Stephen Walther
Kevin Hoffman
Nate Dudek

ASP.NET 4

UNLEASHED

SAMS

Stephen Walther
Kevin Hoffman
Nate Dudek

ASP.NET 4

UNLEASHED

SAMS | 800 East 96th Street, Indianapolis, Indiana 46240 USA

ASP.NET 4 Unleashed

Copyright ©2011 by Pearson Education, Inc.

All rights reserved. No part of this book shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission from the publisher. No patent liability is assumed with respect to the use of the information contained herein. Although every precaution has been taken in the preparation of this book, the publisher and authors assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained herein.

ISBN-13: 978-0-672-33112-1

ISBN-10: 0-672-33112-8

Library of Congress Cataloging-in-Publication Data:

Walther, Stephen.

ASP.NET 4.0 unleashed / Stephen Walther, Kevin Hoffman, Nate Dudek.

p. cm.

Includes index.

ISBN 978-0-672-33112-1

1. Active server pages. 2. Web sites—Design. 3. Web site development. 4. Microsoft .NET. I. Hoffman, Kevin. II. Dudek, Nate. III. Title.

TK5105.8885.A26W3517 2011

006.7’882—dc22

2010034058

Printed in the United States of America

First Printing September 2010

Trademarks

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Sams Publishing cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Warning and Disclaimer

Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied. The information provided is on an “as is” basis. The authors and the publisher shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book.

Bulk Sales

Sams Publishing offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales. For more information, please contact

U.S. Corporate and Government Sales
1-800-382-3419
corpsales@pearsontechgroup.com

For sales outside of the U.S., please contact

International Sales
international@pearson.com

Editor-in-Chief
Karen Gettman

Executive Editor
Neil Rowe

Development Editor
Mark Renfrow

Managing Editor
Kristy Hart

Project Editors
Jovana San Nicolas-Shirley and
Alexandra Maurer

Copy Editor
Apostrophe Editing Services

Indexer
WordWise Publishing Services LLC

Proofreader
Debbie Williams

Technical Editor
J. Boyd Nolan

Publishing Coordinator
Cindy Teeters

Cover Designer
Gary Adair

Composition
Gloria Schurick

Contents at a Glance

Introduction 1

Part I Building ASP.NET Pages

1 Overview of the ASP.NET Framework 5

2 Using the Standard Controls 59

3 Using the Validation Controls 121

4 Using the Rich Controls 179

Part II Designing ASP.NET Websites

5 Designing Websites with Master Pages 237

6 Designing Websites with Themes 269

7 Creating Custom Controls with User Controls 299

Part III Performing Data Access

8 Overview of Data Access 337

9 Using the SqlDataSource Control 379

10 Using List Controls 439

11 Using the GridView Control 485

12 Using the DetailsView and FormView Controls 567

13 Using the Repeater and DataList Controls 623

14 Using the ListView and DataPager Controls 661

15 Using the Chart Control 689

16 Using the QueryExtender Control 711

Part IV: Building Components

17 Building Components 721

18 Using the ObjectDataSource Control 771

19 Building Data Access Components with ADO.NET 843

20 Data Access with LINQ to SQL 935

21 Data Access with WCF Data Services 1001

Part V Site Navigation

22 Using the Navigation Controls 1011

23 Using Site Maps 1083

24 Advanced Navigation 1123

25 Using the ASP.NET URL Routing Engine 1137

Part VI **Security**

26 Using the `Login` Controls 1147
27 Using ASP.NET Membership 1205

Part VII **Building ASP.NET Applications**

28 Maintaining Application State 1263
29 Caching Application Pages and Data 1335
30 Localizing Applications for Multiple Languages 1419
31 Working with the HTTP Runtime 1453
32 Building Dynamic Data Applications 1491
33 Building ASP.NET MVC Applications 1501
34 Configuring Applications 1511
35 Deploying ASP.NET Web Applications 1565

Part VIII **Custom Control Building**

36 Building Custom Controls 1575
37 Building Templated Databound Controls 1647

Part IX **ASP.NET AJAX**

38 Using Server-Side ASP.NET AJAX 1673
39 Using the ASP.NET AJAX Control Toolkit 1733
40 Client-Side AJAX with jQuery 1769

Index 1783

Table of Contents

Introduction 1

Part I Building ASP.NET Pages

1 Overview of the ASP.NET Framework 5

ASP.NET and the .NET Framework 9

Understanding the Framework Class Library 9

Understanding the Common Language Runtime 17

Understanding ASP.NET Controls 19

Overview of ASP.NET Controls 19

Understanding HTML Controls 21

Understanding and Handling Control Events 22

Understanding View State 28

Understanding ASP.NET Pages 34

Understanding Dynamic Compilation 34

Understanding Control Trees 38

Using Code-Behind Pages 39

Handling Page Events 44

Using the Page.IsPostBack Property 46

Debugging and Tracing ASP.NET Pages 48

Tracing Page Execution 52

Installing ASP.NET 55

Summary 57

2 Using the Standard Controls 59

Displaying Information 59

Using the Label Control 60

Using the Literal Control 65

Accepting User Input 68

Using the TextBox Control 68

Using the CheckBox Control 76

Using the RadioButton Control 79

Submitting Form Data 84

Using the Button Control 84

Using the LinkButton Control 86

Using the ImageButton Control 89

Using Client Scripts with Button Controls 94

Performing Cross-Page Posts 96

Specifying a Default Button 100

Handling the Command Event 101

Displaying Images 104
Using the Image Control 104
Using the ImageMap Control 106
Using the Panel Control 111
Using the HyperLink Control 118
Summary 119

3 Using the Validation Controls 121

Overview of the Validation Controls 121
Validation Controls and JavaScript 126
Using Page.IsValid 126
Setting the Display Property 127
Highlighting Validation Errors 128
Using Validation Groups 133
Disabling Validation 138
Using the RequiredFieldValidator Control 139
Using the RangeValidator Control 143
Using the CompareValidator Control 146
Using the RegularExpressionValidator Control 151
Using the CustomValidator Control 154
Using the ValidationSummary Control 162
Creating Custom Validation Controls 167
Creating a LengthValidator Control 168
Creating an AjaxValidator Control 170
Summary 177

4 Using the Rich Controls 179

Accepting File Uploads 180
Saving Files to the File System 181
Saving Files to a Database 185
Uploading Large Files 189
Displaying a Calendar 195
Creating a Pop-Up Date Picker 199
Rendering a Calendar from a Database Table 202
Displaying Advertisements 207
Storing Advertisements in an XML File 208
Storing Advertisements in a Database Table 212
Tracking Impressions and Transfers 213
Displaying Different Page Views 220
Displaying a Tabbed Page View 220
Displaying a Multipart Form 223
Displaying a Wizard 226
Displaying Silverlight Content 231
Summary 234

Part II Designing ASP.NET Websites

5 Designing Websites with Master Pages 237
Creating Master Pages..238
Creating Default Content...242
Nesting Master Pages 245
Using Images and Hyperlinks in Master Pages 251
Registering Master Pages in Web Configuration.............................254
Modifying Master Page Content..255
Using the Title Attribute...255
Using the Page Header Property..256
Exposing Master Page Properties...258
Using FindControl with Master Pages 261
Loading Master Pages Dynamically...263
Loading Master Pages Dynamically for Multiple Content Pages 266
Summary 268

6 Designing Websites with Themes...269
Creating Themes 270
Adding Skins to Themes 270
Creating Named Skins 273
Themes Versus StyleSheetThemes 275
Disabling Themes 277
Registering Themes in the Web Configuration File........................280
Adding Cascading Style Sheets to Themes...281
Adding Multiple Cascading Style Sheets to a Theme 285
Changing Page Layouts with Cascading Style Sheets.....................285
Creating Global Themes 290
Applying Themes Dynamically 291
Applying Skins Dynamically 293
Summary 298

7 Creating Custom Controls with User Controls 299
Creating User Controls 300
Registering User Controls in the Web Configuration File...............303
Exposing Properties from a User Control 304
Exposing Events from a User Control..307
Creating an AddressForm Control 311
AJAX and User Controls 317
Dynamically Loading User Controls 320
Using the Reference Directive 322
Creating a Multipage Wizard 327
Summary 336

Part III Performing Data Access

8 Overview of Data Access 337

Using DataBound Controls 337

Working with List Controls 338

Working with Tabular DataBound Controls 341

Working with Hierarchical DataBound Controls 345

Working with Other Controls 348

Using DataSource Controls 351

Using ASP.NET Parameters with DataSource Controls 353

Using Programmatic DataBinding 356

Understanding Templates and DataBinding Expressions 358

Using Templates 359

Using DataBinding Expressions 362

Using Two-Way DataBinding Expressions 366

Overview of SQL Server 2008 Express 369

Features of SQL Server Express 369

SQL Server 2008 Express Management Tools 370

Server Databases Versus Local Databases 371

Sample Database-Driven Web Application 375

Summary 378

9 Using the SqlDataSource Control 379

Creating Database Connections 380

Connecting to Microsoft SQL Server 380

Connecting to Other Databases 384

Storing Connection Strings in the Web Configuration File 386

Encrypting Connection Strings 388

Executing Database Commands 389

Executing Inline SQL Statements 389

Executing Stored Procedures 392

Filtering Database Rows 395

Changing the Data Source Mode 397

Handling SQL Command Execution Errors 398

Canceling Command Execution 403

Using ASP.NET Parameters with the SqlDataSource Control 406

Using the ASP.NET Parameter Object 406

Using the ASP.NET ControlParameter Object 410

Using the ASP.NET CookieParameter Object 415

Using the ASP.NET FormParameter Object 417

Using the ASP.NET ProfileParameter Object 420

Using the QueryStringParameter Object 422

Using the SessionParameter Object 424

Programmatically Executing SqlDataSource Commands....426
Adding ADO.NET Parameters....426
Executing Insert, Update, and Delete Commands....428
Executing Select Commands....430
Caching Database Data with the SqlDataSource Control....435
Summary....438

10 Using List Controls....439
Overview of the List Controls....439
Declaring List Items....439
Binding to a Data Source....442
Determining the Selected List Item....446
Appending Data Items....450
Enabling Automatic PostBacks....452
Using the Items Collection....454
Working with the DropDownList Control....458
Working with the RadioButtonList Control....461
Working with the ListBox Control....463
Working with the CheckBoxList Control....467
Working with the BulletedList Control....470
Creating a Custom List Control....475
Summary....484

11 Using the GridView Control....485
GridView Control Fundamentals....486
Displaying Data....486
Selecting Data....489
Using Data Keys....492
Sorting Data....495
Paging Through Data....501
Editing Data....510
Displaying Empty Data....514
Formatting the GridView Control....519
Using ViewState with the GridView Control....521
Using Fields with the GridView Control....522
Using BoundFields....523
Using CheckBoxFields....526
Using CommandFields....527
Using Button Fields....530
Using HyperLink Fields....534
Using ImageFields....539
Using TemplateFields....543
Working with GridView Control Events....546
Highlighting GridView Rows....547

Displaying Column Summaries..550
Displaying Nested Master/Details Forms 552
Extending the GridView Control 555
Creating a LongTextField..556
Creating a DeleteButtonField 560
Creating a ValidatedField 563
Summary 566

12 Using the DetailsView and FormView Controls 567

Using the DetailsView Control...567
Displaying Data with the DetailsView Control 567
Using Fields with the DetailsView Control.....................................571
Displaying Empty Data with the DetailsView Control....................573
Paging through Data with the DetailsView Control 576
Updating Data with the DetailsView Control 584
Inserting Data with the DetailsView Control 593
Deleting Data with the DetailsView Control..................................597
Working with DetailsView Control Events......................................598
Formatting the DetailsView Control...601
Using the FormView Control 604
Displaying Data with the FormView Control.................................605
Paging Through Data with the FormView Control 607
Editing Data with the FormView Control.......................................612
Inserting Data with the FormView Control....................................616
Deleting Data with the FormView Control.....................................619
Summary 622

13 Using the Repeater and DataList Controls 623

Using the Repeater Control..623
Displaying Data with the Repeater Control...................................624
Using Templates with the Repeater Control...................................627
Handling Repeater Control Events...633
Using the DataList Control...640
Displaying Data with the DataList Control 640
Displaying Data in Multiple Columns 644
Using Templates with the DataList Control 645
Selecting Data with the DataList Control 648
Editing Data with the DataList Control..651
Formatting the DataList Control..657
Summary 660

14 Using the ListView and DataPager Controls 661

Using the ListView Control 661
Using the LayoutTemplate and ItemTemplate................................662
Using the GroupTemplate 666

Selecting a Row 669

Sorting Database Data 672

Editing Database Data 675

Using the DataPager Control..681

Creating a Custom User Interface for Paging 684

Data Source Paging with the DataPager Control..................................686

Summary 688

15 Using the Chart Control..689

Chart Control Fundamentals 690

Displaying Data with the Chart Control 690

Sorting and Filtering Data 694

Using Statistical Formulas 698

Customizing a Chart's Appearance 704

Customizing the Background and Plotting Area 704

Three-Dimensional Charts 708

Drill-Down Reports 709

Summary 710

16 Using the QueryExtender Control 711

Introduction to the QueryExtender Control..711

Querying Data with Declarative Syntax..712

Querying with the ControlFilterExpression..712

Querying with the CustomExpression..712

Querying with the DynamicFilterExpression 713

Querying with the MethodExpression..713

Sorting with the OrderByExpression..714

Querying with the PropertyExpression..715

Querying with the RangeExpression..715

Querying with the SearchExpression 716

Building a Sample Page with the QueryExtender Control 717

Summary 720

Part IV: Building Components

17 Building Components..721

Building Basic Components 722

Components and Dynamic Compilation 724

Mixing Different Language Components in the App_Code Folder 725

Declaring Methods 726

Declaring Fields and Properties..728

Declaring Constructors..733

Overloading Methods and Constructors..734

Declaring Namespaces 736

Creating Partial Classes 738
Inheritance and Abstract Classes....739
Declaring Interfaces....742
Using Access Modifiers 743
Intellisense and Components....744
Using ASP.NET Intrinsics in a Component....746
Building Component Libraries 749
Compiling Component Libraries 750
Adding a Reference to a Class Library....753
Architectural Considerations....757
Building Multitier Applications....757
Creating the User Interface Layer 758
Creating the Business Logic Layer....761
Creating the Data Access Layer 765
Summary 769

18 Using the ObjectDataSource Control....771

Representing Objects with the ObjectDataSource Control....772
Binding to a Component 772
Binding to a DataReader....774
Binding to a DataSet....776
Binding to a LINQ to SQL Query 778
Binding to a Web Service 779
Using Parameters with the ObjectDataSource Control....783
Using Different Parameter Types....787
Passing Objects as Parameters 790
Paging, Sorting, and Filtering Data with the ObjectDataSource Control....795
User Interface Paging....795
Data Source Paging 798
User Interface Sorting....805
Data Source Sorting 808
Filtering Data 813
Handling ObjectDataSource Control Events....817
Adding and Modifying Parameters 818
Handling Method Errors....822
Handling the ObjectCreating Event....826
Concurrency and the ObjectDataSource Control 828
Extending the ObjectDataSource Control....832
Creating a Custom ObjectDataSource Control....832
Creating Custom Parameter Objects....835
Summary 841

19 Building Data Access Components with ADO.NET 843
Connected Data Access 844
Using the Connection Object 848
Using the Command Object 859
Using the DataReader Object 881
Disconnected Data Access 888
Using the DataAdapter Object 889
Using the DataTable Object 896
Using the DataView Object 906
Using the DataSet Object 908
Executing Asynchronous Database Commands...911
Using Asynchronous ADO.NET Methods 912
Using Asynchronous ASP.NET Pages 914
Building Database Objects with the .NET Framework..............................918
Enabling CLR Integration...918
Creating User-Defined Types with .NET Framework..919
Building a Data Access Layer with a User-Defined Type 924
Creating Stored Procedures with .NET Framework..928
Creating the Stored Procedure Assembly 928
Summary 934
20 Data Access with LINQ to SQL...935
New C# and VB.NET Language Features 936
Understanding Automatic Properties...936
Understanding Initializers 937
Understanding Type Inference 939
Understanding Anonymous Types...940
Understanding Generics 941
Understanding Lambda Expressions 944
Understanding Extension Methods 948
Understanding LINQ 949
Creating LINQ to SQL Entities 951
Building Entities by Hand 952
Building Entities with the LINQ to SQL Designer 956
Building Entity Associations..959
Using the LinqDataSource Control 960
Performing Standard Database Commands with LINQ to SQL................964
LINQ to Objects Versus LINQ to SQL 964
Selecting with LINQ to SQL 965
Inserting with LINQ to SQL 973
Updating with LINQ to SQL...976
Deleting with LINQ to SQL..981

Dynamic Queries 983
Debugging LINQ to SQL....987
Creating a Custom LINQ Entity Base Class 991
Using the Entity Base Class 991
Performing Standard Data-Access Operations with the EntityBase Class....992
Performing Validation with the EntityBase Class....994
Summary 1000

21 Data Access with WCF Data Services....1001
Overview of WCF Data Services 1001
Using Data Services with a Service Reference 1002
Using Data Services with a Data Context 1007
Summary 1009

Part V Site Navigation

22 Using the Navigation Controls 1011
Understanding Site Maps....1011
Using the SiteMapPath Control....1013
Formatting the SiteMapPath Control 1016
Using the Menu Control 1020
Declaratively Adding Menu Items....1021
Using the Menu Control with the MultiView Control 1024
Binding to a Site Map....1027
Binding to an XML File....1031
Binding to Database Data....1035
Formatting the Menu Control 1040
Using Templates with the Menu Control 1046
Using the TreeView Control 1049
Declaratively Adding Tree Nodes 1049
Displaying Check Boxes with the TreeView Control....1054
Binding to a Site Map....1057
Binding to an XML File....1058
Binding to Database Data....1061
Using Populate On Demand and AJAX 1064
Formatting the TreeView Control 1070
Building a SQL Hierarchical Data Source Control 1077
Summary 1081

23 Using Site Maps....1083
Using the SiteMapDataSource Control....1084
Setting SiteMapDataSource Properties 1086
Using the SiteMap Class 1090
Using the SiteMapNode Class 1093

Advanced Site Map Configuration 1095
Using Security Trimming 1096
Merging Multiple Site Maps 1101
Creating Custom Site Map Attributes 1103
Creating Custom Site Map Providers 1105
Creating the AutoSiteMapProvider 1106
Creating the SqlSiteMapProvider 1111
Generating a Google SiteMap File 1116
Summary 1120

24 Advanced Navigation 1123
Remapping URLs 1124
Creating a Custom UrlRemapper Module 1127
Using the VirtualPathProvider Class 1132
Limitations of the VirtualPathProvider Class 1133
Understanding the VirtualPathProvider Class 1133
Registering a VirtualPathProvider Class 1135
Summary 1136

25 Using the ASP.NET URL Routing Engine 1137
Introduction to URL Routing 1137
Basic URL Routing Scenarios 1139
Mapping Basic URLs 1139
Mapping URLs with Parameters 1140
Mapping URLs with Multiple Segments 1141
Linking to Other Pages with Routes 1142
Advanced URL Routing 1143
Using Routes with Default Parameters 1143
Using Constrained Routes 1144
Security Concerns with Routes 1146
Summary 1146

Part VI Security

26 Using the Login Controls 1147
Overview of the Login Controls 1148
Using the Login Control 1152
Automatically Redirecting a User to the Referring Page 1155
Automatically Hiding the Login Control from Authenticated Users 1155
Using a Template with the Login Control 1157
Performing Custom Authentication with the Login Control 1160
Using the CreateUserWizard Control 1162
Configuring Create User Form Fields 1163
Sending a Create User Email Message 1166

Automatically Redirecting a User to the Referring Page.................1169
Automatically Generating a Password 1171
Using Templates with the CreateUserWizard Control....................1177
Adding Steps to the CreateUserWizard Control 1181
Using the LoginStatus Control 1183
Using the LoginName Control 1185
Using the ChangePassword Control...1186
Sending a Change Password Email 1188
Using Templates with the ChangePassword Control 1190
Using the PasswordRecovery Control 1193
Sending the Original Password 1195
Requiring a Security Question and Answer 1196
Using Templates with the PasswordRecovery Control 1197
Using the LoginView Control...1199
Using Roles with the LoginView Control 1201
Summary 1204

27 Using ASP.NET Membership..1205

Configuring Authentication 1206
Configuring Forms Authentication..1207
Using Cookieless Forms Authentication 1209
Using Sliding Expiration with Forms Authentication 1210
Using Forms Authentication Across Applications 1210
Using Forms Authentication Across Domains 1214
Using the FormsAuthentication Class..1216
Using the User Class 1220
Configuring Authorization 1221
Authorizing by Role..1223
Authorizing Files by Location 1224
Using Authorization with Images and Other File Types 1225
Using ASP.NET Membership 1226
Using the Membership Application Programming Interface 1227
Encrypting and Hashing User Passwords.......................................1232
Modifying User Password Requirements.......................................1234
Locking Out Bad Users 1235
Configuring the SQLMembershipProvider 1237
Configuring the ActiveDirectoryMembershipProvider..................1239
Creating a Custom Membership Provider 1244
Using the Role Manager 1246
Configuring the SqlRoleProvider 1247
Configuring the WindowsTokenRoleProvider 1252
Configuring the AuthorizationStoreRoleProvider 1254

Caching Roles in a Browser Cookie 1256
Using the Roles Application Programming Interface 1258
Summary 1260

Part VII Building ASP.NET Applications

28 Maintaining Application State..1263
Using Browser Cookies 1264
Cookie Security Restrictions..1265
Creating Cookies 1266
Reading Cookies 1269
Setting Cookie Properties 1272
Deleting Cookies..1273
Working with Multivalued Cookies 1275
Using Session State..1279
Storing Database Data in Session State 1281
Using the Session Object..1284
Handling Session Events 1285
Controlling When a Session Times Out 1287
Using Cookieless Session State 1288
Configuring a Session State Store..1290
Configuring SQL Server Session State 1294
Using Profiles 1297
Creating Profile Groups..1301
Supporting Anonymous Users..1303
Migrating Anonymous Profiles 1307
Inheriting a Profile from a Custom Class 1308
Creating Complex Profile Properties..1310
Saving Profiles Automatically..1317
Accessing Profiles from Components..1321
Using the Profile Manager..1323
Configuring the Profile Provider..1326
Creating a Custom Profile Provider 1327
Summary 1332

29 Caching Application Pages and Data 1335
Overview of Caching 1335
Using Page Output Caching 1337
Varying the Output Cache by Parameter......................................1338
Varying the Output Cache by Control 1342
Varying the Output Cache by Header..1345
Varying the Output Cache by Browser 1346
Varying the Output Cache by a Custom Function........................1347
Specifying the Cache Location..1349

Creating a Page Output Cache File Dependency 1351
Expiring the Page Output Cache Programmatically 1353
Manipulating the Page Output Cache Programmatically 1358
Creating Page Output Cache Profiles 1360
Using Partial Page Caching 1361
Using Post-Cache Substitution 1362
Caching with a User Control 1366
Sharing a User Control Output Cache 1370
Manipulating a User Control Cache Programmatically 1370
Creating a User Control Cache File Dependency 1372
Caching Dynamically Loaded User Controls 1373
Using DataSource Caching 1375
Using an Absolute Cache Expiration Policy 1376
Using a Sliding Cache Expiration Policy 1378
Caching with the ObjectDataSource Control 1380
Caching with the XmlDataSource Control 1382
Creating a DataSource Control Key Dependency 1383
Using Data Caching 1387
Using the Cache Application Programming Interface 1387
Adding Items to the Cache 1390
Adding Items with an Absolute Expiration Policy 1390
Adding Items with a Sliding Expiration Policy 1393
Adding Items with Dependencies 1394
Specifying Cache Item Priorities 1396
Configuring the Cache 1397
Using SQL Cache Dependencies 1399
Using Polling SQL Cache Dependencies 1399
Configuring Polling SQL Cache Dependencies 1400
Using Polling SQL Cache Dependencies with Page Output Caching 1403
Using Polling SQL Cache Dependencies with DataSource Caching 1405
Using Polling SQL Cache Dependencies with Data Caching 1406
Using Push SQL Cache Dependencies 1408
Configuring Push SQL Cache Dependencies 1409
Using Push SQL Cache Dependencies with Page Output Caching 1411
Using Push SQL Cache Dependencies with DataSource Caching 1412
Using Push SQL Cache Dependencies with Data Caching 1415
Summary 1416

