Operators | 765 |
| JScript Microsoft-Specific Extensions | 769 |
| JScript Constants | 770 |
| JScript Runtime Error Constants | 771 |
| JScript Syntax Error Constants | 774 |
| JScript Feature Availability, by Version | 775 |
| <i>Index</i> | 785 |