30 Localizing Applications for Multiple Languages......................................1419
Setting the Current Culture..1420
Setting a Culture Manually 1421
Automatically Detecting a Culture 1428
Setting the Culture in the Web Configuration File 1432
Culture and ASP.NET Controls..1433
Using the CultureInfo Class 1434
Using the CultureInfo Class to Format String Values....................1435
Comparing and Sorting String Values...1437
Creating Local Resources 1438
Explicit Localization Expressions 1438
Implicit Localization Expressions..1441
Using Local Resources with Page Properties 1443
Retrieving Local Resources Programmatically...............................1444
Creating Global Resources...1446
Retrieving Global Resources Programmatically 1448
Using Strongly Typed Localization Expressions 1449
Using the Localize Control...1450
Summary 1452
31 Working with the HTTP Runtime 1453
Creating a Custom BuildProvider...1454
Creating a Simple BuildProvider 1454
Creating a Data Access Component BuildProvider 1458
Creating a Custom ExpressionBuilder..1464
Creating a Lookup ExpressionBuilder...1465
Creating HTTP Handlers...1470
Creating a Generic Handler..1470
Implementing the IHttpHandler Interface 1473
Creating an Asynchronous HTTP Handler 1478
Working with HTTP Applications and HTTP Modules 1482
Creating a Global.asax File...1483
Creating Custom HTTP Modules 1487
Summary 1489
32 Building Dynamic Data Applications..1491
Introducing ASP.NET Dynamic Data 1491
Building a Dynamic Data Application 1492
Working with Dynamic Data Templates...1496
Making Shared Template Changes..1497
Creating Type and Entity-Specific Templates 1497
Summary 1500

33 Building ASP.NET MVC Applications 1501
Introducing the ASP.NET MVC Framework 1501
Models 1502
Views 1502
Controllers 1502
Creating Your First ASP.NET MVC Application 1502
Building an MVC Page 1505
Accepting Form Input 1507
Summary 1509

34 Configuring Applications 1511
Overview of Website Configuration 1511
Using the Web Site Administration Tool 1513
Using the ASP.NET Microsoft Management Console Snap-In 1514
ASP.NET Configuration Sections 1515
Applying Configuration Settings to a Particular Path 1517
Locking Configuration Settings 1518
Adding Custom Application Settings 1520
Placing Configuration Settings in an External File 1522
Using the Configuration API 1523
Reading Configuration Sections from the Current Application 1524
Opening a Configuration File 1529
Opening a Configuration File on a Remote Server 1533
Using the Configuration Class 1536
Modifying Configuration Sections 1539
Provisioning a New Website 1543
Creating Custom Configuration Sections 1546
Creating a Configuration Element Collection 1551
Creating Encrypted Configuration Sections 1556
Encrypting Sections with the aspnet_regiis Tool 1557
Encrypting Sections Programmatically 1558
Deploying Encrypted Web Configuration Files 1561
Summary 1564

35 Deploying ASP.NET Web Applications 1565
Packaging Web Applications 1565
Using Web.config Transformations 1569
Deploying Databases 1571
One-Click Publishing Web Applications 1572
Summary 1573

Part VIII Custom Control Building

36 Building Custom Controls 1575

Overview of Custom Control Building 1575

Building Fully Rendered Controls 1576

Building Composite Controls 1587

Building Hybrid Controls 1590

View State and Control State 1596

Supporting View State 1596

Supporting Control State 1599

Processing Postback Data and Events 1602

Handling Postback Data 1602

Handling Postback Events 1606

Working with Control Property Collections 1618

Using the ParseChildren Attribute 1618

Using the AddParsedSubObject() Method 1626

Using a ControlBuilder 1626

Creating a Better Designer Experience 1631

Applying Design-Time Attributes to a Control 1631

Creating Control Designers 1636

Creating a Container ControlDesigner 1636

Adding Smart Tasks 1640

Summary 1645

37 Building Templated Databound Controls 1647

Creating Templated Controls 1647

Implementing the ITemplate Interface 1648

Creating a Default Template 1652

Supporting Simplified Databinding 1656

Supporting Two-Way Databinding 1660

Creating Templated Databound Controls 1666

Creating a DivView Control 1667

Summary 1672

Part IX ASP.NET AJAX

38 Using Server-Side ASP.NET AJAX 1673

The Ajax Vision 1674

Server-Side Ajax Versus Client-Side Ajax 1675

Debugging Ajax Applications 1675

Using the UpdatePanel Control 1677

Specifying UpdatePanel Triggers 1686

Nesting UpdatePanel Controls 1688

Updating UpdatePanels Programmatically 1692

UpdatePanels and JavaScript 1694
UpdatePanel Server-Side Page Execution Life Cycle 1697
UpdatePanel Client-Side Page Execution Life Cycle 1700
Canceling the Current Asynchronous Postback 1707
Aborting the Previous Asynchronous Postback 1710
Passing Additional Information During an Asynchronous Postback 1712
Handling UpdatePanel Errors Gracefully 1714
UpdatePanel Performance 1720
Using the Timer Control 1724
Using the UpdateProgress Control 1728
Summary 1731

39 Using the ASP.NET AJAX Control Toolkit 1733
Using the ASP.NET AJAX Control Toolkit 1734
Overview of the Toolkit Controls 1737
Using the AutoComplete Control 1739
Using the AutoCompleteExtender with a Page Method 1740
Using the AutoCompleteExtender with a Web Service Method 1743
Using Text and Value Pairs with the AutoCompleteExtender........1746
Using the DragPanel Control 1750
Using the FilteredTextBox Control 1754
Using the MaskedEdit Control 1756
Using the Animation Control 1759
Using the UpdatePanelAnimation Control 1764
Summary 1768

40 Client-Side Ajax with jQuery 1769
What Is jQuery? 1769
Using the jQuery Library 1770
Creating a jQuery File 1771
The $ Method and $(document).ready() 1773
jQuery Selectors 1774
Calling Web Services from the Client 1776
Calling an External Web Service 1776
Calling a Static Page Method 1780
Summary 1781

Index 1783

Introduction

Web development has been rapidly evolving over the years. The features and functionality that today's web developers are asked to produce are exponentially more involved and complex than they were just a few years ago. As the demands of today's business and commercial software grow, so too must the power of the tools and development frameworks developers use every day.

Even in the years since ASP.NET was first introduced, it has undergone dramatic growth in terms of ease of use, power, flexibility, scalability, and time to market. Some of the largest websites hosted on the Internet have ASP.NET and the .NET Framework to thank for their speed, power, and scalability including Dell, MySpace, and Microsoft.

Whether you plan on building the next greatest social network, a simple blogging site, or a year-long project to build a suite of Line of Business applications for the enterprise—ASP.NET might be the right tool for the job.

Who Should Read This Book?

ASP.NET 4 Unleashed is for professional programmers who need to create rich, interactive websites. This book is a comprehensive reference for building websites with all the tools and technology that are part of the ever-growing ASP.NET umbrella. There are hundreds of code samples on the accomanying website that you can use to immediately begin building your website.

If you are new to building websites with ASP.NET, you can use this book to learn everything you need to know to build a website with the ASP.NET Framework. If you are an experienced ASP.NET developer, you can use this book to refresh your memory on some lesser-used features and learn about the new features in ASP.NET 4.

What Do You Need to Know Before You Read This Book?

To get the most out of this book, you should have a decent familiarity with the core concepts of the .NET Framework. This book can be used as a reference for ASP.NET veterans as well as serve as a full, cover-to-cover learning experience for developers new to building ASP.NET web applications. The samples are designed to be clear and easy to read, regardless of whether your background is in VB.NET or C#.

To get the most from the database chapters, you should have some experience working with a database, such as Microsoft SQL Server, Oracle, or Microsoft Access. Purely to make the samples easier to install and test, the data-driven samples either work from data files or from SQL Server databases.

If you want to run every sample from this book, you should be running at least Windows Vista, Windows Server 2008, or Windows 7 for best results. You should also have the latest version of Internet Information Server (IIS) installed and some version of Visual Studio 2010 installed.

Changes to This Book

This edition of the book reflects many important transitions in the ASP.NET Framework. There are several new chapters in this book that cover features introduced in ASP.NET 4, such as a new charting control, a new URL routing engine, use of the ADO.NET Entity Framework and WCF Data Services and much more.

Another area that has seen large amounts of change and improvement in ASP.NET 4 is the use of client-side scripting such as Ajax. The entire Ajax section of this book has been completely redone since the previous version.

How This Book Is Organized

Although we encourage you to read this book from start to finish, reading chapter by chapter, some experienced ASP.NET developers might want to pick and choose chapters and skip the ones that are review. If necessary, you can use this book solely as a reference and jump to a chapter only when the need arises. It might be helpful, therefore, to have an idea of the overall organization of this book.

- **Part I: Building ASP.NET Pages**—Provides you with an overview of the basic controls included in the ASP.NET Framework. You learn how to build interactive Web Forms with the form controls. You also learn how to validate form data with the validation controls. Finally, you learn how to upload files and display interactive calendars and wizards with the rich controls.

- **Part II: Designing ASP.NET Websites**—Discusses how you can create a common layout and style for the pages on your website. You learn how to use Master Pages to share content across multiple pages. You also learn how to use Themes to create a consistent page style.

- **Part III: Performing Data Access**—Focuses on data access. You learn how to use the `ListView` and `GridView` controls to display, page, sort, and edit a set of database records. You learn how to use the `DetailsView` and `FormView` controls to display and edit a single database record at a time. WCF Data Services are introduced in this section.

- **Part IV: Building Components**—Focuses on building custom components. You learn how to design and create multitiered applications. You also learn how to build data access components by taking advantage of both LINQ to SQL and ADO.NET.

- **Part V: Site Navigation—**Discusses the various navigation controls included in the ASP.NET Framework, such as the `TreeView` and `Menu` controls. You learn how to use these controls with a Site Map to allow users to easily navigate a website. You also learn how to use the `VirtualPathProvider` class to abstract a website from the file system.
- **Part VI: Security—**Focuses on the Login controls and Membership API. You learn how to create a user registration and authentication system. You learn how to store Membership information in either a SQL Server database or Active Directory.
- **Part VII: Building ASP.NET Applications—**Discusses a variety of topics related to building ASP.NET applications. For example, you learn how to improve the performance of your ASP.NET applications by taking advantage of caching. You also learn how to localize your ASP.NET applications so that they can be easily translated and presented in multiple human languages.
- **Part VIII: Custom Control Building—**Concentrates on extending the ASP.NET Framework with custom controls. For example, you learn how to create custom data access controls that work like the `ListView` and `GridView` controls.
- **Part IX: ASP.NET AJAX—**Concentrates on extending the ASP.NET Framework with custom controls. For example, you learn how to create custom data access controls that work like the `ListView` and `GridView` controls.

NOTE

The book's website contains all of the code samples found in this book in C# and VB.NET. The code samples are posted online at www.informit.com/title/9780672331121 in the Books Section of the website.

This page intentionally left blank

CHAPTER 1

Overview of the ASP.NET Framework

IN THIS CHAPTER

- ASP.NET and the.NET Framework
- Understanding ASP.NET Controls
- Understanding ASP.NET Pages
- Installing ASP.NET
- Summary

Let's start by building a simple ASP.NET page.

NOTE

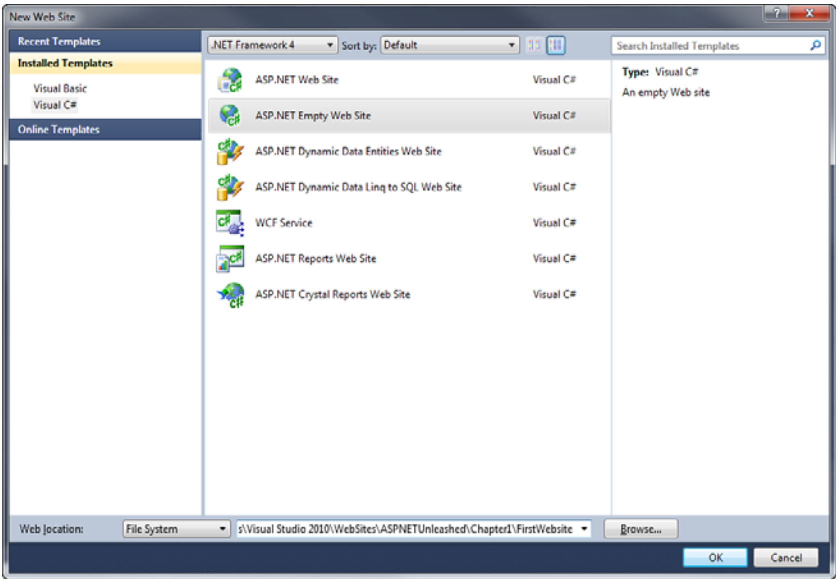
For information on installing ASP.NET, see the last section of this chapter.

If you use Visual Web Developer or Visual Studio, you first need to create a new website. Start Visual Web Developer and select File, New Web Site. The New Web Site dialog box appears (see Figure 1.1). Enter the folder in which you want your new website to be created (such as "Chapter1") in the Location field and click the OK button.

NOTE

When you create a new website, you might receive an error message warning you that you need to enable script debugging in Internet Explorer. You need to enable script debugging to build Ajax applications. We discuss Ajax later in the book.

After you create a new website, you can add an ASP.NET page to it. Select Web Site, Add New Item. Select Web Form and enter the value `FirstPage.aspx` in the Name field. Make sure that both the Place Code in Separate File and Select Master Page check boxes are unchecked, and click the Add button to create the new ASP.NET page (see Figure 1.2).



New Web Site
Recent Templates
Installed Templates
Visual Basic
Visual C#
Online Templates
Search Installed Templates
ASP.NET Web Site Visual C#
ASP.NET Empty Web Site Visual C#
ASP.NET Dynamic Data Entities Web Site Visual C#
ASP.NET Dynamic Data Linq to SQL Web Site Visual C#
WCF Service Visual C#
ASP.NET Reports Web Site Visual C#
ASP.NET Crystal Reports Web Site Visual C#
Browse...
OK
Cancel

.NET Framework 4

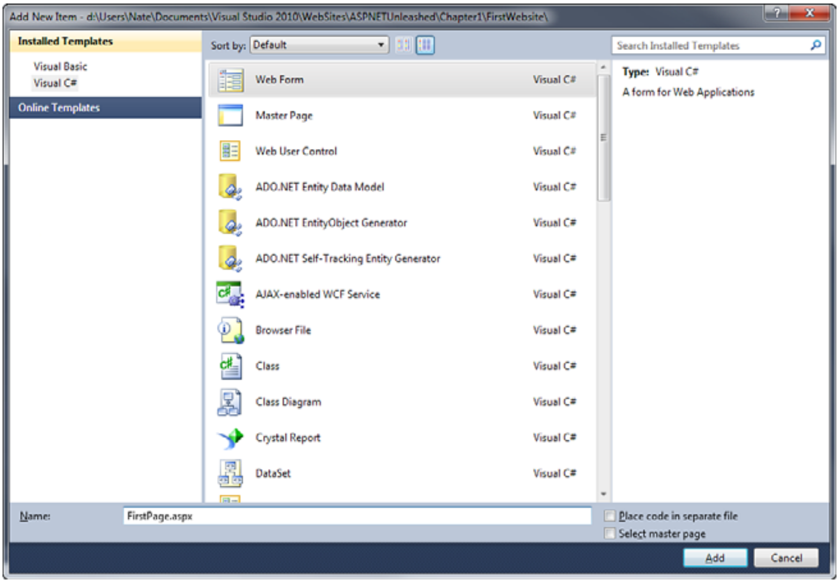
Sort by: Default

Type: Visual C#

An empty Web site

Web location: File System s\Visual Studio 2010\WebSites\ASPNETUnleashed\Chapter1\FirstWebsite

FIGURE 1.1 Creating a new website.



Add New Item - d:\Users\Nate\Documents\Visual Studio 2010\WebSites\ASPNETUnleashed\Chapter1\FirstWebsite\
Installed Templates
Visual Basic
Visual C#
Online Templates
Search Installed Templates
Web Form Visual C#
Master Page Visual C#
Web User Control Visual C#
ADO.NET Entity Data Model Visual C#
ADO.NET EntityObject Generator Visual C#
ADO.NET Self-Tracking Entity Generator Visual C#
AJAX-enabled WCF Service Visual C#
Browser File Visual C#
Class Visual C#
Class Diagram Visual C#
Crystal Report Visual C#
DataSet Visual C#
Add
Cancel

Sort by: Default

Type: Visual C#

A form for Web Applications

Name: FirstPage.aspx

☐ Place code in separate file

☐ Select master page

FIGURE 1.2 Adding a new ASP.NET page.

Make sure that your code for `FirstPage.aspx` looks like the code contained in Listing 1.1.

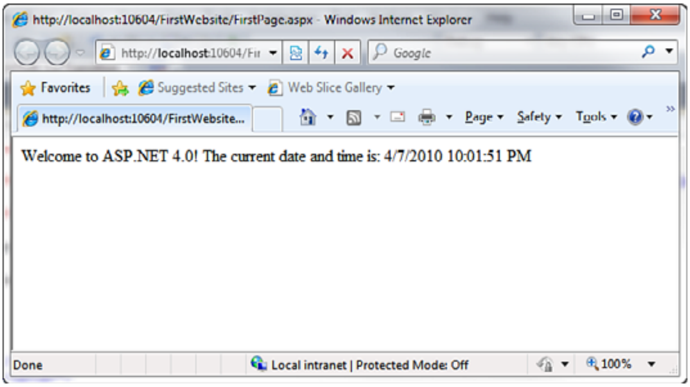
LISTING 1.1 FirstPage.aspx

```
<%@ Page Language="C#" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<script runat="server">

    void Page_Load()
    {
        lblServerTime.Text = DateTime.Now.ToString();
    }

</script>
<html xmlns="http://www.w3.org/1999/xhtml" >
<head>
    <title>First Page</title>
</head>
<body>
    <form id="form1" runat="server">
    <div>

    Welcome to ASP.NET 4.0! The current date and time is:

    <asp:Label
        id="lblServerTime"
        Runat="server" />

    </div>
    </form>
</body>
</html>
```

NOTE

The book's website contains all the code samples found in this book in both C# and VB.NET.

The ASP.NET page in Listing 1.1 displays a brief message and the server's current date and time. You can view the page in Listing 1.1 in a browser by right-clicking the page and selecting View in Browser (see Figure 1.3).



http://localhost:10604/FirstWebsite/FirstPage.aspx - Windows Internet Explorer
http://localhost:10604/Fir
Google
Favorites
Suggested Sites
Web Slice Gallery
http://localhost:10604/FirstWebsite...
Page
Safety
Tools
Done
Local intranet | Protected Mode: Off
100%

Welcome to ASP.NET 4.0! The current date and time is: 4/7/2010 10:01:51 PM

FIGURE 1.3 Viewing `FirstPage.aspx` in a browser.

The page in Listing 1.1 is an extremely simple page. However, it does illustrate the most common elements of an ASP.NET page. The page contains a directive, a code declaration block, and a page render block.

The first line, in Listing 1.1, contains a directive that looks like this:

```
<%@ Page Language="C#" %>
```

A directive always begins with the special characters `<%@` and ends with the characters `%>`. Directives are used primarily to provide the compiler with the information it needs to compile the page.

For example, the directive in Listing 1.1 indicates that the code contained in the page is C# code. The page is compiled by the C# compiler and not another compiler, such as the Visual Basic .NET (VB.NET) compiler.

The next part of the page begins with the opening `<script runat="server">` tag and ends with the closing `</script>` tag. The `<script>` tag contains the *code declaration block*.

The code declaration block contains all the methods used in the page. It contains all the page's functions and subroutines. The code declaration block in Listing 1.1 includes a single method named `Page_Load()`, which looks like this:

```
void Page_Load()
{
    lblServerTime.Text = DateTime.Now.ToString();
}
```

This method assigns the current date and time to the `Text` property of a `Label` control contained in the body of the page named `lblServerTime`.

The `Page_Load()` method is an example of an event handler. This method handles the `Page Load` event. Every time the page loads, the method automatically executes and assigns the current date and time to the `Label` control.

The final part of the page is called the *page render block*, which contains everything rendered to the browser. In Listing 1.1, the render block includes everything between the opening and closing `<html>` tags.

The majority of the page render block consists of everyday HTML. For example, the page contains the standard HTML `<head>` and `<body>` tags. In Listing 1.1, two special things are contained in the page render block.

First, notice that the page contains a `<form>` tag that looks like this:

```
<form id="form1" runat="server">
```

This is an example of an ASP.NET control. Because the tag includes a `runat="server"` attribute, the tag represents an ASP.NET control that executes on the server.

ASP.NET pages are often called *web form* pages because they almost always contain a server-side form element.

The page render block also contains a Label control. The Label control is declared with the `<asp:Label>` tag. In Listing 1.1, the Label control is used to display the current date and time.

Controls are the heart of ASP.NET Framework. Most of the ink contained in this book is devoted to describing the properties and features of ASP.NET controls. Controls are discussed in more detail shortly; however, first you need to understand .NET Framework.

NOTE

By default, ASP.NET pages are compatible with the XHTML 1.0 Transitional standard. The page in Listing 1.1 includes an XHTML 1.0 Transitional DOCTYPE. For details on how ASP.NET Framework complies with both XHTML and accessibility standards, see the article at the Microsoft MSDN website (msdn.Microsoft.com), “Building ASP.NET 2.0 Web Sites Using Web Standards.”

ASP.NET and the .NET Framework

ASP.NET is part of the Microsoft .NET Framework. To build ASP.NET pages, you need to take advantage of the features of .NET Framework, which consists of two parts: the Framework Class Library and the Common Language Runtime.

Understanding the Framework Class Library

The .NET Framework contains thousands of classes that you can use when building an application. Framework Class Library was designed to make it easier to perform the most common programming tasks. Following are just a few examples of the classes in the framework:

- **`File` class**—Enables you to represent a file on your hard drive. You can use the `File` class to check whether a file exists, create a new file, delete a file, and perform many other file-related tasks.
- **`Graphics` class**—Enables you to work with different types of images such as GIF, PNG, BMP, and JPEG. You can use the `Graphics` class to draw rectangles, arcs, ellipses, and other elements on an image
- **`Random` class**—Enables you to generate a random number.
- **`SmtpClient` class**—Enables you to send email. You can use the `SmtpClient` class to send emails that contain attachments and HTML content.

Framework has only four examples of classes. The .NET Framework contains more than 13,000 classes you can use when building applications.

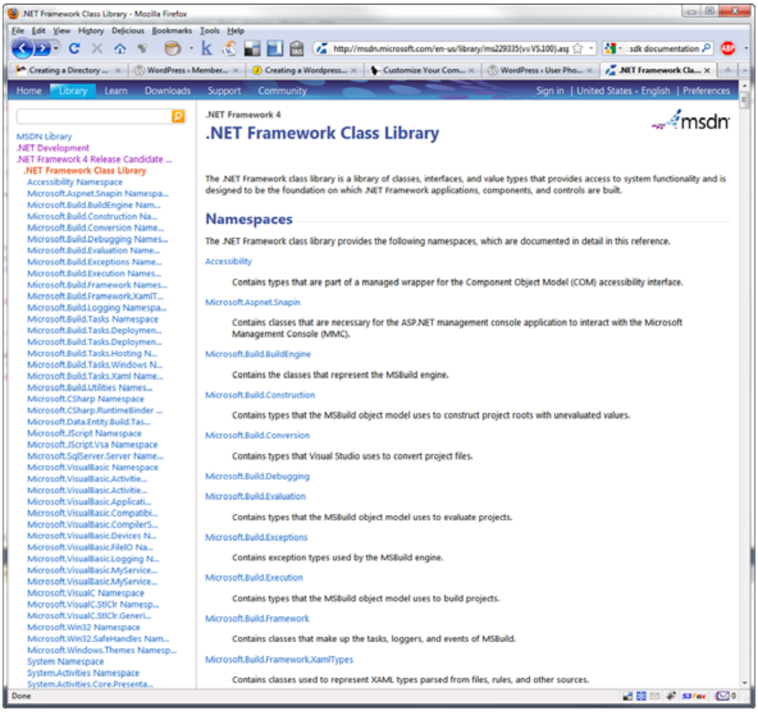
You can view all the classes contained in the framework by opening the Microsoft .NET Framework SDK documentation on Microsoft's .NET Framework Developer Center website and expanding the Class Library node (see Figure 1.4). The SDK documentation website is located at http://msdn.microsoft.com/en-us/netframework/default.aspx.

Each class in the Framework can include properties, methods, and events. The properties, methods, and events exposed by a class are the members of a class. For example, following is a partial list of the members of the `SmtpClient` class:

- Properties
  - **`Host`**—The name or IP address of your email server
  - **`Port`**—The number of the port to use when sending an email message
- Methods
  - **`Send`**—Enables you to send an email message synchronously
  - **`SendAsync`**—Enables you to send an email message asynchronously
- Events
  - **`SendCompleted`**—Raised when an asynchronous send operation completes

If you know the members of a class, you know everything that you can do with a class. For example, the `SmtpClient` class includes two properties named `Host` and `Port`, which enable you to specify the email server and port to use when sending an email message.

The `SmtpClient` class also includes two methods you can use to send an email: `Send()` and `SendAsync()`. The `Send` method blocks further program execution until the send operation is completed. The `SendAsync()` method, on the other hand, sends the email asynchronously. Unlike the `Send()` method, the `SendAsync()` method does not wait to check whether the send operation was successful.


.NET Framework Class Library - Mozilla Firefox
File Edit View History Delicious Bookmarks Tools Help
http://msdn.microsoft.com/en-us/library/ms229335(v=VS.100).asp
sdk documentation
Creating a Directory ...
WordPress › Member...
Creating a Wordpress...
Customize Your Com...
WordPress › User Pho...
.NET Framework Cla...
Home
Library
Learn
Downloads
Support
Community
Sign in | United States - English | Preferences
MSDN Library
.NET Development
.NET Framework 4 Release Candidate ...
.NET Framework Class Library
Accessibility Namespace
Microsoft.Aspnet.Snapin Namespa...
Microsoft.Build.BuildEngine Nam...
Microsoft.Build.Construction Na...
Microsoft.Build.Conversion Name...
Microsoft.Build.Debugging Names...
Microsoft.Build.Evaluation Name...
Microsoft.Build.Exceptions Name...
Microsoft.Build.Execution Names...
Microsoft.Build.Framework Names...
Microsoft.Build.Framework.XamlT...
Microsoft.Build.Logging Namespa...
Microsoft.Build.Tasks Namespace
Microsoft.Build.Tasks.Deploymen...
Microsoft.Build.Tasks.Deploymen...
Microsoft.Build.Tasks.Hosting N...
Microsoft.Build.Tasks.Windows N...
Microsoft.Build.Tasks.Xaml Name...
Microsoft.Build.Utilities Names...
Microsoft.CSharp Namespace
Microsoft.CSharp.RuntimeBinder ...
Microsoft.Data.Entity.Build.Tas...
Microsoft.JScript Namespace
Microsoft.JScript.Vsa Namespace
Microsoft.SqlServer.Server Name...
Microsoft.VisualBasic Namespace
Microsoft.VisualBasic.Activitie...
Microsoft.VisualBasic.Activitie...
Microsoft.VisualBasic.Applicati...
Microsoft.VisualBasic.Compatibi...
Microsoft.VisualBasic.CompilerS...
Microsoft.VisualBasic.Devices N...
Microsoft.VisualBasic.FileIO Na...
Microsoft.VisualBasic.Logging N...
Microsoft.VisualBasic.MyService...
Microsoft.VisualBasic.MyService...
Microsoft.VisualC Namespace
Microsoft.VisualC.StlClr Namesp...
Microsoft.VisualC.StlClr.Generi...
Microsoft.Win32 Namespace
Microsoft.Win32.SafeHandles Nam...
Microsoft.Windows.Themes Namesp...
System Namespace
System.Activities Namespace
System.Activities.Core.Presenta...
msdn
Done

.NET Framework 4

.NET Framework Class Library

The .NET Framework class library is a library of classes, interfaces, and value types that provides access to system functionality and is designed to be the foundation on which .NET Framework applications, components, and controls are built.

Namespaces

The .NET Framework class library provides the following namespaces, which are documented in detail in this reference.

Accessibility

Contains types that are part of a managed wrapper for the Component Object Model (COM) accessibility interface.

Microsoft.Aspnet.Snapin

Contains classes that are necessary for the ASP.NET management console application to interact with the Microsoft Management Console (MMC).

Microsoft.Build.BuildEngine

Contains the classes that represent the MSBuild engine.

Microsoft.Build.Construction

Contains types that the MSBuild object model uses to construct project roots with unevaluated values.

Microsoft.Build.Conversion

Contains types that Visual Studio uses to convert project files.

Microsoft.Build.Debugging

Microsoft.Build.Evaluation

Contains types that the MSBuild object model uses to evaluate projects.

Microsoft.Build.Exceptions

Contains exception types used by the MSBuild engine.

Microsoft.Build.Execution

Contains types that the MSBuild object model uses to build projects.

Microsoft.Build.Framework

Contains classes that make up the tasks, loggers, and events of MSBuild.

Microsoft.Build.Framework.XamlTypes

Contains classes used to represent XAML types parsed from files, rules, and other sources.

FIGURE 1.4 The online Microsoft .NET Framework SDK documentation.

Finally, the SmtpClient class includes an event named SendCompleted, which is raised when an asynchronous send operation completes. You can create an event handler for the SendCompleted event that displays a message when the email has been successfully sent.

The page in Listing 1.2 sends an email by using the SmtpClient class and calling its Send() method.

LISTING 1.2 SendMail.aspx

```
<%@ Page Language="C#" %>
<%@ Import Namespace="System.Net.Mail" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<script runat="server">

    void Page_Load()
    {
        SmtpClient client = new SmtpClient();
        client.Host = "localhost";
        client.Port = 25;
        client.Send("nate@somewhere", "nate@exclaimcomputing.com",
           "Beware!", "Watch out for zombies!");
    }

</script>
<html xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1" runat="server">
    <title>Send Mail</title>
</head>
<body>
    <form id="form1" runat="server">
    <div>
    Email sent!

    </div>
    </form>
</body>
</html>
```

The page in Listing 1.2 calls the `SmtpClient Send()` method to send the email. The first parameter is the from: address; the second parameter is the to: address; the third parameter is the subject; and the final parameter is the body of the email.

WARNING

The page in Listing 1.2 sends the email by using the local SMTP Server. If your SMTP Server is not enabled, you receive the error An Existing Connection Was Forcibly Closed by the Remote Host. You can enable your local SMTP Server by opening Internet Information Services, right-clicking Default SMTP Virtual Server, and selecting Start.

Understanding Namespaces

There are more than 13,000 classes in .NET Framework. This is an overwhelming number. If Microsoft simply jumbled all the classes together, you would never find anything. Fortunately, Microsoft divided the classes in the framework into separate namespaces.

A *namespace* is simply a category. For example, all the classes related to working with the file system are located in the `System.IO` namespace. All the classes for working a Microsoft SQL Server database are located in the `System.Data.SqlClient` namespace.

Before you can use a class in a page, you must indicate the namespace associated with the class. There are multiple ways of doing this.

First, you can fully qualify a class name with its namespace. For example, because the `File` class is contained in the `System.IO` namespace, you can use the following statement to check whether a file exists:

```
System.IO.File.Exists("SomeFile.txt")
```

Specifying a namespace every time you use a class can quickly become tedious. (It involves a lot of typing.) A second option is to import a namespace.

You can add an <$I<%@ Import % directive><%@ Import %> directive to a page to import a particular namespace. In Listing 1.2, we imported the System.Net.Mail namespace because the SmtpClient is part of this namespace. The page in Listing 1.2 includes the following directive near the top of the page:

```
<%@ Import Namespace="System.Net.Mail" %>
```

After you import a particular namespace, you can use all the classes in that namespace without qualifying the class names.

Finally, if you discover that you use a namespace in multiple pages in your application, you can configure all the pages in your application to recognize the namespace.

NOTE

A web configuration file is a special type of file that you can add to your application to configure your application. Be aware that the file is an XML file and, therefore, all the tags contained in the file are case-sensitive. You can add a web configuration file to your application by selecting Web Site, Add New Item Web Configuration File. Chapter 34, "Configuring Applications," discusses web configuration files in detail.

If you add the web configuration file in Listing 1.3 to your application, you do not need to import the System.Net.Mail namespace in a page to use the classes from this namespace. For example, if you include the Web.config file in your project, you can remove the <%@ Import %> directive from the page in Listing 1.2.

LISTING 1.3 Web.Config

```
<?xml version="1.0"?>
<configuration>
    <system.web>
      <pages>
        <namespaces>
          <add namespace="System.Net.Mail"/>
        </namespaces>
      </pages>
    </system.web>
</configuration>
```

You don't have to import every namespace. ASP.NET gives you the most commonly used namespaces for free:

- System
- System.Collections
- System.Collections.Generic
- System.Collections.Specialized
- System.ComponentModel.DataAnnotations
- System.Configuration
- System.Data.Entity.Linq
- System.Data.Linq
- System.Text
- System.Text.RegularExpressions
- System.Web
- System.Web.Caching
- System.Web.DynamicData
- System.Web.SessionState
- System.Web.Security
- System.Web.Profile
- System.Web.UI
- System.Web.UI.WebControls
- System.Web.UI.WebControls.WebParts
- System.Web.UI.HtmlControls
- System.Xml.Linq

The default namespaces are listed inside the `pages` element in the root web configuration file located at the following path:

```
\Windows\Microsoft.NET\Framework\v4.0.30128\Config\Web.Config
```

Understanding Assemblies

An assembly is the actual `.dll` file on your hard drive in which the classes in.NET Framework are stored. For example, all the classes contained in the ASP.NET Framework are located in an assembly named `System.Web.dll`.

More accurately, an assembly is the primary unit of deployment, security, and version control in .NET Framework. Because an assembly can span multiple files, an assembly is often referred to as a “logical” dll.

The two types of assemblies are private and shared. A private assembly can be used by only a single application. A shared assembly, on the other hand, can be used by all applications located on the same server.

Shared assemblies are located in the Global Assembly Cache (GAC). For example, the `System.Web.dll` assembly and all the other assemblies included with.NET Framework are located in the Global Assembly Cache.

NOTE

The Global Assembly Cache is located physically in your computer’s \WINDOWS\Assembly folder.

Before you can use a class contained in an assembly in your application, you must add a reference to the assembly. By default, an ASP.NET 4 application references the most common assemblies contained in the Global Assembly Cache:

- `mscorlib.dll`
- `Microsoft.CSharp`
- `System.dll`
- `System.Configuration.dll`
- `System.Web.dll`
- `System.Data.dll`
- `System.Web.Services.dll`
- `System.Xml.dll`
- `System.Drawing.dll`
- `System.EnterpriseServices.dll`
- `System.IdentityModel.dll`
- `System.Runtime.Serialization.dll`
- `System.ServiceModel.dll`
- `System.ServiceModel.Activation.dll`
- `System.ServiceModel.Web.dll`
- `System.Activities.dll`
- `System.ServiceModel.Activities.dll`

- `System.WorkflowServices.dll`
- `System.Core.dll`
- `System.Web.Extensions.dll`
- `System.Data.DataSetExtensions.dll`
- `System.Xml.Linq.dll`
- `System.ComponentModel.DataAnnotations.dll`
- `System.Web.DynamicData.dll`
- `System.Data.Entity.dll`
- `System.Web.Entity.dll`
- `System.Data.Linq.dll`
- `System.Data.Entity.Design.dll`
- `System.Web.ApplicationServices.dll`

All these assemblies are part of .NET 4 Framework. Websites created on previous versions of .NET referenced a different set of assemblies.

NOTE

You can target a website to work with .NET Framework 2.0, .NET Framework 3.0, .NET Framework 3.5, or .NET Framework 4. Within Visual Web Developer, select Website, Start Options and the Build tab. You can select the framework to target from a drop-down list.

To use any particular class in .NET Framework, you must do two things. First, your application must reference the assembly that contains the class. Second, your application must import the namespace associated with the class.

In most cases, you won't worry about referencing the necessary assembly because the most common assemblies are referenced automatically. However, if you need to use a specialized assembly, you need to add a reference explicitly to the assembly. For example, if you need to interact with Active Directory by using the classes in the System.DirectoryServices namespace, you need to add a reference to the System.DirectoryServices.dll assembly to your application.

Each class entry in .NET Framework SDK documentation lists the assembly and namespace associated with the class. For example, if you look up the `MessageQueue` class in the documentation, you discover that this class is located in the `System.Messaging` namespace located in the `System.Messaging.dll` assembly.

If you use Visual Web Developer, you can add a reference to an assembly explicitly by selecting Web Site, Add Reference, and the name of the assembly that you need to reference. For example, adding a reference to the `System.Messaging.dll` assembly results in the web configuration file in Listing 1.4 being added to your application.

LISTING 1.4 `Web.Config`

```
<?xml version="1.0"?>
<configuration>
<system.web>
  <compilation>
  <assemblies>
  <add
    assembly="System.Messaging, Version=4.0.0.0,
    Culture=neutral, PublicKeyToken=B03F5F7F11D50A3A"/>
  </assemblies>
  </compilation>
</system.web>
</configuration>
```

If you prefer not to use Visual Web Developer, you can add the reference to the `System.Messaging.dll` assembly by creating the file in Listing 1.4 manually.

Understanding the Common Language Runtime

The second part of .NET Framework is the Common Language Runtime (CLR). The CLR is responsible for executing your application code.

When you write an application for .NET Framework with a language such as C# or Visual Basic .NET, your source code is never compiled directly into machine code. Instead, the C# or Visual Basic compiler converts your code into a special language named Microsoft Intermediate Language (MSIL).

MSIL looks like an object-oriented assembly language; however, unlike a typical assembly language, it is not CPU-specific. MSIL is a low-level and platform-independent language.

When your application actually executes, the MSIL code is just-in-time compiled into machine code by the JITTER (the Just-In-Time compiler). Normally, your entire application is not compiled from MSIL into machine code. Instead, only the methods actually called during execution are compiled.

In reality, .NET Framework understands only one language: MSIL. However, you can write applications using languages such as Visual Basic .NET and C# for .NET Framework because .NET Framework includes compilers for these languages that enable you to compile your code into MSIL.

You can write code for .NET Framework using any one of dozens of different languages, including

- Ada
- Apl
- Caml
- COBOL
- Eiffel
- Forth
- Fortran
- JavaScript
- Oberon
- PERL
- Pascal
- PHP
- Python
- RPG
- Ruby
- Scheme
- Small Talk

The vast majority of developers building ASP.NET applications write the applications in either C# or Visual Basic .NET. Many of the other .NET languages in the preceding list are academic experiments. Once upon a time, if you wanted to become a developer, you concentrated on becoming proficient at a particular language. For example, you became a C++ programmer, a COBOL programmer, or a Visual Basic Programmer.

For .NET Framework, however, knowing a particular language is not particularly important. The choice of which language to use when building a .NET application is largely a preference choice. If you like case-sensitivity and curly braces, you should use the C# programming language. If you want to be lazy about casing and you don't like semicolons, write your code with Visual Basic .NET.

All the real action in .NET Framework happens in the Framework Class Library. If you want to become a good programmer using Microsoft technologies, you need to learn how to use the methods, properties, and events of the 13,000 classes included in the Framework. From the point of view of .NET Framework, it doesn't matter whether you use these classes from a Visual Basic .NET or C# application.

NOTE

All the code samples in this book were written in both C# and Visual Basic. All of the code samples can be found on the book's website.

Understanding ASP.NET Controls

ASP.NET controls are the heart of ASP.NET Framework. An ASP.NET control is a .NET class that executes on the server and renders certain content to the browser. For example, in the first ASP.NET page created at the beginning of this chapter, a Label control was used to display the current date and time. The ASP.NET framework includes more than 90 controls, which enable you to do everything from displaying a list of database records to displaying a randomly rotating banner advertisement.

This section provides an overview of the controls included in ASP.NET Framework. You also learn how to handle events raised by controls and how to take advantage of View State.

Overview of ASP.NET Controls

The ASP.NET Framework contains more than 90 controls. These controls can be divided into seven groups:

- **Standard Controls**—Enable you to render standard form elements such as buttons, input fields, and labels. We examine these controls in detail in Chapter 2, "Using the Standard Controls."
- **Validation Controls**—Enable you to validate form data before you submit the data to the server. For example, you can use a `RequiredFieldValidator` control to check whether a user entered a value for a required input field. These controls are discussed in Chapter 3, "Using the Validation Controls."
- **Rich Controls**—Enable you to render things such as calendars, file upload buttons, rotating banner advertisements, and multistep wizards. Chapter 4, "Using the Rich Controls," discusses these controls.
- **Data Controls**—Enable you to work with data such as database data. For example, you can use these controls to submit new records to a database table or display a list of database records. Part III, "Performing Data Access," discusses these controls.
- **Navigation Controls**—Enable you to display standard navigation elements such as menus, tree views, and bread crumb trails. Chapter 22, "Using the Navigation Controls," discusses these controls.

- **Login Controls—**Enables you to display login, change password, and registration forms. Chapter 26, "Using the Login Controls," discusses these controls.
- **HTML Controls—**Enable you to convert any HTML tag into a server-side control. This group of controls are discussed in the next section.

With the exception of the HTML controls, you declare and use all ASP.NET controls in a page in exactly the same way. For example, if you want to display a text input field in a page, you can declare a `TextBox` control like this:

```
<asp:TextBox id="TextBox1" runat="Server" />
```

This control declaration looks like the declaration for an HTML tag. Remember, however, unlike an HTML tag, a control is a .NET class that executes on the server and not in the web browser.

When the TextBox control is rendered to the browser, it renders the following content:

```
<input name="TextBox1" type="text" id="TextBox1" />
```

The first part of the control declaration, the `asp:` prefix, indicates the namespace for the control. All the standard ASP.NET controls are contained in the `System.Web.UI.WebControls` namespace. The prefix `asp:` represents this namespace.

Next, the declaration contains the name of the control being declared. In this case, a `TextBox` control is declared.

This declaration also includes an ID attribute. You use the ID to refer to the control in the page within your code. Every control must have a unique ID.

NOTE

You should always assign an ID attribute to every control even when you don't need to program against it. If you don't provide an ID attribute, certain features of ASP.NET Framework (such as two-way databinding) won't work.

The declaration also includes a `runat="Server"` attribute. This attribute marks the tag as representing a server-side control. If you neglect to include this attribute, the `TextBox` tag would be passed, without being executed, to the browser. The browser would simply ignore the tag.

Finally, notice that the tag ends with a forward slash. The forward slash is shorthand for creating a closing `</asp:TextBox>` tag. You can, if you prefer, declare the `TextBox` control like this:

```
<asp:TextBox id="TextBox1" runat="server"></asp:TextBox>
```

In this case, the opening tag does not contain a forward slash and an explicit closing tag is included.

Understanding HTML Controls

You declare HTML controls in a different way than you declare standard ASP.NET controls. The ASP.NET Framework enables you to take any HTML tag (real or imaginary) and add a runat="server" attribute to the tag. The runat="server" attribute converts the HTML tag into a server-side ASP.NET control.

For example, the page in Listing 1.5 contains a <span> tag, which has been converted into an ASP.NET control.

LISTING 1.5 HtmlControls.aspx

```
<%@ Page Language="C#" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<script runat="server">

    void Page_Load()
    {
        spanNow.InnerText = DateTime.Now.ToString("T");
    }

</script>
<html xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1" runat="server">
    <title>HTML Controls</title>
</head>
<body>
    <form id="form1" runat="server">
    <div>

    At the tone, the time will be:
    <span id="spanNow" runat="server" />

    </div>
    </form>
</body>
</html>
```

The <span> tag in Listing 1.5 looks just like a normal HTML <span> tag except for the addition of the runat="server" attribute.

Because the <span> tag in Listing 1.5 is a server-side HTML control, you can program against it. In Listing 1.5, the current date and time are assigned to the <span> tag in the Page_Load() method.

The HTML controls are included in ASP.NET Framework to make it easier to convert existing HTML pages to use ASP.NET Framework. I rarely use the HTML controls in this book because, in general, the standard ASP.NET controls provide all the same functionality and more.

Understanding and Handling Control Events

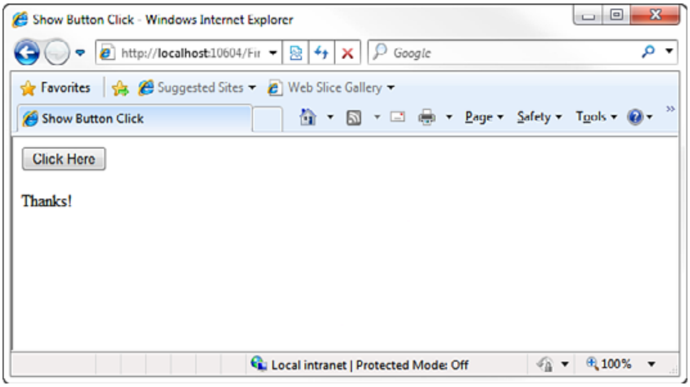
The majority of ASP.NET controls support one or more events. For example, the ASP.NET `Button` control supports the `Click` event. The `Click` event is raised on the server after you click the button rendered by the `Button` control in the browser.

The page in Listing 1.6 illustrates how you can write code that executes when a user clicks the button rendered by the `Button` control (in other words, it illustrates how you can create a `Click` event handler):

LISTING 1.6 `ShowButtonClick.aspx`

```
<%@ Page Language="C#" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<script runat="server">

    protected void btnSubmit_Click(object sender, EventArgs e)
    {
        Label1.Text = "Thanks!";
    }
</script>
<html xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1" runat="server">
    <title>Show Button Click</title>
</head>
<body>
    <form id="form1" runat="server">
    <div>

    <asp:Button
        id="btnSubmit"
        Text="Click Here"
        OnClick="btnSubmit_Click"
        Runat="server" />

    <br /><br />
    <asp:Label
        id="Label1"
        Runat="server" />

    </div>
    </form>
</body>
</html>
```

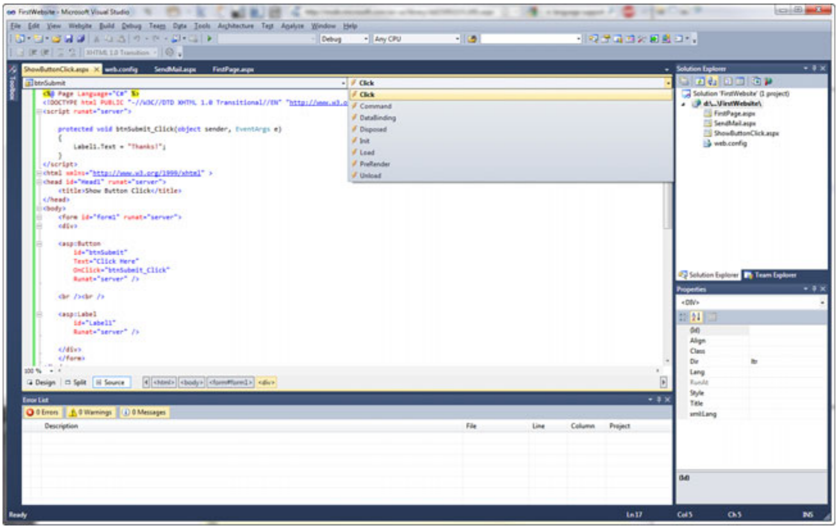
Notice that the `Button` control in Listing 1.6 includes an `OnClick` attribute. This attribute points to a subroutine named `btnSubmit_Click()`. The `btnSubmit_Click()` subroutine is the handler for the `Button Click` event. This subroutine executes whenever you click the button (see Figure 1.5).


Show Button Click - Windows Internet Explorer
http://localhost:10604/Fir
Google
Favorites
Suggested Sites
Web Slice Gallery
Show Button Click
Page
Safety
Tools
Click Here
Local intranet | Protected Mode: Off
100%

Thanks!

FIGURE 1.5 Raising a `Click` event.

You can add an event handler automatically to a control in multiple ways when using Visual Web Developer. In Source view, add a handler by selecting a control from the top-left drop-down list and selecting an event from the top-right drop-down list. The event handler code is added to the page automatically (see Figure 1.6).

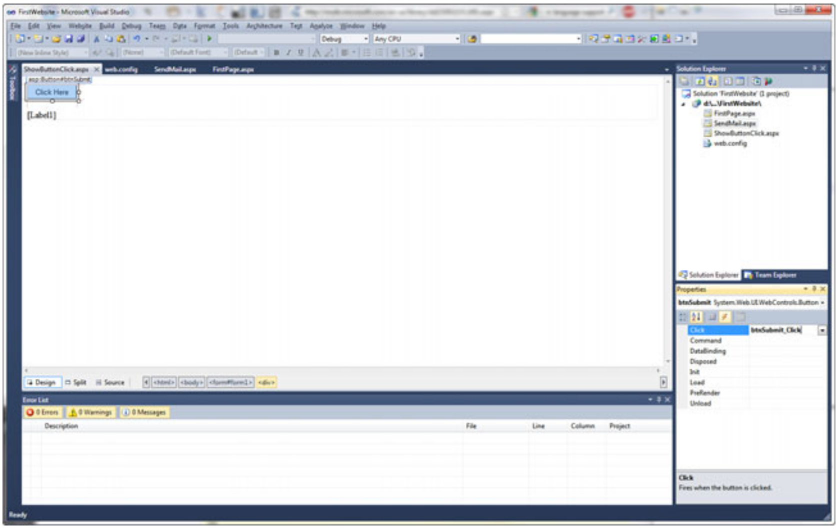


```
<%@ Page Language="C#" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.o
<script runat="server">

    protected void btnSubmit_Click(object sender, EventArgs e)
    {
        Label1.Text = "Thanks!";
    }
</script>
<html xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1" runat="server">
    <title>Show Button Click</title>
</head>
<body>
    <form id="form1" runat="server">
    <div>

    <asp:Button
        id="btnSubmit"
        Text="Click Here"
        OnClick="btnSubmit_Click"
        Runat="server" />

    <br /><br />

    <asp:Label
        id="Label1"
        Runat="server" />

    </div>
    </form>
```

FIGURE 1.6 Adding an event handler from Source view.

In Design view, you can double-click a control to add a handler for the control's default event. Double-clicking a control switches you to Source view and adds the event handler.

Finally, from Design view, after selecting a control on the designer surface, you can add an event handler from the Properties window by clicking the Events button (the lightning bolt) and double-clicking next to the name of any of the events (see Figure 1.7).



btnSubmit System.Web.UI.WebControls.Button

Click	btnSubmit_Click
Command	
DataBinding	
Disposed	
Init	
Load	
PreRender	
Unload	

Click
Fires when the button is clicked.

FIGURE 1.7 Adding an event handler from the Properties window.

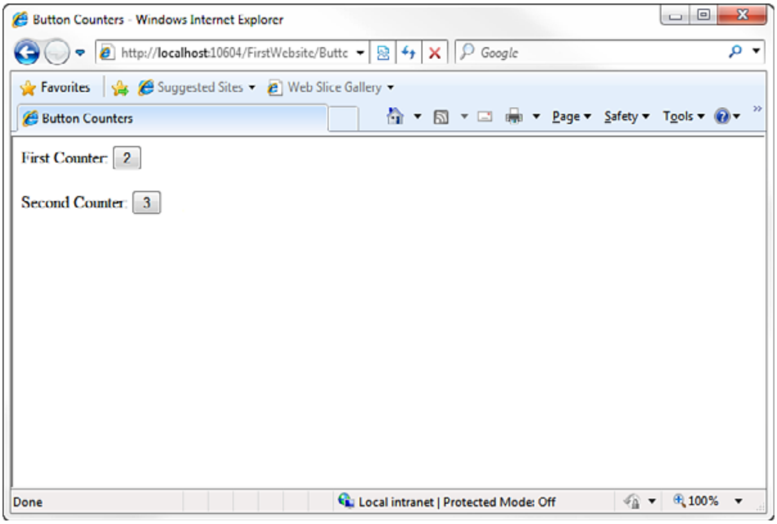
You need to understand that all ASP.NET control events happen on the server. For example, the `Click` event is not raised when you actually click a button. The `Click` event is not raised until the page containing the Button control is posted back to the server.

The ASP.NET Framework is a server-side web application framework. The .NET Framework code that you write executes on the server and not within the web browser. From the perspective of ASP.NET, nothing happens until the page is posted back to the server and can execute within the context of .NET Framework.

Notice that two parameters are passed to the `btnSubmit_Click()` handler in Listing 1.6. All event handlers for ASP.NET controls have the same general signature.

The first parameter, the object parameter named `sender`, represents the control that raised the event. In other words, it represents the `Button` control that you clicked.

You can wire multiple controls in a page to the same event handler and use this first parameter to determine the particular control that raised the event. For example, the page in Listing 1.7 includes two `Button` controls. When you click either `Button` control, the text displayed by the `Button` control is updated (see Figure 1.8).


Button Counters - Windows Internet Explorer
http://localhost:10604/FirstWebsite/Buttc
Google
Favorites
Suggested Sites
Web Slice Gallery
Button Counters
Page
Safety
Tools
Done
Local intranet | Protected Mode: Off
100%

First Counter: 2

Second Counter: 3

FIGURE 1.8 Handling two `Button` controls with one event handler.

LISTING 1.7 `ButtonCounters.aspx`

```
 <%@ Page Language="C#" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<script runat="server">

    protected void Button_Click(object sender, EventArgs e)
    {
        Button btn = (Button)sender;
        btn.Text = (Int32.Parse(btn.Text) + 1).ToString();
    }
</script>
<html xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1" runat="server">
    <title>Button Counters</title>
</head>
<body>
    <form id="form1" runat="server">
    <div>

    First Counter:
    <asp:Button
        id="Button1"
        Text="0"
        OnClick="Button_Click"
        Runat="server" />

    <br /><br />

    Second Counter:
    <asp:Button
        id="Button2"
        Text="0"
        OnClick="Button_Click"
        Runat="server" />

    </div>
    </form>
</body>
</html>
```

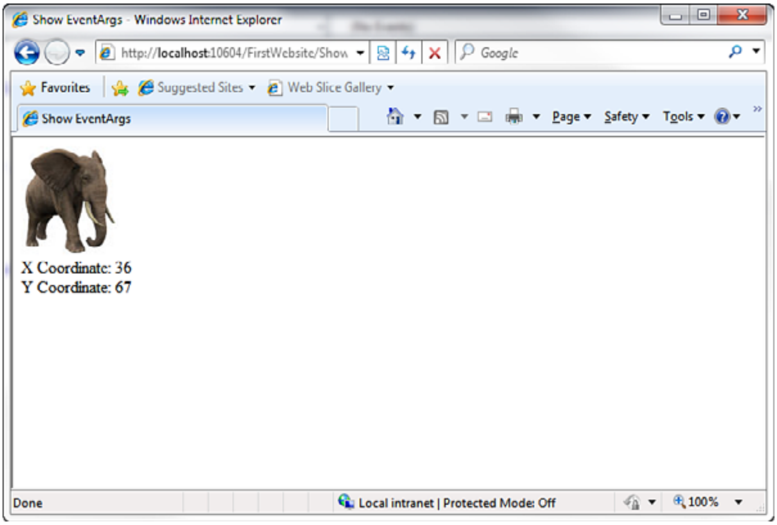
The second parameter passed to the `Click` event handler, the `EventArgs` parameter named `e`, represents any additional event information associated with the event. No additional event information is associated with clicking a button, so this second parameter does not represent anything useful in either Listing 1.6 or Listing 1.7.

When you click an `ImageButton` control instead of a `Button` control, on the other hand, additional event information is passed to the event handler. When you click an `ImageButton` control, the X and Y coordinates of where you clicked are passed to the handler.

The page in Listing 1.8 contains an ImageButton control that displays a picture. When you click the picture, the X and Y coordinates of the spot you clicked display in a Label control (see Figure 1.9).

LISTING 1.8 ShowEventArgs.aspx

```
<%@ Page Language="C#" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<script runat="server">

    protected void btnElephant_Click(object sender, ImageClickEventArgs e)
    {
        lblX.Text = e.X.ToString();
        lblY.Text = e.Y.ToString();
    }
</script>
<html xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1" runat="server">
    <title>Show EventArgs</title>
</head>
<body>
    <form id="form1" runat="server">
    <div>

    <asp:ImageButton
        id="btnElephant"
        ImageUrl="Elephant.jpg"
        Runat="server" OnClick="btnElephant_Click" />

    <br />
    X Coordinate:
    <asp:Label
        id="lblX"
        Runat="server" />
    <br />
    Y Coordinate:
    <asp:Label
        id="lblY"
        Runat="server" />

    </div>
    </form>
</body>
</html>
```


Show EventArgs - Windows Internet Explorer
http://localhost:10604/FirstWebsite/Show
Google
Favorites
Suggested Sites
Web Slice Gallery
Show EventArgs
Page
Safety
Tools
Done
Local intranet | Protected Mode: Off
100%

X Coordinate: 36
Y Coordinate: 67

FIGURE 1.9 Clicking an `ImageButton`.

The second parameter passed to the `btnElephant_Click()` method is an `ImageClickEventArgs` parameter. Whenever the second parameter is not the default `EventArgs` parameter, you know that additional event information is passed to the handler.

Understanding View State

The HTTP protocol, the fundamental protocol of the World Wide Web, is a stateless protocol. Each time you request a web page from a website, from the website's perspective, you are a completely new person.

The ASP.NET Framework, however, manages to transcend this limitation of the HTTP protocol. For example, if you assign a value to a Label control's `Text` property, the Label control retains this value across multiple page requests.

Consider the page in Listing 1.9. This page contains a Button control and a Label control. Each time you click the Button control, the value displayed by the Label control is incremented by 1 (see Figure 1.10). How does the Label control preserve its value across postbacks to the web server?

LISTING 1.9 ShowViewState.aspx

```
<%@ Page Language="C#" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<script runat="server">

    protected void btnAdd_Click(object sender, EventArgs e)
    {
        lblCounter.Text = (Int32.Parse(lblCounter.Text) + 1).ToString();
    }
</script>
<html xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1" runat="server">
    <title>Show  View State</title>
</head>
<body>
    <form id="form1" runat="server">
    <div>

    <asp:Button
        id="btnAdd"
        Text="Add"
        OnClick="btnAdd_Click"
        Runat="server" />

    <asp:Label
        id="lblCounter"
        Text="0"
        Runat="server" />

    </div>
    </form>
</body>
</html>
```

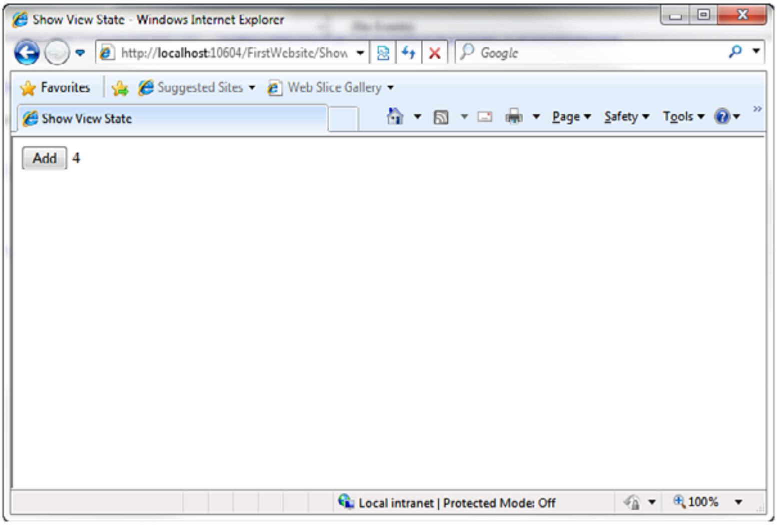

Show View State - Windows Internet Explorer
http://localhost:10604/FirstWebsite/Show
Google
Favorites
Suggested Sites
Web Slice Gallery
Show View State
Page
Safety
Tools
Add
4
Local intranet | Protected Mode: Off
100%

FIGURE 1.10 Preserving state between postbacks.

The ASP.NET Framework uses a trick called View State. If you open the page in Listing 1.9 in your browser and select View Source, you notice that the page includes a hidden form field named `__VIEWSTATE` that looks like this:

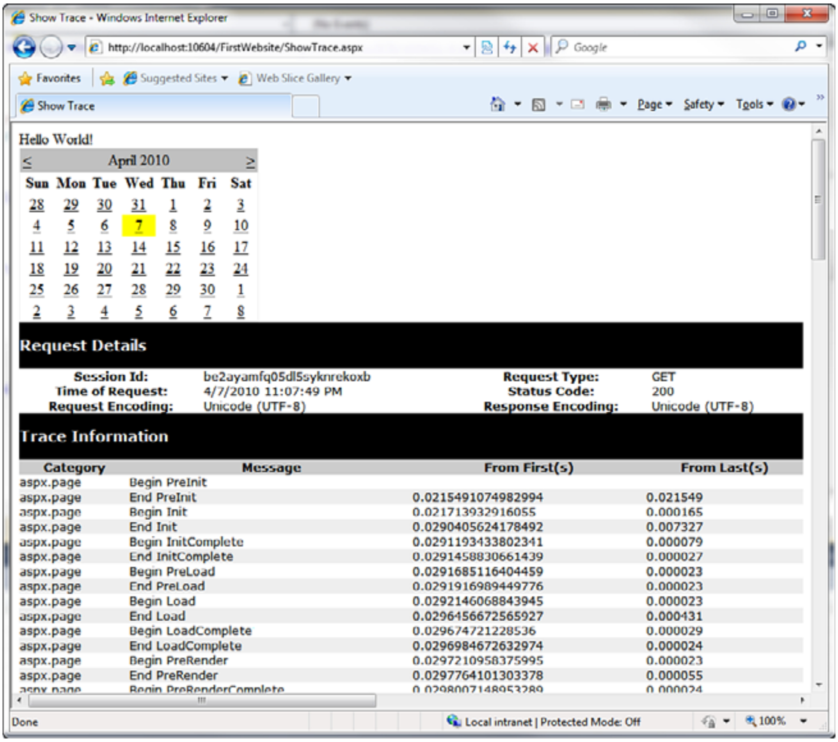
```
<input type="hidden" name="__VIEWSTATE" id="__
  VIEWSTATE" value="/wEPDwUKLTc2ODE1OTYxNw9kFgICBA9kFgIC
  Aw8PFgIeBFRleHQFATFkZGT3tMnThg9KZpGak55p367vfInj1w==" />
```

This hidden form field contains the value of the Label control's `Text` property (and the values of any other control properties stored in View State). When the page is posted back to the server, ASP.NET Framework rips apart this string and re-creates the values of all the properties stored in View State. In this way, ASP.NET Framework preserves the state of control properties across postbacks to the web server.

By default, View State is enabled for every control in ASP.NET Framework. If you change the background color of a Calendar control, the new background color is remembered across postbacks. If you change the selected item in a `DropDownList`, the selected item is remembered across postbacks. The values of these properties are automatically stored in View State.

View State is a good thing, but sometimes it can be too much of a good thing. The `__VIEWSTATE` hidden form field can become large. Stuffing too much data into View State can slow down the rendering of a page because the contents of the hidden field must be pushed back and forth between the web server and web browser.

You can determine how much View State each control contained in a page is consuming by enabling tracing for a page (see Figure 1.11). The page in Listing 1.10 includes a `Trace="true"` attribute in its `<%@ Page %>` directive, which enables tracing.


Show Trace - Windows Internet Explorer
http://localhost:10604/FirstWebsite/ShowTrace.aspx
Google
Favorites
Suggested Sites
Web Slice Gallery
Show Trace
Page
Safety
Tools
Done
Local intranet | Protected Mode: Off
100%

Hello World!

<			April 2010			>
Sun	Mon	Tue	Wed	Thu	Fri	Sat
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	1
2	3	4	5	6	7	8

Request Details

Session Id:	be2ayamfq05dl5syknrekoxb	**Request Type:**	GET
Time of Request:	4/7/2010 11:07:49 PM	**Status Code:**	200
Request Encoding:	Unicode (UTF-8)	**Response Encoding:**	Unicode (UTF-8)

Trace Information

Category	Message	From First(s)	From Last(s)
aspx.page	Begin PreInit		
aspx.page	End PreInit	0.0215491074982994	0.021549
aspx.page	Begin Init	0.021713932916055	0.000165
aspx.page	End Init	0.0290405624178492	0.007327
aspx.page	Begin InitComplete	0.0291193433802341	0.000079
aspx.page	End InitComplete	0.0291458830661439	0.000027
aspx.page	Begin PreLoad	0.0291685116404459	0.000023
aspx.page	End PreLoad	0.0291916989449776	0.000023
aspx.page	Begin Load	0.0292146068843945	0.000023
aspx.page	End Load	0.0296456672565927	0.000431
aspx.page	Begin LoadComplete	0.029674721228536	0.000029
aspx.page	End LoadComplete	0.0296984672632974	0.000024
aspx.page	Begin PreRender	0.0297210958375995	0.000023
aspx.page	End PreRender	0.0297764101303378	0.000055
aspx.page	Begin PreRenderComplete	0.0298007148953289	0.000024

FIGURE 1.11 Viewing View State size for each control.

LISTING 1.10 ShowTrace.aspx

```
<%@ Page Language="C#" Trace="true" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<script runat="server">

    void Page_Load()
    {
        Label1.Text = "Hello World!";
        Calendar1.TodaysDate = DateTime.Now;
    }

</script>
<html xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1" runat="server">
    <title>Show Trace</title>
</head>
<body>
```

```
    <form id="form1" runat="server">
    <div>

    <asp:Label
        id="Label1"
        Runat="server" />
    <asp:Calendar
        id="Calendar1"
        TodayDayStyle-BackColor="Yellow"
        Runat="server" />

    </div>
    </form>
</body>
</html>
```

When you open the page in Listing 1.10, additional information about the page is appended to the bottom of the page. The Control Tree section displays the amount of View State used by each ASP.NET control contained in the page.

Every ASP.NET control includes a property named `EnableViewState`. If you set this property to the value `False`, View State is disabled for the control. In that case, the values of the control properties are not remembered across postbacks to the server.

For example, the page in Listing 1.11 contains two Label controls and a Button control. The first Label has View State disabled, and the second Label has View State enabled. When you click the button, only the value of the second Label control is incremented past 1.

LISTING 1.11 `DisableViewState.aspx`

```
<%@ Page Language="C#" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<script runat="server">

    protected void btnAdd_Click(object sender, EventArgs e)
    {
        Label1.Text = (Int32.Parse(Label1.Text) + 1).ToString();
        Label2.Text = (Int32.Parse(Label2.Text) + 1).ToString();
    }
</script>
<html xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1" runat="server">
    <title>Disable  View State</title>
</head>
```

```
<body>
    <form id="form1" runat="server">
    <div>

    Label 1:
    <asp:Label
        id="Label1"
        EnableViewState="false"
        Text="0"
        Runat="server" />

    <br />

    Label 2:
    <asp:Label
        id="Label2"
        Text="0"
        Runat="server" />

    <br /><br />

    <asp:Button
        id="btnAdd"
        Text="Add"
        OnClick="btnAdd_Click"
        Runat="server" />

    </div>
    </form>
</body>
</html>
```

Sometimes, you might want to disable View State even when you aren't concerned with the size of the __VIEWSTATE hidden form field. For example, if you use a Label control to display a form validation error message, you might want to start from scratch each time the page is submitted. In that case, simply disable View State for the Label control.

NOTE

The ASP.NET Framework version 2.0 introduced a new feature called Control State, which is similar to View State except that it is used to preserve only critical state information. For example, the GridView control uses Control State to store the selected row. Even if you disable View State, the GridView control remembers which row is selected.

Understanding ASP.NET Pages

This section examines ASP.NET pages in more detail. You learn about dynamic compilation and code-behind files. We also discuss the events supported by the `Page` class.

Understanding Dynamic Compilation

Strangely enough, when you create an ASP.NET page, you are actually creating the source code for a .NET class. You are creating a new instance of the `System.Web.UI.Page` class. The entire contents of an ASP.NET page, including all script and HTML content, are compiled into a .NET class.

When you request an ASP.NET page, ASP.NET Framework checks for a .NET class that corresponds to the page. If a corresponding class does not exist, the Framework automatically compiles the page into a new class and stores the compiled class (the assembly) in the Temporary ASP.NET Files folder located at the following path:

```
\WINDOWS\Microsoft.NET\Framework\v4.0.30128\Temporary ASP.NET Files
```

The next time anyone requests the same page in the future, the page is not compiled again. The previously compiled class is executed, and the results are returned to the browser.

Even if you unplug your web server, move to Borneo for 3 years, and start up your web server again, the next time someone requests the same page, the page does not need to be recompiled. The compiled class is preserved in the Temporary ASP.NET Files folder until the source code for your application is modified.

When the class is added to the Temporary ASP.NET Files folder, a file dependency is created between the class and the original ASP.NET page. If the ASP.NET page is modified in any way, the corresponding .NET class is automatically deleted. The next time someone requests the page, the Framework automatically compiles the modified page source into a new .NET class.

This process is called *dynamic compilation*, which enables ASP.NET applications to support thousands of simultaneous users. Unlike an ASP Classic page, for example, an ASP.NET page does not need to be parsed and compiled every time it is requested. An ASP.NET page is compiled only when an application is modified.

NOTE

You can precompile an entire ASP.NET application by using the `aspnet_compiler.exe` command-line tool. If you precompile an application, users don't experience the compilation delay resulting from the first page request.

NOTE

You can disable dynamic compilation for a single page, the pages in a folder, or an entire website with the `CompilationMode` attribute. When the `CompilationMode` attribute is used with the `<%@ Page %>` directive, it enables you to disable dynamic compilation for a single page. When the `compilationMode` attribute is used with the `pages` element in a web configuration file, it enables you to disable dynamic compilation for an entire folder or application.

Disabling compilation is useful when you have thousands of pages in a website and you don't want to load too many assemblies into memory. When the `CompilationMode` attribute is set to the value `Never`, the page is never compiled, and an assembly is never generated for the page. The page is interpreted at runtime.

You cannot disable compilation for pages that include server-side code. In particular, a no compile page cannot include a server-side `<script>...</script>` block. On the other hand, a no compile page can contain ASP.NET controls and databinding expressions.

If you are curious, I've included the source code for the class that corresponds to the `FirstPage.aspx` page in Listing 1.12. (I cleaned up the code and made it shorter to save space.) I copied this file from the Temporary ASP.NET Files folder after enabling debugging for the application.

LISTING 1.12 `FirstPage.aspx` Source

```
namespace ASP
{
    using System.Web.Security;
    using System.Web;
    using System.Web.SessionState;
    using System.Text;
    using System.Collections.Specialized;
    using System.Web.Profile;
    using System.Net.Mail;
    using System.Collections;
    using System.Web.UI.WebControls.WebParts;
    using System.Configuration;
    using System;
    using System.Web.Caching;
    using System.Web.UI;
    using System.Text.RegularExpressions;
    using System.Web.UI.WebControls;
    using System.Web.UI.HtmlControls;
    [System.Runtime.CompilerServices.CompilerGlobalScopeAttribute()]
    public class firstpage_aspx : global::System.Web.UI.Page,
System.Web.SessionState.IRequiresSessionState, System.Web.IHttpHandler
    {
        protected global::System.Web.UI.WebControls.Label lblServerTime;
        protected global::System.Web.UI.HtmlControls.HtmlForm form1;
        private static bool @__initialized;
        private static object @__fileDependencies;

        void Page_Load()
        {
            lblServerTime.Text = DateTime.Now.ToString();
        }

        public firstpage_aspx()
        {
            string[] dependencies;
             ((global::System.Web.UI.Page)(this)).AppRelativeVirtualPath =
➥"~/FirstPage.aspx";
            if ((global::ASP.firstpage_aspx.@__initialized == false))
            {
                dependencies = new string[1];
                dependencies[0] = "~/FirstPage.aspx";
                global::ASP.firstpage_aspx.@__fileDependencies =
this.GetWrappedFileDependencies(dependencies);
                global::ASP.firstpage_aspx.@__initialized = true;
            }
            this.Server.ScriptTimeout = 30000000;
        }

        protected System.Web.Profile.DefaultProfile Profile
        {
            get
            {
                return ((System.Web.Profile.DefaultProfile)(this.Context.Profile));
            }
        }

        protected System.Web.HttpApplication ApplicationInstance
        {
            get
            {
                return ((System.Web.HttpApplication)(this.Context.Application
➥Instance));
            }
        }

        private global::System.Web.UI.WebControls.Label @__BuildControllbl
➥ServerTime()
        {
        ...code...
        }

        private global::System.Web.UI.HtmlControls.HtmlForm @__BuildControlform1()
        {
        ...code...
        }

        private void @__BuildControlTree(firstpage_aspx @__ctrl)
        {
        ...code...
        }

        protected override void FrameworkInitialize()
        {
            base.FrameworkInitialize();
            this.@__BuildControlTree(this);
            this.AddWrappedFileDependencies(global::ASP.firstpage_aspx.@__file
➥Dependencies);
            this.Request.ValidateInput();
        }

        public override int GetTypeHashCode()
        {
            return 243955639;
        }

        public override void ProcessRequest(System.Web.HttpContext context)
        {
            base.ProcessRequest(context);
        }
    }
}
```

The class in Listing 1.12 inherits from the `System.Web.UI.Page` class. The `ProcessRequest()` method is called by ASP.NET Framework when the page is displayed. This method builds the page's control tree, which is the subject of the next section.

Understanding Control Trees

In the previous section, you learned that an ASP.NET page is actually the source code for a .NET class. Alternatively, you can think of an ASP.NET page as a bag of controls. More accurately, because some controls might contain child controls, you can think of an ASP.NET page as a control tree.

For example, the page in Listing 1.13 contains a DropDownList control and a Button control. Furthermore, because the `<%@ Page %>` directive has the `Trace="true"` attribute, tracing is enabled for the page.

LISTING 1.13 `ShowControlTree.aspx`

```
<%@ Page Language="C#" Trace="true" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1" runat="server">
    <title>Show Control Tree</title>
</head>
<body>
    <form id="form1" runat="server">
    <div>

    <asp:DropDownList
        id="DropDownList1"
        Runat="server">
        <asp:ListItem Text="Oranges" />
        <asp:ListItem Text="Apples" />
    </asp:DropDownList>

    <asp:Button
        id="Button1"
        Text="Submit"
        Runat="server" />

    </div>
    </form>
</body>
</html>
```

When you open the page in Listing 1.13 in your browser, you can see the control tree for the page appended to the bottom of the page. It looks like this:

```
__Page ASP.showcontroltree_aspx
    ctl02 System.Web.UI.LiteralControl
```

```
ctl00 System.Web.UI.HtmlControls.HtmlHead
    ctl01 System.Web.UI.HtmlControls.HtmlTitle
ctl03 System.Web.UI.LiteralControl
form1 System.Web.UI.HtmlControls.HtmlForm
    ctl04 System.Web.UI.LiteralControl
    DropDownList1 System.Web.UI.WebControls.DropDownList
    ctl05 System.Web.UI.LiteralControl
    Button1 System.Web.UI.WebControls.Button
    ctl06 System.Web.UI.LiteralControl
ctl07
```

The root node in the control tree is the page itself. The page has an ID of `__Page`. The page class contains all the other controls in its child controls collection. The control tree also contains an instance of the `HtmlForm` class named `form1`. This control is the server-side form tag contained in the page. It contains all the other form controls—the DropDownList and Button controls—as child controls.

Several `LiteralControl` controls are interspersed between the other controls in the control tree. What are these controls?

Remember that everything in an ASP.NET page is converted into a .NET class, including any HTML or plain text content in a page. The `LiteralControl` class represents the HTML content in the page (including any carriage returns between tags).

NOTE

Normally, you refer to a control in a page by its ID. However, there are situations in which this is not possible. In those cases, you can use the `FindControl()` method of the Control class to retrieve a control with a particular ID. The `FindControl()` method is similar to the JavaScript `getElementById()` method.

Using Code-Behind Pages

The ASP.NET Framework (and Visual Web Developer) enables you to create two different types of ASP.NET pages. You can create both single-file and two-file ASP.NET pages.

All the code samples in this book are written as single-file ASP.NET pages. In a single-file ASP.NET page, a single file contains both the page code and page controls. The page code is contained in a `<script runat="server">` tag.

As an alternative to a single-file ASP.NET page, you can create a two-file ASP.NET page. A two-file ASP.NET page is normally referred to as a *code-behind* page. In a code-behind page, the page code is contained in a separate file.

NOTE

Code-behind pages work in a different way after ASP.NET 2.0 Framework than they did in ASP.NET 1.x Framework. In ASP.NET 1.x, the two halves of a code-behind page were related by inheritance. After ASP.NET 2.0 Framework, the two halves of a code-behind page are related by a combination of partial classes and inheritance.

For example, Listing 1.14 and Listing 1.15 contain the two halves of a code-behind page.

VISUAL WEB DEVELOPER NOTE

When using Visual Web Developer, you create a code-behind page by selecting Web Site, Add New Item and selecting the Web Form Item and checking the Place Code in Separate File check box before adding the page.

LISTING 1.14 `FirstPageCodeBehind.aspx`

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="FirstPageCodeBehind.aspx.cs
"Inherits="FirstPageCodeBehind" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1" runat="server">
    <title>First Page Code-Behind</title>
</head>
<body>
    <form id="form1" runat="server">
    <div>

    <asp:Button
        id="Button1"
        Text="Click Here"
        OnClick="Button1_Click"
        Runat="server" />

    <br /><br />

    <asp:Label
        id="Label1"
        Runat="server" />
    </div>
    </form>
</body>
</html>
```

LISTING 1.15 `FirstPageCodeBehind.aspx.cs`

```
using System;
using System.Collections.Generic;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class FirstPageCodeBehind : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        Label1.Text = "Click the Button";
    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        Label1.Text = "Thanks!";
    }
}
```

The page in Listing 1.14 is called the *presentation page*. It contains a Button control and a Label control. However, the page does not contain any code. All the code is contained in the code-behind file.

VISUAL WEB DEVELOPER NOTE

You can flip to the code-behind file for a page by right-clicking a page and selecting View Code.

The code-behind the file in Listing 1.15 contains the `Page_Load()` and `Button1_Click()` handlers. The code-behind file in Listing 1.15 does not contain any controls.

Notice that the page in Listing 1.14 includes both a `CodeFile` and `Inherits` attribute in its `<%@ Page %>` directive. These attributes link the page to its code-behind file.

How Code-Behind Works: The Ugly Details

In the early versions of ASP.NET Framework (ASP.NET 1.x), two classes were generated by a code-behind page. One class corresponded to the presentation page, and one class corresponded to the code-behind file. These classes were related to one another through class inheritance. The presentation page class inherited from the code-behind file class.

The problem with this method of associating presentation pages with their code-behind files was that it was brittle. Inheritance is a one-way relationship. Anything that is true of the mother is true of the daughter, but not the other way around. Any control that you declared in the presentation page was required to be declared in the code-behind file. Furthermore, the control had to be declared with exactly the same ID. Otherwise, the inheritance relationship would be broken and events raised by a control could not be handled in the code-behind file.

In the beta version of ASP.NET 2.0, a completely different method of associating presentation pages with their code-behind files was used. This new method was far less brittle. The two halves of a code-behind page were no longer related through inheritance, but through a new technology supported by .NET 2.0 Framework called *partial classes*.

NOTE

Chapter 17, "Building Components," discusses partial classes.

Partial classes enable you to declare a class in more than one physical file. When the class is compiled, one class is generated from all the partial classes. Any members of one partial class—including any private fields, methods, and properties—are accessible to any other partial classes of the same class. This makes sense because partial classes are combined eventually to create one final class.

The advantage of using partial classes is that you don't need to worry about declaring a control in both the presentation page and code-behind file. Anything that you declare in the presentation page is available automatically in the code-behind file, and anything you declare in the code-behind file is available automatically in the presentation page.

The beta version of ASP.NET 2.0 Framework used partial classes to relate a presentation page with its code-behind file. However, certain advanced features of ASP.NET 1.x Framework were not compatible with using partial classes. To support these advanced features, a more complex method of associating presentation pages with code-behind files is used in the final release of ASP.NET 2.0 Framework. This method is still the standard in ASP.NET 4.

Since version 2.0 of ASP.NET, the framework uses a combination of inheritance and partial classes to relate presentation pages and code-behind files. Three classes are generated whenever you create a code-behind page.

The first two classes correspond to the presentation page. For example, when you create the `FirstPageCodeBehind.aspx` page, the following two classes are generated automatically in the Temporary ASP.NET Files folder:

```
public partial class FirstPageCodeBehind
{
    protected System.Web.UI.WebControls.Button Button1;
    protected System.Web.UI.WebControls.Label Label1;

    ... additional code ...
}

public class firstpagecodebehind_aspx : FirstPageCodeBehind
{
    ... additional code ...
}
```

A third class is generated that corresponds to the code-behind file. Corresponding to the `FirstPageCodeBehind.aspx.cs` file, the following class is generated:

```
public partial class FirstPageCodeBehind : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        Label1.Text = "Click the Button";
    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        Label1.Text = "Thanks!";
    }
}
```

The `firstpagecodebehind_aspx` class is executed when the `FirstPageCodeBehind.aspx` page is requested from a browser. This class inherits from the `FirstPageCodeBehind` class. The `FirstPageCodeBehind` class is a partial class. It gets generated twice: once by the presentation page and once by the code-behind file.

The ASP.NET Framework uses a combination of partial classes and inheritance to relate presentation pages and code-behind files. Because the page and code-behind classes are partial classes, unlike the previous version of ASP.NET, you no longer need to declare controls in both the presentation and code-behind page. Any control declared in the presentation page is accessible in the code-behind file automatically. Because the page class inherits from the code-behind class, ASP.NET Framework continues to support advanced features of ASP.NET 1.x Framework, such as custom base `Page` classes.

Deciding Between Single-File and Code-Behind Pages

So, when should you use single-file ASP.NET pages and when should you use code-behind pages? This decision is a preference choice. There are intense arguments over this topic contained in blogs spread across the Internet.

I've heard it argued that code-behind pages are superior to single-file pages because code-behind pages enable you to more cleanly separate your user interface from your application logic. The problem with this argument is that the normal justification for separating your user interface from your application logic is code reuse. Building code-behind pages doesn't promote code reuse. A better way to reuse application logic across multiple pages is to build separate component libraries. (Part IV of this book explores this topic.)

Ultimately, it is a personal preference. The majority of enterprise projects that we have worked on utilize code-behind pages, but single-file pages are perfectly acceptable under the right circumstances.

Handling Page Events

Whenever you request an ASP.NET page, a particular set of events is raised in a particular sequence. This sequence of events is called the *page execution lifecycle*.

For example, we have already used the `Page Load` event in previous code samples. You normally use the `Page Load` event to initialize the properties of controls contained in a page. However, the `Page Load` event is only one event supported by the `Page` class.

Following is the sequence of events raised whenever you request a page:

1. `PreInit`
2. `Init`
3. `InitComplete`
4. `PreLoad`
5. `Load`
6. `LoadComplete`
7. `PreRender`
8. `PreRenderComplete`
9. `SaveStateComplete`
10. `Unload`

Why so many events? Different things happen and different information is available at different stages in the page execution lifecycle.

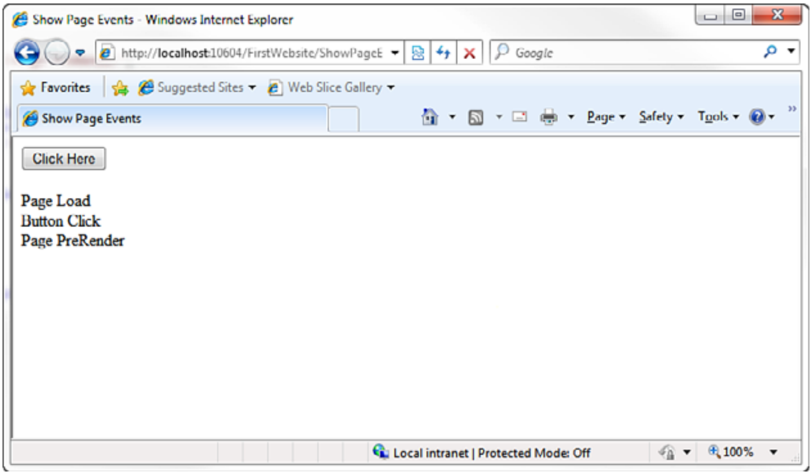
For example, View State is not loaded until after the `InitComplete` event. Data posted to the server from a form control, such as a `TextBox` control, is also not available until after this event.

Ninety-nine percent of the time, you won't handle any of these events except for the `Load` and the `PreRender` events. The difference between these two events is that the `Load` event happens before any control events, and the `PreRender` event happens after any control events.

The page in Listing 1.16 illustrates the difference between the `Load` and `PreRender` events. The page contains three event handlers: one for the `Load` event, one for the `Button Click` event, and one for the PreRender event. Each handler adds a message to a Label control (Figure 1.12).

LISTING 1.16 ShowPageEvents.aspx

```
<%@ Page Language="C#" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<script runat="server">

    void Page_Load(object sender, EventArgs e)
    {
        Label1.Text = "Page Load";
    }

    void Button1_Click(object sender, EventArgs e)
    {
        Label1.Text += "<br />Button Click";
    }

    void Page_PreRender()
    {
        Label1.Text += "<br />Page PreRender";
    }
</script>
<html xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1" runat="server">
    <title>Show Page Events</title>
</head>
<body>
    <form id="form1" runat="server">
    <div>

    <asp:Button
        id="Button1"
        Text="Click Here"
        OnClick="Button1_Click"
        Runat="server" />

    <br /><br />

    <asp:Label
        id="Label1"
        Runat="server" />

    </div>
    </form>
</body>
</html>
```



Show Page Events - Windows Internet Explorer
http://localhost:10604/FirstWebsite/ShowPageE
Google
Favorites
Suggested Sites
Web Slice Gallery
Show Page Events
Page
Safety
Tools
Local intranet | Protected Mode: Off
100%
Click Here

Page Load
Button Click
Page PreRender

FIGURE 1.12 Viewing the sequence of page events.

When you click the Button control, the `Click` event does not happen on the server until after the `Load` event and before the `PreRender` event.

The other thing you should notice about the page in Listing 1.16 is the way the event handlers are wired to the `Page` events. ASP.NET pages support a feature named `AutoEventWireUp`, which is enabled by default. If you name a subroutine `Page_Load()`, the subroutine automatically handles the `Page Load` event; if you name a subroutine `Page_PreRender()`, the subroutine automatically handles the `Page PreRender` event, and so on.

WARNING

`AutoEventWireUp` does not work for every page event. For example, it does not work for the `Page_InitComplete()` event.

Using the `Page.IsPostBack` Property

The `Page` class includes a property called the `IsPostBack` property, which you can use to detect whether the page has already been posted back to the server.

Because of View State, when you initialize a control property, you do not want to initialize the property every time a page loads. Because View State saves the state of control properties across page posts, you typically initialize a control property only once, when the page first loads.

Many controls don't work correctly if you reinitialize the properties of the control with each page load. In these cases, you must use the IsPostBack property to detect whether the page has been posted.

The page in Listing 1.17 illustrates how you can use the Page.IsPostBack property when adding items to a DropDownList control.

LISTING 1.17 ShowIsPostBack.aspx

```
<%@ Page Language="C#" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<script runat="server">

    void Page_Load()
    {
        if (!Page.IsPostBack)
        {
            // Create collection of items
            ArrayList items = new ArrayList();
            items.Add("Apples");
            items.Add("Oranges");

            // Bind to DropDownList
            DropDownList1.DataSource = items;
            DropDownList1.DataBind();
        }
    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        Label1.Text = DropDownList1.SelectedItem.Text;
    }
</script>
<html xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1" runat="server">
    <title>Show IsPostBack</title>
</head>
<body>
    <form id="form1" runat="server">
    <div>

    <asp:DropDownList
        id="DropDownList1"
        Runat="server" />

    <asp:Button
```

```
        id="Button1"
        Text="Select"
        OnClick="Button1_Click"
        Runat="server" />

    <br /><br />

    You selected:
    <asp:Label
        id="Label1"
        Runat="server" />

    </div>
    </form>
</body>
</html>
```

In Listing 1.17, the code in the `Page_Load()` event handler executes only once when the page first loads. When you post the page again, the `IsPostBack` property returns `True` and the code contained in the `Page_Load()` handler is skipped.

If you remove the `IsPostBack` check from the `Page_Load()` method, you get a strange result. The `DropDownList` always displays its first item as the selected item. Binding the `DropDownList` to a collection of items reinitializes the `DropDownList` control. Therefore, you want to bind the `DropDownList` control only once, when the page first loads.

Debugging and Tracing ASP.NET Pages

The sad fact of life is that you spend the majority of your development time when building applications debugging the application. In this section, you learn how to get detailed error messages when developing ASP.NET pages. You also learn how you can display custom trace messages that you can use when debugging a page.

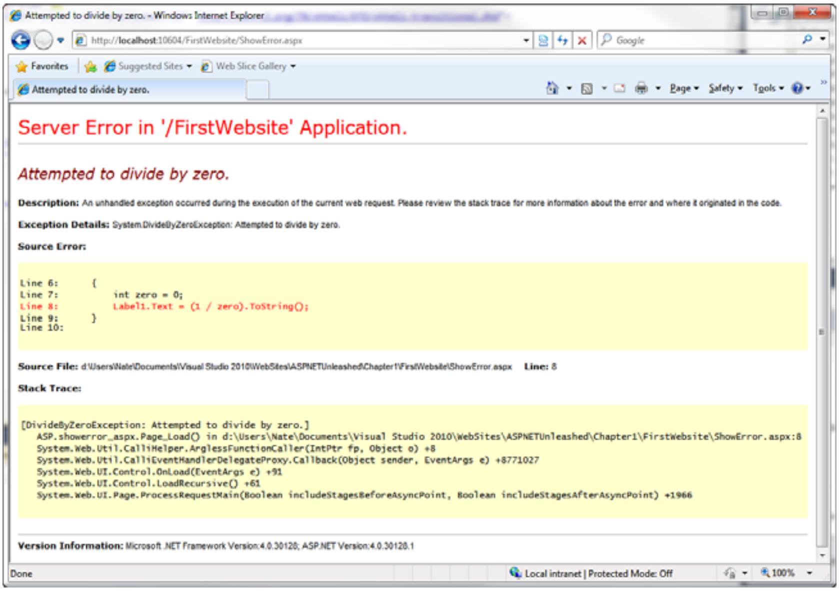
Debugging ASP.NET Pages

If you need to view detailed error messages when you execute a page, you need to enable debugging for either the page or your entire application. You can enable debugging for a page by adding a `Debug="true"` attribute to the `<%@ Page %>` directive. For example, the page in Listing 1.18 has debugging enabled.

LISTING 1.18 `ShowError.aspx`

```
 <%@ Page Language="C#" Debug="true" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<script runat="server">

    void Page_Load()
    {
        int zero = 0;
        Label1.Text = (1 / zero).ToString();
    }

</script>
<html xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1" runat="server">
    <title>Show Error</title>
</head>
<body>
    <form id="form1" runat="server">
    <div>

    <asp:Label
        id="Label1"
        Runat="server" />

    </div>
    </form>
</body>
</html>
```

When you open the page in Listing 1.18 in your web browser, a detailed error message displays (see Figure 1.13).


Attempted to divide by zero. - Windows Internet Explorer
http://localhost:10604/FirstWebsite/ShowError.aspx
Google
Favorites
Suggested Sites
Web Slice Gallery
Attempted to divide by zero.
Page
Safety
Tools
Done
Local intranet | Protected Mode: Off
100%

Server Error in '/FirstWebsite' Application.

Attempted to divide by zero.

Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.

Exception Details: System.DivideByZeroException: Attempted to divide by zero.

Source Error:

```
Line 6:       {
Line 7:           int zero = 0;
Line 8:           Label1.Text = (1 / zero).ToString();
Line 9:       }
Line 10:
```

Source File: d:\Users\Nate\Documents\Visual Studio 2010\WebSites\ASPNETUnleashed\Chapter1\FirstWebsite\ShowError.aspx **Line:** 8

Stack Trace:

```
[DivideByZeroException: Attempted to divide by zero.]
   ASP.showerror_aspx.Page_Load() in d:\Users\Nate\Documents\Visual Studio 2010\WebSites\ASPNETUnleashed\Chapter1\FirstWebsite\ShowError.aspx:8
   System.Web.Util.CalliHelper.ArglessFunctionCaller(IntPtr fp, Object o) +8
   System.Web.Util.CalliEventHandlerDelegateProxy.Callback(Object sender, EventArgs e) +8771027
   System.Web.UI.Control.OnLoad(EventArgs e) +91
   System.Web.UI.Control.LoadRecursive() +61
   System.Web.UI.Page.ProcessRequestMain(Boolean includeStagesBeforeAsyncPoint, Boolean includeStagesAfterAsyncPoint) +1966
```

Version Information: Microsoft .NET Framework Version:4.0.30128; ASP.NET Version:4.0.30128.1

FIGURE 1.13 Viewing a detailed error message.

WARNING

Make sure that you disable debugging before placing your application into production. When an application is compiled in debug mode, the compiler can't make certain performance optimizations.

Rather than enable debugging for a single page, you can enable debugging for an entire application by adding the web configuration file in Listing 1.19 to your application.

LISTING 1.19 `Web.Config`

```
<?xml version="1.0"?>
<configuration>
<system.web>
  <compilation debug="true" />
</system.web>
</configuration>
```

When debugging an ASP.NET application located on a remote web server, you need to disable custom errors. For security reasons, by default, ASP.NET Framework doesn't display error messages when you request a page from a remote machine. When custom errors are enabled, you don't see errors on a remote machine. The modified web configuration file in Listing 1.20 disables custom errors.

LISTING 1.20 `Web.Config`

```
<?xml version="1.0"?>
<configuration>
<system.web>
  <compilation debug="true" />
  <customErrors mode="Off" />
</system.web>
</configuration>
```

WARNING

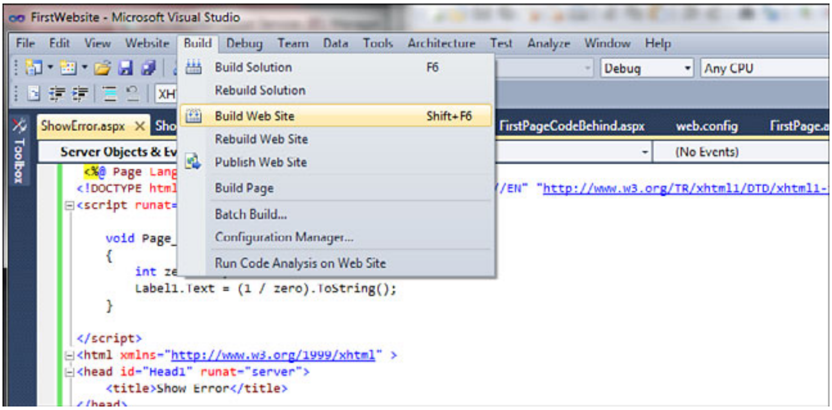
For security and performance reasons, don't put websites into production with debug enabled, custom errors disabled, or trace enabled. On your production server, add the following element inside the system.web section of your machine.config file:

```
<deployment retail="true"/>
```

Adding this element disables debug mode, enables remote custom errors, and disables trace. You should add this element to the machine.config file located on all your production servers.

Debugging Pages with Visual Web Developer

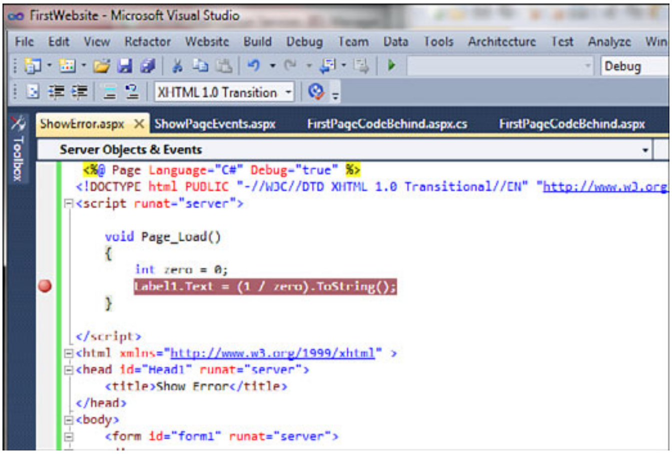
If you use Visual Web Developer, you can display compilation error messages by performing a build on a page or an entire website. Select Build, Build Page or Build, Build Web Site. A list of compilation error messages and warnings appears in the Error List window (see Figure 1.14). You can double-click any of the errors to navigate directly to the code that caused the error.



```
<%@ Page Lang
<!DOCTYPE html                                    /EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
<script runat=

    void Page_
    {
        int ze
        Label1.Text = (1 / zero).ToString();
    }

</script>
<html xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1" runat="server">
    <title>Show Error</title>
</head>
```

FIGURE 1.14 Performing a build in Visual Web Developer.

If you need to perform more advanced debugging, you can use the Visual Web Developer's debugger. The debugger enables you to set breakpoints and step line by line through your code. You set a breakpoint by double-clicking the left-most column in Source view. When you add a breakpoint, a red circle appears (see Figure 1.15).


FirstWebsite - Microsoft Visual Studio
File Edit View Refactor Website Build Debug Team Data Tools Architecture Test Analyze Win
Debug
XHTML 1.0 Transition
ShowError.aspx
ShowPageEvents.aspx
FirstPageCodeBehind.aspx.cs
FirstPageCodeBehind.aspx
Server Objects & Events
Toolbox

```
<%@ Page Language="C#" Debug="true" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org
<script runat="server">

    void Page_Load()
    {
        int zero = 0;
        Label1.Text = (1 / zero).ToString();
    }

</script>
<html xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1" runat="server">
    <title>Show Error</title>
</head>
<body>
    <form id="form1" runat="server">
```

FIGURE 1.15 Setting a breakpoint.

After you set a breakpoint, run your application by selecting Debug, Start Debugging. Execution stops when the breakpoint is hit. At that point, you can hover your mouse over any variable or control property to view the current value of the variable or control property.

NOTE

You can designate one of the pages in your application as the Start Page. That way, whenever you run your application, the Start Page is executed regardless of the page that you have open. Set the Start Page by right-clicking a page in the Solution Explorer window and selecting Set As Start Page.

After you hit a breakpoint, you can continue execution by selecting Step Into, Step Over, or Step Out from the Debug menu or the toolbar. Here's an explanation of each of these options:

- **Step Into—**Executes the next line of code
- **Step Over—**Executes the next line of code without leaving the current method
- **Step Out—**Executes the next line of code and returns to the method that called the current method

When you finish debugging a page, you can continue, stop, or restart your application by selecting a particular option from the Debug menu or the toolbar.

Tracing Page Execution

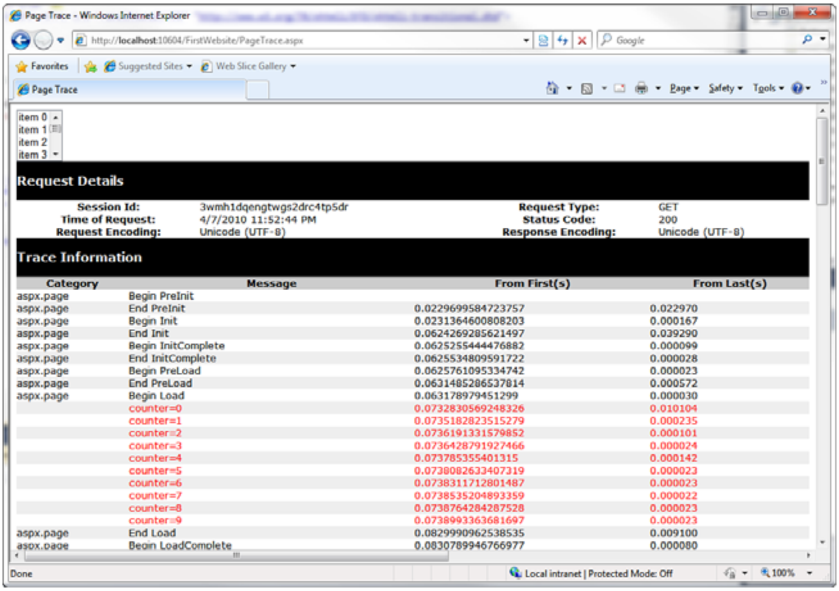
If you want to output trace messages while a page executes, you can enable tracing for a particular page or an entire application. The ASP.NET Framework supports both page-level tracing and application-level tracing.

The page in Listing 1.21 illustrates how you can take advantage of page-level tracing.

LISTING 1.21 PageTrace.aspx

```
 <%@ Page Language="C#" Trace="true" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<script runat="server">

    void Page_Load()
    {
        for (int counter = 0; counter < 10; counter++)
        {
           ListBox1.Items.Add("item " + counter.ToString());
            Trace.Warn("counter=" + counter.ToString());
        }
    }

</script>
<html xmlns="http://www.w3.org/1999/xhtml" >
<head id="Head1" runat="server">
    <title>Page Trace</title>
</head>
<body>
    <form id="form1" runat="server">
    <div>

    <asp:ListBox
        id="ListBox1"
        Runat="server" />

    </div>
    </form>
</body>
</html>
```

The `<%@ Page %>` directive in Listing 1.21 includes a `trace="true"` attribute. This attribute enables tracing and causes a Trace Information section to be appended to the bottom of the page (see Figure 1.16).


Page Trace - Windows Internet Explorer
http://localhost:10604/FirstWebsite/PageTrace.aspx
Google
Favorites
Suggested Sites
Web Slice Gallery
Page Trace
Page
Safety
Tools
Done
Local intranet | Protected Mode: Off
100%

item 0
item 1
item 2
item 3

Request Details

Session Id:	3wmh1dqengtwgs2drc4tp5dr	**Request Type:**	GET
Time of Request:	4/7/2010 11:52:44 PM	**Status Code:**	200
Request Encoding:	Unicode (UTF-8)	**Response Encoding:**	Unicode (UTF-8)

Trace Information

Category	Message	From First(s)	From Last(s)
aspx.page	Begin PreInit		
aspx.page	End PreInit	0.0229699584723757	0.022970
aspx.page	Begin Init	0.0231364600808203	0.000167
aspx.page	End Init	0.0624269285621497	0.039290
aspx.page	Begin InitComplete	0.0625255444476882	0.000099
aspx.page	End InitComplete	0.0625534809591722	0.000028
aspx.page	Begin PreLoad	0.0625761095334742	0.000023
aspx.page	End PreLoad	0.0631485286537814	0.000572
aspx.page	Begin Load	0.063178979451299	0.000030
	counter=0	0.0732830569248326	0.010104
	counter=1	0.0735182823515279	0.000235
	counter=2	0.0736191331579852	0.000101
	counter=3	0.0736428791927466	0.000024
	counter=4	0.073785355401315	0.000142
	counter=5	0.0738082633407319	0.000023
	counter=6	0.0738311712801487	0.000023
	counter=7	0.0738535204893359	0.000022
	counter=8	0.0738764284287528	0.000023
	counter=9	0.0738993363681697	0.000023
aspx.page	End Load	0.0829990962538535	0.009100
aspx.page	Begin LoadComplete	0.0830789946766977	0.000080

FIGURE 1.16 Viewing page trace information.

Notice, furthermore, that the Page_Load() handler uses the Trace.Warn() method to write messages to the Trace Information section. You can output any string to the Trace Information section that you want. In Listing 1.21, the current value of a variable named counter displays.

You need to take advantage of page tracing when you want to determine exactly what is happening when a page executes. You can call the Trace.Warn() method wherever you need in your code. Because the Trace Information section appears even when an error exists on your page, you can use tracing to diagnose the causes of any page errors.

One disadvantage of page tracing is that everyone in the world gets to see your trace information. You can get around this problem by taking advantage of application-level tracing. When application-level tracing is enabled, trace information appears only when you request a special page named Trace.axd.

To enable application-level tracing, you need to add the web configuration file in Listing 1.22 to your application.

LISTING 1.22 Web.Config

```
<?xml version="1.0"?>
<configuration>
<system.web>
    <trace enabled="true" />
</system.web>
</configuration>
```

After you add the Web.Config file in Listing 1.22 to your application, you can request the Trace.axd page in your browser. The last 10 page requests made after application-level tracing is enabled display.

WARNING

By default, the Trace.axd page cannot be requested from a remote machine. If you need to access the Trace.axd page remotely, you need to add a localOnly="false" attribute to the trace element in the web configuration file.

If you click the View Details link next to any of the listed page requests, you can view all the trace messages outputted by the page. Messages written with the Trace.Warn() method display by the Trace.axd page even when page-level tracing is disabled.

NOTE

You can use the new `writeToDiagnosticsTrace` attribute of the `trace` element to write all trace messages to the Output window of Visual Web Developer when you run an application. You can use the new `mostRecent` attribute to display the last 10 page requests rather than the 10 page requests after tracing was enabled.

WARNING

If you don’t enable the `mostRecent` attribute when application level tracing is enabled, tracing stops after 10 pages.

Installing ASP.NET

The easiest way to install ASP.NET Framework is to install Visual Web Developer Express. You can download the latest version of Visual Web Developer from www.ASP.net, which is the official Microsoft ASP.NET website.

Installing Visual Web Developer Express also installs the following components:

- Microsoft .NET Framework version 4
- SQL Server Express

Visual Web Developer Express is compatible with the following operating systems:

- Windows XP (x86) Service Pack 3
- Windows XP (x64) Service Pack 2
- Windows Server 2003 Service Pack 2
- Windows Server 2003 R2
- Windows Server 2008 Service Pack 2
- Windows Server 2008 R2
- Windows Vista
- Windows 7

You can install Visual Web Developer Express on a computer that already has other versions of Visual Studio or Visual Web Developer installed. Different versions of the development environments can coexist peacefully.

Furthermore, the same web server can serve ASP.NET 1.1 pages, ASP.NET 2.0 pages, ASP.NET 3.0 pages, ASP.NET 3.5 pages, and ASP.NET 4 pages. Each version of .NET Framework is installed in the following folder:

```
C:\WINDOWS\Microsoft.NET\Framework
```

For example, on my computer, I have the following six versions of .NET Framework installed (version 1.0, version 1.1, version 2.0, version 3.0, version 3.5, and version 4):

```
C:\WINDOWS\Microsoft.NET\Framework\v1.0.3705
C:\WINDOWS\Microsoft.NET\Framework\v1.1.4322
C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727
C:\WINDOWS\Microsoft.NET\Framework\v3.0
C:\WINDOWS\Microsoft.NET\Framework\v3.5
C:\WINDOWS\Microsoft.NET\Framework\v4.0.30128
```

NOTE

The Framework directory contains the 32-bit (x86) version of .NET. If you are running on a 64-bit (x64) operating system, you also have another directory named Framework64.

All the folders except for v3.0 and v3.5 include a command-line tool named aspnet_regiis.exe. You can use this tool to associate a particular virtual directory on your machine with a particular version of .NET Framework.

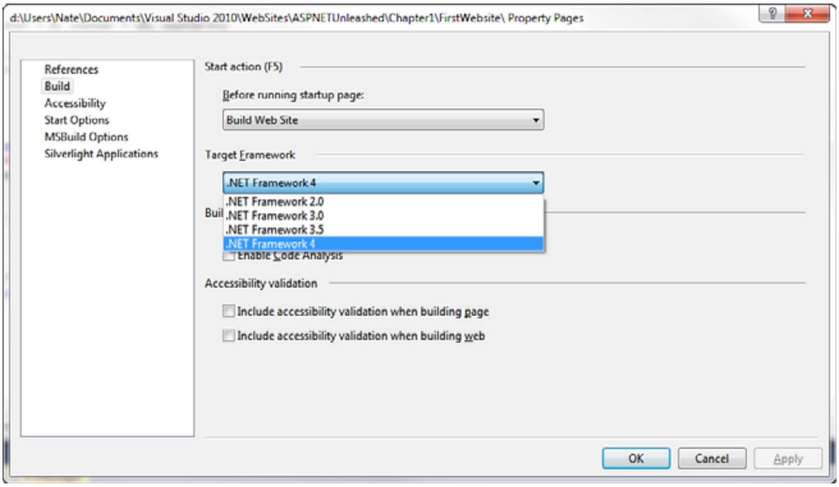
For example, executing the following command from a command prompt located in the v1.0.3705, v1.1.4322, v2.0.50727, or v4.0.30128 folders enables the 1.0, 1.1, 2.0, or 4 version of ASP.NET for a virtual directory named MyApplication:

```
aspnet_regiis -s W3SVC/1/ROOT/MyApplication
```

By executing the aspnet_regiis.exe tool located in the different .NET Framework version folders, you can map a particular virtual directory to any version of ASP.NET Framework.

The .NET Frameworks 3.0 and 3.5 work differently than earlier versions. The 3.0 and 3.5 versions build on top of the existing .NET Framework 2.0. To use these versions of .NET Framework, you need to add the correct assembly references to your website and use the correct versions of the C# or VB.NET compilers. You reference these assemblies and configure the compiler within your application's web.config file. When you create a new website in Visual Web Developer, the necessary configuration settings are included in your web.config file automatically. The .NET Framework 4 is the first version since 2.0 that does not build off of a previous version.

You also have the option of targeting a particular version of .NET Framework. To do this, select Website, Start Options and select the Build tab. You can choose to target .NET Framework 2.0, .NET Framework 3.0, .NET Framework 3.5, or .NET Framework 4 (see Figure 1.17).



d:\Users\Nate\Documents\Visual Studio 2010\WebSites\ASPNETUnleashed\Chapter1\FirstWebsite\ Property Pages

Start action (F5)

Before running startup page: Build Web Site

Target Framework: .NET Framework 4

□ Enable Code Analysis

Accessibility validation

□ Include accessibility validation when building page

□ Include accessibility validation when building web

FIGURE 1.17 Targeting a particular version of .NET Framework.

NOTE

If you load an existing ASP.NET 2.0, 3.0, or 3.5 website into Visual Web Developer 2010, Visual Web Developer prompts you to upgrade the website to ASP.NET 4. When Visual Web Developer upgrades your website, it modifies your web.config file.

Summary

In this chapter, you were introduced to ASP.NET 4 Framework. First, we built a simple ASP.NET page. You learned about the three main elements of an ASP.NET page: directives, code declaration blocks, and page render blocks.

Next, we discussed .NET Framework. You learned about the 13,000 classes contained in the Framework Class Library and about the features of the Common Language Runtime.

You also were provided with an overview of ASP.NET controls. You learned about the different groups of controls included in .NET Framework. You also learned how to handle control events and take advantage of View State.

We also discussed ASP.NET pages. You learned how ASP.NET pages are dynamically compiled when they are first requested. We also examined how you can divide a single-file ASP.NET page into a code-behind page. You learned how to debug and trace the execution of an ASP.NET page.

At the end of the chapter we covered installation issues in getting ASP.NET Framework up and running. You learned how to map different Virtual Directories to different versions of ASP.NET Framework. You also learned how to target different versions of .NET Framework in your web configuration file.

Index

Symbols

$.ajax call, 1779

@Director parameter, 512

$(document).ready( ) method, jQuery, 1773-1774

/ (forward slash), 1124

$get( ) method, 1754

@Id parameter, 512

$( ) method, jQuery, 1773-1774

%@ OutputCache % directive, 1337, 1369

- CacheProfile attribute, 1361
- Location attribute, 1349, 1351
- VaryByControl attribute, 1342, 1344
- VaryByCustom attribute, 1346-1349
- VaryByHeader attribute, 1345-1346
- VaryByParam attribute, 1338, 1342

~ (tilde character), 1124

@Title parameter, 512

_ (underscore character), 730

A

a templates, PagerTemplate, 509

Abandon( ) method, 1284-1285

About.asax file, 1092-1093

absolute cache expiration policy, 1376-1377

absolute expiration policy, Data Caching, 1390-1392

absolute layouts, 288-290

Absolute.css file, 288-290

Accept Changes button, 906

Accept-Language headers, 1428

AcceptChanges( ) method, 902

accepting

form input, 1507-1509

user input. *See* user input, accepting

access. *See also* data access

controllers, 1506

WCF Data Services, 1001

data context, 1007-1009

service references, 1002-1007

access modifiers, 743-744

AccessDataSource control, 351, 379

accessibility guidelines

GridView control, 487

labels, 65

AccessibleHeaderText property (BoundField), 525

AccessKey property

Button control, 85

CheckBox control, 78

ImageButton control, 93

ImageMap control, 110

LinkButton control, 88

RadioButton control, 81

TextBox control, 70

Accordion control, 1737

ActionUrl property (PostBackOptions class), 1614

Activate event

View control, 220

WizardStep, 227

Active Directory Application Mode (ADAM), 1240-1242

ActiveDirectoryMembershipProvider, 1226, 1239, 1242-1244

ActiveStep property (Wizard control), 226

ActiveStepChanged event, 227

ActiveStepIndex property (Wizard control), 226

ActiveViewChanged event, 220

ActiveViewIndex property (MultiView control), 220

AD LDS instances, creating, 1240

ADAM (Active Directory Application Mode), 1240-1242

AdCreated event, 207, 214

AdCreated event handler, 214-215

Add Extender task, 1735

Add New Item command (Website menu), 5, 722

Add New Item dialog box, 373

Add( ) method, 984, 1387

AddAttribute( ) method, 1580, 1582

AddAttributesToRender( ) method, 1586, 1616

AddDesignTimeCssAttributes( ) method, 1639

Added value, 901

AddFile( ) method, 193

AddFileDependencies( ) method, 1353

AddFileDependency( ) method, 1353

adding

automatic properties, 937

comments, 745

entities, 1006

event handlers, 24

GridView control, 487

items to caches, 1390

Login controls, 1157

Menu items, 1021-1024

service references, 1003

SiteMapPath controls, 1015

SqlDataSource control, 487

titles to charts, 706

AddMovie.aspx page, 1354-1355

AddMovieDataSourceKeyDependency.aspx page, 1386

AddMovieKeyDependency.aspx page, 1357

AddOnPreRenderCompleteAsync( ) method, 917

AddParameter.aspx page, 426, 428

AddParsedSubObject( ) method, 1619, 1626

addresses (email), validating, 151, 153

AddressForm control, 311, 315, 317

AddressForm.ascx, 311, 315

Checkout.ascx example, 315, 317

AddStyleAttribute( ) method, 1580

AddUsersToRole( ) method, 1258

AddUsersToRoles( ) method, 1258

AddUserToRole( ) method, 1258

AddUserToRoles( ) method, 1258

AddValidationCallback( ) method, 1358

AddWithValue( ) method, 863-864

AdHandler.ashx page, 216

AdList.xml file, 211-212

AdminUsersSection class, 1553

AdminUsersSection.cs file, 1553

ADO.NET

- asynchronous ASP.NET pages, 914, 917
- asynchronous database commands, 911-912, 914
- Command object, 859
  - executing commands with parameters, 863-864
  - executing commands without parameters, 860, 862
  - executing stored procedures, 865-872
  - returning resultsets, 876-881
  - returning single values, 874, 876
- Connection object, 848-849
  - Connection pooling, 856-858
  - provider statistics, retrieving, 849-856
- DataAdapter object, 889
  - batch updates, performing, 892-893, 896
  - displaying data with, 890, 892
  - populating DataTables with, 889-890
- DataReader object, 881
  - Multiple Active Results Sets (MARS), 888
  - multiple resultsets, returning, 882, 884
  - Read( ) method, 881
- DataSet object, 908, 911
- DataTable object, 896
  - building programmatically, 896, 898
  - DataRow states and versions, 901, 903, 906
  - DataRows, selecting, 898, 901
- DataView object, 906
- parameters, adding to SqlDataSource commands, 426, 428

AdRotator control, 214, 1336

- AdCreated event, 207
- displaying advertisements, 207-208
- properties, 207
- storing advertisements in database tables, 212-213
- storing advertisements in XML files, 208-212
  - AdList.xml file, 211-212
  - AdRotatorXML.aspx page, 208
- tracking impressions and transfers, 213-219
  - AdHandler.ashx page, 216
  - AdRotatorStats.aspx page, 218
  - AdRotatorTrack.aspx page, 214

AdRotatorDatabase.aspx page, 212

AdRotatorStats.aspx page, 218

AdRotatorTrack.aspx page, 214

AdRotatorXML.aspx page, 208

AdStats database table, 215

AdvancedCheckBox control

- AdvancedCheckBox.cs file, 1614
- ShowAdvancedCheckBox.aspx page, 1616

AdvancedCheckBox.cs file, 1614

AdvertisementFile property (AdRotator control), 207

advertisements

- displaying, 207-208
- impressions and transfers, tracking, 213-219
  - AdHandler.ashx page, 216
  - AdRotatorStats.aspx page, 218
  - AdRotatorTrack.aspx page, 214
- storing in database tables, 212-213
- storing in XML files, 208-212
  - AdList.xml file, 211-212
  - AdRotatorXML.aspx page, 208

Aggregate( ) method, 951

AggregateCacheDependency class, 1394

How can we make this index more useful? Email us at indexes@samspublishing.com

AJAX (Asynchronous JavaScript and XML), 317, 320, 578
- debugging, 1675-1677
- jQuery. *See* jQuery
- overview of, 1674-1675
- paging through data with, 578-579
- paging through GridView data, 503-504
- server-side versus client-side, 1675
- sorting data with, 496-497
- Timer controls, 1724-1728
- TreeView controls, 1066-1067, 1070
- UpdatePanel control, 1677-1685
  - aborting current asynchronous postbacks, 1710
  - canceling current asynchronous postbacks, 1707, 1709
  - execution life cycles, 1697-1707
  - handling errors, 1714-1717, 1720
  - JavaScript, 1694-1697
  - nesting, 1688, 1692
  - passing current asynchronous postbacks, 1712, 1714
  - performance, 1720, 1723-1724
  - triggers, 1686-1687
  - updating, 1692-1693
- UpdateProgress controls, 1728-1729, 1731
- Web services, calling, 1776-1781

AJAX Control Toolkit, 1734, 1736
- controls, 1737-1739
  - Animation, 1759, 1761, 1763-1764
  - AutoComplete, 1739-1750
  - DragPanel, 1750-1751, 1754
  - FilteredTextBox, 1754, 1756
  - MaskedEdit, 1756, 1758-1759
  - UpdatePanelAnimation, 1764-1768

Ajax Request file, 1720

Ajax Response file, 1720

AjaxPaging.aspx page, 503-504

AjaxSorting.aspx page, 496-497

AjaxValidator control, 167
- AjaxValidator.js file, 173-174
- AjaxValidator.cs file, 171
- ShowAjaxValidator.aspx page, 174

AjaxValidator.js file, 173-174

AjaxValidator.cs file, 171

AjaxValidatorEvaluateIsValid( ) function, 173-174

AjaxValidatorResult( ) method, 174

alert boxes, displaying, 165-166

All value, 1054

allowAnonymous attribute, 1298, 1303-1304

allowCustomSqlDatabase attribute, 1297

AllowPaging property (GridView control), 501, 503

AllowReturn property (WizardStep control), 227

AllowSorting property (GridView), 495-496

AllowWhiteSpaceLiterals( ) method, 1626

ALTER DATABASE command, 1409

AlternateText property
- Image control, 105
- ImageButton control, 93
- ImageField, 543
- ImageMap control, 110

AlternateTextField property (AdRotator control), 207

AlternatingItemStyle property (DataList control), 657

AlternatingItemTemplate property (TemplateField), 546

AlternatingItemTemplate template, 627, 645, 662, 664

AlternatingRowStyle property (DetailsView control), 601

alternative text for images, 90

AlwaysVisibleControl control, 1737

analysis, Chart control, 698, 700-701, 704

And( ) method, 984

Animation control, 1737, 1759-1764

anonymous functions, 1775

anonymous HTTP Handlers, creating, 1478-1481
- IHttpAsyncHandler interface, 1478
- RSSHandler.cs file, 1479
- ShowRSSHandler.aspx page, 1481
- Web.Config file, 1480

Anonymous Identification, 1303

anonymous profiles, 1306-1308

anonymous types, 940

anonymous users, supporting in Profiles, 1303-1307

anonymousIdentification configuration section, 1515

Answer ID, 1180

Any value, 1349

AppendCacheExtension( ) method, 1358

AppendDataBoundItems property (List controls), 450, 452

appending list items, 450, 452

AppendListItems.aspx page, 451

AppendLiteralString( ) method, 1626

AppendTrailingSlash( ) method, 1135

AppInitialize( ) method, 1135

Application.init event, 1700

Application.load event, 1700

Application.unload event, 1700

ApplicationLoadEventArgs, 1704

applications

- AJAX. *See* AJAX
- architecture, 757
  - Business Logic layer, 761, 765
  - Data Access layer, 765, 769
  - multi-tier applications, building, 757-758
  - User Interface layer, 758-761
- configuring for Polling SQL cache dependencies, 1402-1403
- configuring for Push SQL cache dependencies, 1410
- databases, deploying, 1571-1572
- dynamic data, 1491-1492
  - building, 1492, 1494, 1496
  - templates, 1496-1497, 1500
- importing, 1568
- localizing, 1419-1420
  - applying Localize controls, 1450-1452
  - ASP.NET controls, 1433-1434
  - configuring cultures manually, 1421-1428
  - CultureInfo class, 1434-1438
  - deleting cultures automatically, 1428, 1431-1432
  - global resources, 1446-1450
  - local resources, 1438-1445
  - UICulture property, 1420-1421
  - web configuration files, 1432-1433
- MVC Framework, 1501
  - accepting form input, 1507, 1509
  - controllers, 1502
  - creating, 1502, 1504-1505
  - models, 1502
  - pages, 1505-1507
  - views, 1502
- one-click, 1572-1573
- packaging, 1565-1566, 1568
- reading configuration sections from, 1524-1527
  - ShowConfigApp.aspx page, 1524-1525
  - ShowConfigPath.aspx page, 1528
  - ShowConfigRelative.aspx page, 1525, 1527
- URL routing, 1137-1139, 1143-1146
  - linking to other pages with routes, 1142-1143
  - mapping, 1139-1142
  - scenarios, 1139
- Web.config file transformations, 1569-1571

Application_BeginRequest( ) handler, 1485

Application_Start( ) event handler, 1091

Application_Start( ) method, 1523

ApplyAppPathModifier( ) method, 1290

ApplyFormatInEditMode property (BoundField), 525

applying

- constrained routes, 1144-1145
- partial classes, 42
- Visual Web Developer, 40

appSettings configuration, 1521

appSettings element (web.config file), 1520, 1522-1523

AppSettings property

- Configuration class, 1536
- WebConfigurationManager class, 1523

appSettings.config file, 1523

AppSettingsExpressionBuilder class, 1464

App_Code folder, mixing different language components, 725-726

App_Code\AutoSiteMapProvider.cs file, 1106, 1109

App_Code\Movie1.cs file, 845

App_Code\SqlSiteMapProvider.cs file, 1111, 1114

App_GlobalResources folder, 1447

App_Themes folder, 270

App_Themes\StyleTheme\SimpleStyle.css file, 281-282

App_WebReferences folder, 780

architecture, 757
- Business Logic layer, 761, 765
- Data Access layer, 765, 769
- multi-tier applications, building, 757-758
- User Interface layer, 758-761

archive.xml file, 1567

areas
- charts, 690, 694
- plotting, customizing, 704-708

Argument property (PostBackOptions class), 1614

arrays, initializers, 941

Article control, 1649
- Article.cs file, 1648
- implementing on pages, 1650

Article.cs file, 1648

ArticleWithDefault control
- ArticleWithDefault.cs file, 1652
- implementing on pages, 1655

ArticleWithDefault.cs file, 1652

ascending sorts, 496

ASP.NET configuration sections
- custom configuration sections, creating, 1546-1549
  - ConfigurationProperty attribute, 1548
  - validators, 1548
  - web.config registration, 1549
- encrypted configuration sections, 1556-1557
  - creating programmatically, 1558, 1561
  - creating with aspnet_regiis command, 1557-1558
  - deploying, 1561-1564
  - DpapiProtectedConfigurationProvider, 1557
  - RsaProtectedConfigurationProvider, 1557
- list of, 1515-1517
- modifying, 1539, 1541-1542
- reading from current application, 1524-1525, 1527
  - ShowConfigApp.aspx page, 1524-1525
  - ShowConfigPath.aspx page, 1528
  - ShowConfigRelative.aspx page, 1525, 1527

ASP.NET Framework installation, 55-56

ASP.NET Membership. *See* Membership

ASP.NET pages
- code-behind pages, 39-43
  - FirstPageCodeBehind.aspx page, 40-41
  - FirstPageCodeBehind.aspx.cs file, 41
  - partial classes, 42
  - presentation pages, 41-43
  - when to use, 43
- control trees, 38-39
- creating, 5, 8-9
- debugging, 48, 50, 52
- dynamic compilation, 34-35, 37
- page events, handling, 44-46
- page execution lifestyle, 44
- Page.IsPostBack property, 46, 48
- tracing, 52-55

ASP.NET parameters, DataSource controls, 353, 355

ASP.NET SQL Server Setup Wizard, 1238, 1326

ASPNetSurvey.aspx control, 326-327

aspnet_compiler.exe tool, 34

aspnet_regiis command, 290, 389, 1237, 1533, 1557-1558

aspnet_regsql tool, 1326, 1401-1402

ASPSurvey.aspx control, 325-326

assemblies, 14, 17

adding to Bin folder, 753

adding to Global Assembly Cache, 754-757

definition of, 724

private, 15

referencing, 16-17

shared, 15

AssemblyBuilder class, 1455

AssemblyInfo.cs file, 755

assigning users to roles, 1249

AssociatedControlID property, 63-65

AssociatedUpdatePanelID property, 1731

associations, entities, 959-960

AsyncDataLayer component, 913-914

AsyncDataLayer.cs file, 913-914

AsyncFileUpload control, 1737

asynchronous ASP.NET pages, 914, 917

asynchronous database commands, 911-914

Asynchronous JavaScript and XML. *See* AJAX

asynchronous postbacks, 1707-1714

attacks, XPath Injection Attacks, 1246

attributeMapFailedPasswordAnswerCount( ) method, 1244

attributeMapFailedPasswordAnswerLockoutTime( ) method, 1244

attributeMapFailedPasswordAnswerTime( ) method, 1244

attributeMapPasswordAnswer( ) method, 1244

attributeMapPasswordQuestion( ) method, 1244

Attributes property (ListItem control), 441

Authenticate event, 1161

Authenticate( ) method, 1161, 1217, 1220

AuthenticateEventArgs class, 1161

authentication, 1149

custom authentication, Login control, 1160-1161

Forms authentication, 1206-1207

across applications, 1210-1214

across domains, 1214-1216

configuring, 1207-1208

cookieless Forms authentication, 1209-1210

enabling, 1148-1149

encryption and validation options, 1211-1214

FormsAuthentication class, 1216-1220

sliding expiration, 1210

User class, 1220-1221

Login controls, 1147

.NET Passport authentication, 1206

Windows authentication, 1206

authentication configuration section, 1515

AuthenticationType property (Identity object), 1221

authorization, 1149

configuring, 1221, 1223

by file location, 1224-1225

.gif image files, 1225

with images and other file types, 1225

by role, 1223

authorization configuration section, 1515

Authorization Manager, 1254

AuthorizationStoreRoleProvider, 1247, 1254-1256

AutoComplete class, 72, 74

AutoComplete control, 1737-1750

AutoCompletePageMethod.aspx page, 1740

AutoCompleteTextValue.aspx page, 1746

AutoCompleteType property (TextBox control), 70

AutoCompleteWebService.aspx page, 1744

AutoDetect, 1209-1210, 1289

AutoEventWireUp, 46

Autoformat, 719

AutoGenerateColumns property (GridView control), 525

AutoGenerateDeleteButton property, 597

AutoGenerateInsertButton property, 594

AutoGeneratePassword property (CreateUserWizard control), 1171

AutoGenerateSelectButton property (GridView control), 491

automatic postbacks (List controls), 452, 454

automatic properties, 730, 936

automatically saving Profiles, 1317, 1320

AutoPostBack property, 454

- CheckBox control, 78-79
- List controls, 452, 454
- PostBackOptions class, 1614
- RadioButton control, 81-83
- TextBox control, 70-72

AutoPostBackListControl.aspx page, 453-454

AutoSiteMapProvider, 1106, 1109-1110

- App_Code/AutoSiteMapProvider.cs file, 1106, 1109
- Web.Config file, 1110

AutoSync property, 953

AutoUpgrade parameter, 232

Average( ) method, 951

AxisY axis, 700

B

BackColor property, 61, 704

BackGradientStyle property, 704

Background parameter, 232

backgrounds, Chart Control, 704, 706, 708

BackSecondaryColor property, 704

Bagus.aspx page, 1421-1422

bar charts, 694

base classes, 1575-1576

- custom LINQ to SQL, 991-995, 999-1000

BaseValidator class, 167

batch updates, performing with SqlDataAdapter object, 892-893, 896

BeginExecuteNonQuery( ) method, 912

BeginExecuteReader( ) method, 912

BeginExecuteXmlReader( )method, 912

BeginGetMovies( ) method, 912

beginHandler parameter, 916

BeginProcessRequest( ) method, 1479-1480

BeginRequest event, 1129

BeginRequestEventArgs, 1703

BetterProfileProvider, 1327, 1329

- BetterProfileProviderReport.aspx page, 1332
- ShowBetterProfileProvider.aspx page, 1329
- web.config file, 1328

BetterProfileProviderReport.aspx page, 1332

Bin folder, adding assemblies to, 753

Binary value, 1314

BinaryWrite( ) method, 189

Bindable design-time attribute, 1632

binding, 488. *See also* DataBound controls; DataSource controls

- controls, 348, 351, 353
- declarative databinding, 356
- DivView control to SqlDataSource control, 1670-1671
- List controls, 341
  - declarative databinding, 442-443
  - programmatic databinding, 443, 446
- Menu controls
  - to database data, 1035, 1040
  - to Site Maps, 1027, 1029, 1031
  - to XML files, 1031-1034
- ObjectDataSource control
  - to components, 772, 774
  - to DataReaders, 774-776
  - to DataSets, 776-778
  - to Web Services, 779-780, 783
- programmatic databinding, 356, 358
- simplified databinding, supporting in templated controls, 1656-1660
- TreeView controls
  - to database data, 1061, 1064
  - to Site Maps, 1057-1058
  - to XML files, 1058-1059, 1061
- two-way databinding, supporting in templated controls, 1660, 1663, 1666

binding containers, 1652

BLL/Product.cs file, 762, 765

Blog.aspx page, 1141

BodyFileName property (MailDefinition class), 1167

BodyTitle property, 259

BorderColor property (Label control), 61

BorderSkin-SkinStyle property, 704

BorderStyle property (Label control), 61

BorderWidth property (Label control), 61

BoundField class, 522-524, 526

BoundField fields, 571

BoundFields display, 571

BoundGridView.aspx file, 352-353

box plots, 694

breakpoints

- configuring, 52
- settings, 51

Browsable design-time attribute, 1632

browserCaps configuration section, 1516

browsers

- cookies. *See* cookies
- varying output cache by, 1346-1347

btnAdd_Click( ) method, 458, 1318

btnRemove_Click( ) method, 458

btnSearch_Click( ) method, 955

btnSubmit_Click( ) handler, 25, 28

BuffersReceived statistics, 854

BuffersSent statistics, 854

Build Deployment Package option, 1567

building pages with QueryExtender controls, 717, 719

BuildProviders, 1454

- DataBuildProvider, 1458, 1461-1462, 1464
    - App_Code\Movie.data, 1458
    - DataBuildProvider.vb, 1463
    - movie.data.72cecc2a.vb, 1459
    - ShowDataBuildProvider.aspx, 1462
- SimpleBuildProvider, 1454, 1456-1458
    - CodeDom, 1455
    - CodeSnippetCompileUnit, 1455
    - implementing on pages, 1457-1458
    - Mike class, 1456-1457
    - registering, 1456
    - SimpleBuildProvider.cs file, 1454
    - Web.Config file, 1456

BuildSiteMap( ) method, 1106, 1109, 1114

BuildTree( ) method, 911

BulletedList control, 101, 338, 470-474

- BulletStyle property, 471
- DisplayMode property, 473
- ShowBulletedList.aspx page, 470-471
- ShowBulletedListHyperLinks.aspx page, 474
- ShowBulletedListImage.aspx page, 472-473

BulletStyle property (BulletedList control), 471

Business Logic layer, 761, 765

Button control, 84, 86, 101, 283, 454, 1158, 1614

- Click event, 86
    - handling, 22, 25-28
- client scripts, 94-95
- Command event, 86, 101, 103
- cross-page posts, 96, 98, 100
- displaying on pages, 84
- events, 86
- Focus( ) method, 86
- properties, 85

ButtonCommand.aspx page, 101

ButtonCounters.aspx page, 25-26

ButtonDefaultButton.aspx page, 100-101

ButtonExpando.aspx page, 95

ButtonField class, 522, 530, 533

ButtonField fields, 571

ButtonOnClientClick.aspx page, 94

buttons

- Button control, 84, 86
    - Click event, 86
    - client scripts, 94-95
    - Command event, 86, 101, 103
    - cross-page posts, 96, 98, 100
    - displaying on pages, 84
    - events, 86
    - Focus( ) method, 86
    - properties, 85
- default buttons, specifying, 100-101
- ImageButton control, 89-93
    - Click event, 93
    - client scripts, 94-95

    - Command event, 93, 101, 103
    - cross-page posts, 96, 98, 100
    - displaying on pages, 89-90
    - Focus( ) method, 93
    - image maps, 91-92
    - properties, 93
  - LinkButton control, 86, 88-89
    - Click event, 89
    - client scripts, 94-95
    - Command event, 89, 101, 103
    - cross-page posts, 96, 98, 100
    - displaying on pages, 86-87
    - events, 89
    - Focus( ) method, 89
    - properties, 88-89
  - RadioButton control, 79, 82-83
    - automatic postback, 82-83
    - CheckedChanged event, 82
    - displaying on pages, 79
    - Focus( ) method, 82
    - properties, 81

- **ButtonSearch.aspx page, 96**
- **ButtonSearchResults.aspx page, 97**
- **ButtonSearchResultsTyped.aspx page, 99**
- **ButtonSearchTyped.aspx page, 98**
- **ButtonType property**
  - ButtonField, 533
  - CommandField, 529
- **By CSS Class element, 1774**
- **By Element element, 1774**
- **By ID element, 1774**
- **BytesReceived statistics, 854**
- **BytesSent statistics, 854**

C

- **C#, new features, 936, 938-939**
  - anonymous types, 940
  - automatic properties, 936-937
  - extension methods, 948-949
  - generics, 941-944
  - initializers, 937-939
  - lambda expressions, 944-947
  - LINQ, 949-951
  - type inference, 939-940
- **cache. *See also* caching**
  - caching database data with SqlDataSource control, 435, 437
  - GAC (Global Assembly Cache), 15
    - adding assemblies to, 754-755, 757
  - SQL cache
    - configuring/enabling, 1114
    - dependencies, 1399-1400
- **cache element, 1397**
- **Cache object, 437, 1387, 1389**
- **Cache-Control header, 1349**
- **Cache-Control HTTP header, 1351**
- **Cached property (ControlCachePolicy class), 1370**
- **CacheDependency class, 1394**
- **CacheDuration property, 1377**
- **CacheItemPriority enumeration, 1396**
- **CacheLocation.aspx page, 1350-1351**
- **CachePageOutput.aspx page, 1337**
- **CacheProfile attribute (%@ OutputCache % directive), 1361**
- **cacheRolesInCookie attribute (web.config file), 1257**
- **CacheSqlDataSource.aspx page, 435**
- **caching, 1335-1336. *See also* cache**
  - Data Caching, 1336, 1387
    - absolute expiration policy, 1390, 1392
    - adding items to cache, 1390
    - cache configuration, 1397
    - cache item priorities, 1396
    - Cache object, 1387, 1389
    - dependencies, 1394-1396
    - Polling SQL cache dependencies, 1406-1407
    - Push SQL cache dependencies, 1415-1416
    - sliding expiration policy, 1393-1394

DataSource Caching, 1336, 1375-1376
- absolute cache expiration policy, 1376-1377
- DataSource control key dependencies, 1383, 1385
- ObjectDataSource control, 1380, 1382
- Polling SQL cache dependencies, 1405-1406
- Push SQL cache dependencies, 1412, 1414
- sliding cache expiration policy, 1378-1379
- XmlDataSource control, 1382-1383

Page Output Caching, 1336-1337
- %@ OutputCache % directive, 1337
- CachePageOutput.aspx sample page, 1337
- expiring programmatically, 1353-1357
- file dependencies, 1351, 1353
- HttpCachePolicy class, 1358-1360
- Polling SQL cache dependencies, 1403-1404
- profiles, 1360-1361
- Push SQL cache dependencies, 1411-1412
- specifying cache location, 1349, 1351
- varying output cache by browser, 1346-1347
- varying output cache by control, 1342, 1344
- varying output cache by custom functions, 1347, 1349
- varying output cache by header, 1345-1346
- varying output cache by parameter, 1338-1342

Partial Page Caching, 1336, 1361-1362
- caching dynamically loaded User Controls, 1373, 1375
- ControlCachePolicy class, 1370-1371
- file dependencies, 1372
- post-cache substitution, 1362-1366
- sharing User Control output cache, 1370
- User Controls, 1366-1369

Polling SQL cache dependencies, 1399-1400
- configuring applications for, 1402-1403
- configuring databases for, 1400-1402
- Data Caching, 1406-1407
- DataSource Caching, 1405-1406
- Page Output Caching, 1403-1404

Push SQL cache dependencies, 1408-1409
- configuring applications for, 1410
- configuring databases for, 1409-1410
- Data Caching, 1415-1416
- DataSource Caching, 1412, 1414
- Page Output Caching, 1411-1412

records, 970

caching configuration section, 1516

Calendar control, 1737
- calendars, displaying, 197
- culture settings, 1433-1434
- displaying calendars with, 195-196
- multiple dates, selecting, 198-199
- pop-up calendars, creating, 199
- properties, 197-199
- rendering calendars from databases, 202, 206-207

Calendar.skin file, 279

CalendarDatabase.aspx page, 203, 206

CalendarJS.aspx page, 200

calendars
- displaying, 195-197
- events, 197
- multiple dates, selecting, 198-199
- pop-up calendars, creating, 199
- properties, 197-199
- rendering from databases, 202, 206-207

CallbackValidator, 1548

calling
- stored procedures, 933
- Web services, 1776-1781

CanBeNull property, 953

CancelButtonClick event, 227

CancelCommand.aspx page, 404-405

CancelDestinationPageUrl property (Wizard control), 226

CancelImageUrl property (CommandField), 529

canceling command execution, 403, 406

CancelText property (CommandField), 530

candlestick charts, 694

Cascading Style Sheets. *See* CSS (Cascading Style Sheets)

CascadingDropDown control, 1737

CascadingDropDownsAjax.aspx page, 1681

CascadingDropDownsNoAjax.aspx page, 1679

Case Animation, 1763

catching roles in cookies, 1256-1258

Category design-time attribute, 1632

CategoryId property, 960

CausesValidation property
- ButtonField, 533
- CommandField, 530

CC property (MailDefinition class), 1167

Cells property (GridViewRow object), 549

changefreq element, 1117

ChangePassword control, 1186-1192
- emails, sending, 1188, 1190
- ShowChangePassword.aspx page, 1186, 1188
- templates, 1190, 1192

ChangePassword( ) method, 1231

ChangePassword.txt file, 1189

ChangePasswordEmail.aspx page, 1189

ChangePasswordQuestionAndAnswer( ) method, 1231

ChangePasswordTemplate.aspx page, 1191-1192

Chart control, 689-690
- background and plotting areas, 704-708
- displaying data with, 690-694
- drill-down reports, 709-710
- filtering, 694-698
- sorting, 694-698
- statistical formulas, 698, 700-701, 704
- three-dimensional charts, 708

Chart functionality, 692

ChartAppearance.aspx page, 705

charts
- areas, 690
- lines, viewing, 692
- three-dimensional, 708
- types of, 694

check boxes, TreeView control, 1054, 1057

CheckBox control, 76-77, 79
- automatic postback, 78-79
- CheckedChanged event, 78
- displaying in forms, 76-77
- Focus( ) method, 78
- properties list, 78

CheckBoxAutoPostBack.aspx page, 78-79

CheckBoxField class, 522, 526-527

CheckBoxField fields, 571

CheckBoxList control, 79, 338, 467-470

Checked property
- CheckBox control, 78
- RadioButton control, 81
- TreeNode object, 1073

CheckedChanged event, 78, 82

CheckFileType( ) method, 184

Checkout.aspx file, 315, 317

checkPhysicalUrlAccess parameter, 1146

ChildNodes property (SiteMapNode class), 1093

Children property, 1134

ChildrenAsTriggers property, 1685

chtMovies_DataBound method, 694

CircleHotSpot class, 106

classes. *See* components; *names of specific classes*

Clear( ) method, 1284

ClearAllPools( ) method, 857

ClearPool( ) method, 857

ClearSelection( ) method, 458

Click event, 91
- Button control, 86
- handling, 22-28
- ImageButton control, 93
- ImageMap control, 111
- LinkButton control, 89

Client value, 1349

client-side Ajax, 1675

client-side validation functions, associating with CustomValidator control, 156, 158

ClientLifecycle.aspx page, 1701

clients, calling Web services, 1776-1781

ClientSubmit property (PostBackOptions class), 1614

clientTarget configuration section, 1516

ClientValidationFunction property (CustomValidator control), 154

Clone( ) method, 1093

CloseConnection parameter (CommandBehavior object), 880

CLR (Common Language Runtime), 17-18

CLR integration, 918

CMS (Content Management System), 1123

code declaration blocks, 8

code-behind pages, 39, 41-43

- FirstPageCodeBehind.aspx page, 40-41
- FirstPageCodeBehind.aspx.cs file, 41
- partial classes, 42
- presentation pages, 41-43
- when to use, 43

CodeConfirmation.htm file, 1175

CodeDom, 1455

CodeSnippetCompileUnit, 1455

CollapseImageToolTip property (TreeView control), 1070

CollapseImageUrl property (TreeView control), 1070

CollapsiblePanel control, 1737

collections

- configuration element collections, creating, 1551-1555
  - AdminUsersSection class, 1553
  - ShoppingCartSection class, 1551-1552
  - ShowAdminUsersSection.aspx page, 1556
  - web.config file, 1552, 1554
- control property collections, 1618
  - AddParsedSubObject( ) method, 1626
  - ContentRotator control example, 1618, 1620-1621
  - ControlBuilder class, 1626, 1629, 1631
  - ImageRotator control example, 1623, 1625
  - ItemRotator control example, 1621
  - ParseChildren attribute, 1618-1619, 1622, 1624-1625
  - ShowItemRotator.aspx page, 1622-1623
- DeleteParameters, 783
- FilterParameters, 783
- initializers, 941
- InsertParameters, 783
- SelectParameters, 783
- UpdateParameters, 783

Color Animation, 1764

ColorEditor, 1632

ColorPicker control, 1737

colors, rendering table of HTML colors, 1581-1582

ColorTable control, 1583

ColorTable.cs file, 1581-1582

column summaries, displaying (GridView control), 550, 552

columns, selecting, 965

Columns property (TextBox control), 70

Combine( ) method, 1135

CombineVirtualPaths( ) method, 1133

ComboBox control, 1737

Command class, 844

Command event, 101, 103

- Button control, 86
- ImageButton control, 93
- LinkButton control, 89

Command object. ***See*** **SqlCommand object**

CommandArgument property

- Button control, 85
- ImageButton control, 93
- LinkButton control, 88
- PagerTemplate, 611

CommandField class, 522, 527, 529-530

CommandField fields, 571

CommandName property, 622, 672, 1158

- Button control, 85
- ButtonField, 533

- ImageButton control, 93
- LinkButton control, 88
- PagerTemplate, 611

CommandRowStyle property (DetailsView control), 601

commands, executing. *See also names of specific commands*

- database commands
  - canceling command execution, 403, 406
  - data source mode, changing, 397-398
  - database rows, filtering, 395-397
  - error handling, 398-403
  - GridViewHandleError.aspx page, 401-402
  - HandleError.aspx page, 399-400
  - inline SQL statements, 389, 391-392
  - stored procedures, 392, 395
- SqlDataSource commands, 426
  - ADO.NET parameters, 426, 428
  - Delete command, 428
  - Insert command, 428, 430
  - Select command, 428-435
  - Update command, 428

Comment property (MembershipUser class), 1231

comments, adding, 745

Common Language Runtime. *See* CLR (Common Language Runtime)

Compare( ) method, 1437

CompareAllValues, 828

CompareValidator control, 121, 146-151

- ControlToCompare property, 146
- ControlToValidate property, 146
- data type checks, performing, 146
- Display property, 127-128
- form fields
  - comparing against another form field, 150
  - comparing against fixed values, 148
- implementing on OrderForm page, 122, 124-125
- IsValid property, 126-127
- Operator property, 146
- Text property, 146
- Type property, 146
- ValueToCompare property, 146

comparing string values, 1437-1438

compilation

- component libraries, 750
  - vbc command-line compiler, 750-751
  - Visual Basic .NET Express, 751
  - Visual Studio .NET 2005, 752
- dynamic compilation, 34-35, 37, 724-725

compilation configuration section, 1516

compilation element (web.config file), 1520

CompilationMode attribute, 35

complex profile properties, creating, 1310-1311, 1314-1316

component libraries, 749

- compiling
  - with vbc command-line compiler, 750-751
  - with Visual Basic .NET Express, 751
  - with Visual Studio .NET 2005, 752
- referencing assemblies
  - adding to Bin folder, 753
  - adding to Global Assembly Cache, 754-755, 757

components, 721

- access modifiers, 744
- accessing Profiles from, 1321-1322
- AggregateCacheDependency, 1394
- ASP.NET intrinsic objects in, 746-749
- assemblies, 14, 17
  - adding to Bin folder, 753
  - adding to Global Assembly Cache, 754-755, 757
  - definition of, 724
  - private, 15
  - referencing, 16-17
  - shared, 15
- AuthenticateEventArgs, 1161
- AutoComplete, 72, 74
- BaseValidator, 167
- binding to ObjectDataSource control, 772, 774
- BuildProviders, 1454-1464

CacheDependency, 1394
CircleHotSpot, 106
component libraries, 749-757
Configuration, 1536-1537
ConflictedMovies, 830-831
constructors, 733-736
Control, 1576
ControlBuilder, 1626, 1629-1631
ControlCachePolicy, 1370-1371
creating, 722, 724
CultureInfo, 1434-1438
custom components, inheriting profiles from, 1308-1309
data access. *See* data access components
DBMovie, 920
definition of, 722
different language components, mixing in App_Code folder, 725-726
DpapiProtectedConfigurationProvider, 1557
dynamic compilation, 724-725
EmployeeData, 791-793
EmployeesDSSorting, 810
ExpressionBuilders, 1464-1469
FieldHelloWorld, 728-729
fields
- declaring, 728-731
- shared fields, 733

File, 10
FilterMovies, 815-816
FormsAuthentication, 1216-1220
Framework Class Library. See Framework Class Library
Graphics, 10
Guestbook, 820
HelloWorld, 722, 724
HtmlTextWriter, 1580-1584
HTTP Handlers, 1470
- anonymous HTTP Handlers, 1478-1481
- Generic Handlers, creating, 1470-1473
- IHttpHandler interface, 1475-1476
- IHttpHandler interface, implementing, 1473

HttpApplication, 1482-1483
- custom HTTP Modules, creating, 1487-1489
- Global.asax files, 1483, 1485

HttpCachePolicy, 1358-1360
HttpCookie, 1272-1273
HttpPostedFile, 180
HttpSessionState, 1284-1285
inheritance, 739-741
InsertMovie, 824-825
Intellisense, 744, 746
interfaces, declaring, 742-743
MailDefinition, 1167
Membership, 1227-1230
MembershipProvider, 741
MembershipUser, 1230-1231
methods
- declaring, 726-728
- overloading, 734-736
- shared methods, 726, 728
- signatures, 735

Movie, 1382
Movie4, 865-868
Movie5, 870-872
MovieCollection, 772-773
MovieDataReader, 774-775
MovieDataSet, 776-778
Movies, 783, 785-786
MoviesByCategory, 826-827
MoviesDSPaging, 800-802
MustInherit classes, 739-741
namespaces, declaring, 736-738
PagePropertyParameter, 838-839, 841
partial classes, 42, 738
PolygonHotSpot, 106
PostBackOptions, 1613, 1616
Preferences, 747, 749
ProductConstructor, 736
ProfileManager, 1323, 1325
properties
- declaring, 728-731
- shared properties, 733

PropertyHelloWorld, 730-731
Quote, 734
Random, 10
RandomDataLayer, 931-933
RandomRows, 929
RectangleHotSpot, 106
Roles, 1258
RsaProtectedConfigurationProvider, 1557, 1562, 1564
ServerTime, 731
ServerValidateEventArgs, 155
SharedHelloWorld, 726-728
SiteMap, 1090-1093
SiteMapNode, 1093-1095
SmtpClient, 10, 12
SqlCacheDependency, 1394
SqlCacheDependencyAdmin, 1400
SqlConnectionStringBuilder, 848
SqlPipe, 928
User, 1220-1221
UsernameParameter, 835-837
viewing in Framework Class Library, 10
VirtualPathProvider, 1132-1135
WebConfigurationManager, 1523-1524
WebControl, 1576, 1584, 1586-1587

composite controls, building, 1587-1589
implementing on pages, 1588
RequiredTextBox.cs control, 1587

CompositeControl class, 1576

concatenating strings, 863

concurrency issues
data updates, 591-593
GridView control, 512-514
ObjectDataSource control, 828, 832

Concurrency.aspx page, 512-513, 591-592

Condition Animation, 1763

Condition( ) method, 984

\CONFIG folder, 1512

configuration, 1511. *See also* Configuration API
ActiveDirectoryMembershipProvider, 1239, 1242-1244
ADAM (Active Directory Application Mode), 1240, 1242
authorization, 1221, 1223
AuthorizationStoreRoleProvider, 1254-1256
of breakpoints, 52
Data Caching, 1397
encrypted configuration sections, 1556-1557
creating programmatically, 1558, 1561
creating with aspnet_regiis command, 1557-1558
deploying, 1561-1564
DpapiProtectedConfigurationProvider, 1557
RsaProtectedConfigurationProvider, 1557
Forms authentication, 1207-1208
Profile providers, 1326-1327
Session state, 1291
Session state stores, 1290-1291
Site Maps
custom attributes, 1103-1105
merging, 1101-1102
Security Trimming, 1096-1098, 1100
SQL Server Session state, 1294-1297
SqlMembershipProvider, 1237-1239
SqlRoleProvider, 1247-1248, 1250-1252
State Server Session state, 1291-1294
web configuration files, storing connection strings in, 386-387
website configuration, 1511, 1513. *See also* Web.config file
ASP.NET configuration sections, 1515-1517
configuration settings, applying to particular paths, 1517-1518
configuration settings, locking, 1518, 1520
configuration settings, placing in external files, 1522-1523
custom application settings, 1520, 1522
Machine.config file, 1511
Microsoft Management Console (MMC) Snap-In, 1514
Web Site Administration Tool, 1513-1514
Web.config file, 1512
WindowsTokenRoleProvider, 1252-1254

Configuration API, 1523-1524
- Configuration class, 1536-1537
- configuration element collections, creating, 1551-1555
  - AdminUsersSection class, 1553
  - ShoppingCartSection class, 1551-1552
  - ShowAdminUsersSection.aspx page, 1556
  - web.config file, 1552, 1554
- configuration files, opening
  - on remote servers, 1533-1534, 1536
  - on same server, 1529-1530, 1533
- configuration sections, modifying, 1539, 1541-1542
- configuration sections, reading from current application, 1524-1525, 1527
  - ShowConfigApp.aspx page, 1524-1525
  - ShowConfigPath.aspx page, 1528
  - ShowConfigRelative.aspx page, 1525, 1527
- custom configuration sections, creating, 1546, 1548-1549
  - ConfigurationProperty attribute, 1548
  - validators, 1548
  - web.config registration, 1549
- new websites, provisioning, 1543, 1546
- WebConfigurationManager class, 1523-1524

Configuration class, 1536-1537

configuration element collections, creating, 1551-1555
- AdminUsersSection class, 1553
- ShoppingCartSection class, 1551-1552
- ShowAdminUsersSection.aspx page, 1556
- web.config file, 1552, 1554

configuration files, 1511, 1513. *See also* Machine.config file; web.config file
- opening on remote servers, 1533-1536
- opening on same server, 1529-1530, 1533
  - ShowConfigMachine.aspx page, 1529-1530
  - ShowConfigSites.aspx page, 1531

ConfigurationProperty attribute, 1548-1549, 1554

ConfigurationSaveMode parameter (Save( ) method), 1541

confirm( ) method, 622

confirmation codes, entering, 1175

ConfirmButton control, 1737

ConfirmCode.aspx page, 1176

ConfirmPassword ID, 1180, 1192

ConflictDetection property, 591
- ObjectDataSource control, 828
- SqlDataSource control, 513

ConflictedMovies component, 830-831

ConflictedMovies.cs file, 830-831

connected data access. *See* data access components

connecting to databases, SqlDataSource control, 380
- connection strings, encrypting, 388-389
- connection strings, storing in Web.Config file, 386-387
- ODBC, 384
- OLE DB, 384
- Oracle, 384, 386
- SQL Server, 380, 382-384

Connection class, 844

Connection object. *See* SqlConnection object

Connection Reset, 857

connection strings
- encrypting, 388-389
- security, 382
- storing in Web.Config file, 386-387

Connection Timeout, 857

connectionPassword( ) method, 1243

connectionProtection attribute, 1243

connectionProtection( ) method, 1243

connections
- Ajax
  - asynchronous postbacks, 1707-1714
  - debugging, 1675-1677
  - execution life cycles, 1697-1707
  - handling errors, 1714-1717, 1720
  - JavaScript, 1694-1697
  - overview of, 1674-1675

performance, 1720, 1723-1724
server-side versus client-side, 1675
Timer controls, 1724-1728
UpdatePanel control, 1677-1682, 1685-1688, 1692-1693
UpdateProgress control, 1728-1731
WCF Data Services, 1001
data context, 1007-1009
service references, 1002-1007

connectionStringName attribute, 1243

connectionStringName( ) method, 1243

ConnectionStrings property
Configuration class, 1536
WebConfigurationManager class, 1523

ConnectionStringsExpressionBuilder class, 1465

connectionUsername( ) method, 1243

ConnectOracle.aspx page, 385

Constant( ) method, 984

constrained routes, 1144-1145

constructors
declaring, 733-734
overloading, 734-736
shared constructors, 734

container ControlDesigners, creating, 1636, 1639-1640

Container keyword, 1652

containers
binding containers, 1652
key containers
creating, 1562
exporting contents to XML files, 1563
granting access to, 1563
web.config file, 1562

Content control, 242

Content folder, 1504

Content Management System. *See* CMS (Content Management System)

ContentLength property (FileUpload control), 180

ContentPlaceHolder controls, 239-240, 242

ContentRotator control
ContentRotator.cs file, 1618
ShowContentRotator.aspx page, 1620-1621

ContentRotator.cs file, 1618

ContentTemplateContainer property, 1686

ContentType property (FileUpload control), 180

context connections, 928

Control class, 1576

Control State, 33
supporting in custom controls, 1599
ControlStateControl.vb, 1599
ShowControlState.aspx page, 1601

ControlBuilder class, 1626, 1629, 1631
methods, 1626
ServerTabs control example, 1626, 1629-1631

ControlCachePolicy class, 1370-1371

ControlDesigners, 1636
container ControlDesigners, creating, 1636, 1639-1640
Smart Tasks, 1640, 1644

ControlFilterExpression, querying, 712

ControlID property (ControlParameter object), 410

controllers
access, 1506
MVC Framework, 1502

Controllers folder, 1504

ControlParameter object, 353, 410-412, 414
ObjectDataSource control, 787
properties, 410
ShowControlParameter.aspx page, 410-411
ShowPageControlParameter.aspx page, 413-414

controls, 19. *See also* events
AdRotator, 207-219, 1336
AdvancedCheckBox, 1614-1616
AJAX Control Toolkit, 1737-1739
Animation, 1759-1764
AutoComplete, 1739-1750
DragPanel, 1750-1751, 1754
FilteredTextBox, 1754-1756
MaskedEdit, 1756-1759
UpdatePanelAnimation, 1764-1768
AjaxValidator, 171-174

Article, 1648-1650
ArticleWithDefault, 1652, 1655
binding, 348, 351
Button, 84-86, 1614
  Click events, handling, 22, 25-28
  client scripts, 94-95
  Command event, 86, 101, 103
  cross-page posts, 96-100
Calendar, 195-199
  culture settings, 1433-1434
  rendering calendars from databases, 202, 206-207
ChangePassword, 1186-1192
Chart Control. *See* Chart Control
CheckBox, 76-79
CheckBoxList, 79
ColorTable, 1583
CompareValidator, 121
ContentPlaceHolder, 240, 242
ContentRotator, 1618-1621
Control State, 33
control trees, 38-39
ControlDesigners, 1636
  container ControlDesigners, creating, 1636, 1639-1640
  Smart Tasks, 1640, 1644
CreateUserWizard, 1162
  adding steps to, 1181-1183
  automatically generating passwords, 1171-1174, 1177
  configuring form fields, 1163-1166
  properties, 1162-1163
  redirecting users, 1169-1170
  sending email messages, 1166-1169
  templates, 1177-1181
culture settings, 1433-1434
custom controls, 1576
CustomLinkButton, 1606-1610
CustomTextBox, 1602-1605
CustomValidator, 122
DataBound. *See* DataBound controls
DataGrid, 342, 485
DataList. *See* DataList controls
DataPager, 681-688
DataSource. *See* DataSource controls
declaring standard controls, 20
design-time attributes, 1631-1635
DetailsView. *See* DetailsView controls
DivView, 1667-1671
DropDownList, setting cultures, 1423, 1425
EntityDataSource, 994
FileUpload, 180
  saving files to database, 185, 188-190, 193
  saving files to filesystem, 181, 184
  uploading large files, 189-195
FormView, 604, 977
  deleting data with, 619, 621-622
  displaying data with, 605-606
  editing data with, 612-616
  inserting data with, 616-619
  paging through data with, 607-609, 612
GradientPanel, 1637-1639
GridView. *See* GridView controls
HTML controls, declaring, 21-22
HtmlMeta, 258
HyperLink, 118-119
Image, 104, 106
ImageButton. *See* ImageButton controls
ImageMap, 106-111
ImageRotator, 1623-1625
ItemRotator, 1621-1623
Label, 60-65
LengthValidator, 168-169
LinkButton. *See* LinkButton controls
LinqDataSource, 960, 962-963
List. *See* List controls
ListView. *See* ListView controls
Literal, 65-68
Localize, 1450-1451
Login. *See* Login controls
LoginName, 1185-1186
LoginStatus, 1183, 1185
LoginView, 1199-1203
Menu. *See* Menu controls
modifying in Master Pages, 261-263

MovieDataSource, 832-833, 835
MovieView, 1633-1635
MultiView. *See* MultiView controls
navigation, 1016
NewsRotator, 1364
ObjectDataSource. *See* ObjectDataSource controls
Pager, 1610-1613
Panel, 101, 111-118, 1347
Partial Page Caching, 1366-1367, 1369
PasswordRecovery, 1193-1199
Product, 1657-1659
ProductForm, 1661, 1663
QueryExtender, 711-719
RadioButton, 79-83, 111
RadioButtonList, 83
RangeValidator, 121
RegularExpressionValidator, 122
Repeater, 623-633, 640
ServerTabs, 1626, 1629-1631
SiteMapDataSource, 1084
binding Menu controls to, 1087-1088
binding TreeView controls to, 1085-1086
Default.aspx page, 1085-1086
Products/Default.aspx page, 1089
properties, 1086-1090
Services/Default.aspx page, 1087
Web.sitemap file, 1084-1085
SiteMapPath, 1013
declaring in pages, 1013, 1015
formatting, 1016, 1018-1020
properties, 1015-1016
templates, 1018-1020
SmartImage, 1640, 1644
SqlDataSource. *See* SqlDataSource control
SqlHierarchicalDataSource, 1011, 1077
classes, 1080-1081
ShowMenu.aspx page, 1078-1079
ShowTreeView.aspx page, 1079
Substitution, 1362-1363
templated controls, 1647-1648
default templates, creating, 1652-1655
ITemplate interface, implementing, 1648-1652
simplified databinding, supporting, 1656-1660
two-way databinding, supporting, 1660, 1663, 1666
templated databound controls, 1666
base classes, 1667
DivView, 1667, 1670-1671
TextBox. *See* TextBox controls
Timer, Ajax, 1724-1726, 1728
TreeView. *See* TreeView controls
UpdatePanel. *See* UpdatePanel controls
UpdateProgress, Ajax, 1728-1729, 1731
User. *See* User controls
validation controls. *See* validation controls
ValidationSummary, 122, 162, 166
Display property, 127-128
implementing on OrderForm page, 122, 124-125
IsValid property, 126-127
varying output cache by, 1342, 1344
View, 220
View State, 28-33
Wizard, 226-228, 231
XmlDataSource, DataSource caching, 1382-1383

ControlStateControl.cs file, 1599

ControlToCompare property (CompareValidator control), 146

ControlToValidate property
CompareValidator control, 146
CustomValidator control, 154
RangeValidator control, 144
RequiredFieldValidator control, 139

Convert( ) method, 984

ConvertEmptyStringToNull property
BoundField, 525
Parameter object, 406

COOKIE header, 1210, 1264

CookieDomain property (FormsAuthentication class), 1216

cookieless attribute (sessionState element), 1289

cookieless configuration option (Forms authentication), 1207

cookieless Forms authentication, 1209-1210

cookieless Session state, 1288-1290

CookieMode property
- FormsAuthentication class, 1216
- HttpSessionState class, 1284

cookieName attribute, 1257

CookieName property (CookieParameter object), 415

CookieParameter object, 353, 415, 417
- ObjectDataSource control, 787
- properties, 415
- Vote.aspx page, 415

cookiePath attribute, 1257

cookieProtection attribute, 1257

cookieRequireSSL attribute, 1257

cookies, 1264-1265
- catching roles in, 1256-1258
- creating, 1266, 1268-1269
- deleting, 1273-1275
- HttpCookie class, 1272-1273
- multi-valued cookies, 1275, 1278-1279
  - creating, 1275
  - reading, 1278-1279
- persistent cookies, 1264, 1268-1269
- properties, 1272-1273
- reading, 1269-1270, 1272
- security restrictions, 1265-1266
- session cookies, 1264, 1266, 1268

cookieSlidingExpiration attribute, 1257

CookiesSupported property (FormsAuthentication class), 1216

cookieTimeout attribute, 1257

copying
- encrypted configuration files, 1561
- URLs, 1137

Count property
- Cache object, 1387
- HttpSessionState class, 1284

Count( ) method, 951

counting user sessions, 1287

CountMoviesInCategory stored procedure, 392

Create, 1492

Create Deployment Package as a ZIP File option, 1567

CREATE PROCEDURE command, 931

Create Project dialog box, 1502

CREATE TYPE command, 924

Create( ) method, 943

CreateChildControls( ) method, 1587, 1590, 1650, 1654, 1659, 1669

createPersistentCookie attribute, 1257

CreateQuery( ) method, 1004

CreateRole( ) method, 1258

CreateSpecificCulture( ) method, 1434

CreateUser( ) method, 1227, 1245

CreateUserWizard control, 1151, 1162, 1165-1166
- adding steps to, 1181, 1183
- automatically generating passwords, 1171-1174, 1177
  - CodeConfirmation.htm file, 1175
  - ConfirmCode.aspx page, 1176
  - CreateUserWizardCodeConfirmation.aspx page, 1173
  - CreateUserWizardPasswordConfirmation.aspx page, 1172
  - PasswordConfirmation.htm file, 1173
- configuring form fields, 1163, 1165-1166
- properties, 1162-1163
- redirecting users, 1169-1170
  - CreateUserWizardReturn.aspx page, 1170
  - LoginReturn.aspx page, 1169
- sending email messages, 1166-1169
- templates, 1177, 1179, 1181

CreateUserWizardCodeConfirmation.aspx page, 1173

CreateUserWizardEmail.aspx page, 1166

CreateUserWizardExtra.aspx page, 1181, 1183

CreateUserWizardPasswordConfirmation.aspx page, 1172

CreateUserWizardReturn.aspx page, 1170

CreateUserWizardShort.aspx page, 1165-1166

CreateUserWizardTemplate.aspx page, 1178-1180

CreationDate property (MembershipUser class), 1231

credentials element, 1218

cross-page posts, 96, 98, 100

cryptographically strong keys, generating, 1214

CSS (Cascading Style Sheets), 199

adding to Themes, 281-284

App_Themes\StyleTheme\SimpleStyle.css file, 281-282

multiple Cascading Style Sheets, 285

ShowSimpleCSS.aspx file, 282-283

changing page layouts with, 285-290

absolute layouts, 288, 290

Absolute.css file, 288, 290

Float.css file, 287-288

floating layouts, 287-288

ShowLayout.aspx file, 285, 287

CSS Zen Garden, 290

Internet Explorer extensions, 1584

Login controls, 1154

Master Pages, 240

Themes. *See* Themes

transition filters, 627

CSS Zen Garden, 290

CssClass property

DataList control, 657

DetailsView control, 601

GridView control, 519

Label control, 61

Culture property, 1420-1421

CultureInfo class, 1434

comparing and sorting string values, 1437-1438

formatting string values, 1435-1436

methods, 1434

cultures

Culture property, 1420-1421

CultureInfo class, 1434

comparing and sorting string values, 1437-1438

formatting string values, 1435-1436

methods, 1434

detecting automatically, 1428, 1431-1432

SelectCultureAuto.aspx page, 1431-1432

ShowAcceptLanguages.aspx page, 1430

effect on ASP.NET controls, 1433-1434

global resources

creating, 1446, 1448

retrieving programmatically, 1448-1449

strongly typed localization expressions, 1449-1450

local resources, 1438

explicit localization expressions, 1438-1441

implicit localization expressions, 1441-1443

page properties, 1443-1444

retrieving programmatically, 1444-1445

Localize control, 1450-1451

names, 1420

neutral cultures, 1420

setting in web configuration files, 1432-1433

setting manually, 1421-1423, 1425, 1428

%@ Page % directive, 1421-1422

DropDownList control, 1423, 1425

profiles, 1425, 1427

web configuration files, 1427-1428

specific cultures, 1420

UICulture property, 1420-1421

current URL, determining, 1125

Current value, 901

CurrentNode property (SiteMap class), 1090

CurrentNodeStyle object, 1016

CurrentNodeTemplate, 1018

CurrentPassword ID, 1192

CursorsOpen statistics, 854

custom algorithms, 698

custom attributes, Site Maps, 1103-1105

custom authentication, Login control, 1160-1161

custom components, inheriting profiles from, 1308-1309

custom configuration sections, creating, 1546, 1548-1549
- ConfigurationProperty attribute, 1548
- validators, 1548
- web.config registration, 1549

custom configuration settings, 1520, 1522

custom controls, 1576
- AddressForm control, 311, 315, 317
  - AddressForm.ascx, 311, 315
  - Checkout.ascx example, 315, 317
- AJAX (Asynchronous JavaScript and XML), 317, 320
- ASPNetSurvey control, 326-327
- ASPSurvey control, 325-326
- base classes, 1575-1576
- collections, 1618
  - AddParsedSubObject( ) method, 1626
  - ContentRotator control example, 1618, 1620-1621
  - ControlBuilder class, 1626, 1629, 1631
  - ImageRotator control example, 1623, 1625
  - ItemRotator control example, 1621
  - ParseChildren attribute, 1618-1619, 1622, 1624-1625
  - ShowItemRotator.aspx page, 1622-1623
- composite controls, building, 1587-1589
  - implementing on pages, 1588
  - RequiredTextBox.cs control, 1587
- Control State, supporting, 1599
  - ControlStateControl.vb, 1599
  - ShowControlState.aspx page, 1601
- ControlDesigners, 1636
  - container ControlDesigners, creating, 1636, 1639-1640
  - Smart Tasks, 1640, 1644
- creating, 300-302
- design-time attributes, 1631-1635
- exposing events from, 307, 309, 311
- exposing properties from, 304-306
  - PropertyRandomImage.ascx file, 304-305
  - ShowDeclarative.ascx file, 305
  - ShowProgrammatic.ascx file, 306
- fully rendered controls, building, 1576, 1579-1581, 1584, 1587
  - FullyRenderedControl.cs file, 1577
  - FullyRenderedWebControl.cs file, 1578
  - HtmlTextWriter class, 1580-1584
  - ShowFullyRenderedControl.aspx page, 1577-1578
  - ShowFullyRenderedWebControl.aspx page, 1579-1580
  - WebControl TagKey property, 1584, 1586-1587
- hybrid controls, building, 1590-1595
  - Login.cs file, 1590
  - LoginStandards.cs file, 1593
- loading dynamically, 320-321
  - %@ Reference %> directive, 322, 325, 327
  - LoadControl( ) method, 320
  - multi-page wizards, creating, 327-329, 331, 336
  - ShowFeaturedProduct.aspx example, 320-321
- MultiSelectList, 475
  - MultiSelectList.js, 481-482
  - MultiSelectList.vb, 475, 480
  - ShowMultiSelectList.aspx page, 482, 484
- postback data, handling, 1602, 1604, 1606
  - CustomTextBox example, 160-1605
  - IPostbackDataHandler interface, 1602
  - LoadPostData( ) method, 1602, 1604
  - RaisePostDataChangedEvent( ) method, 1602, 1604
  - RegisterRequiresPostBack( ) method, 1604
- postback events, handling, 1606-1616
  - CustomLinkButton example, 1606, 1608
  - event arguments, passing, 1610, 1613
  - IPostBackEventHandler interface, 1606, 1608
  - postback options, 1613, 1616
  - RaisePostBackEvent( ) method, 1606
- PropertyRandomImage control, 304
- RandomImage example
  - RandomImage.ascx, 300-301
  - ShowRandomImage.ascx, 301-302

RandomQuotation control
RandomQuotation.ascx, 318-319
ShowRandomQuotation.ascx, 319
registering in web configuration file, 303
TabStrip control
ShowTabStrip.ascx, 309, 311
TabClick event handler, 311
TabStrip.ascx, 307, 309
View State, supporting, 1596
ShowViewState.aspx page, 1598
ViewStateControl.cs file, 1596

custom fields, GridView control
DeleteButtonField, 560, 562-563
LongTextField, 556, 559-560
ValidatedField, 563-566

custom functions, varying output cache by, 1347, 1349

custom HTTP Modules, creating, 1487-1489
QueryStringAuthenticationModule.cs file, 1487
Web.Config file, 1488

custom LINQ to SQL base class, 991-995, 999-1000

custom membership providers, creating, 1244-1246

custom ObjectDataSource controls, creating, 832-833, 835

custom paging interface (GridView control), 504, 506, 509
PageGridPreviousNext.aspx page, 505-506
PageTemplateGrid.aspx page, 507, 509

custom parameter objects, creating, 835-837, 839, 841
page property parameters, 838-839, 841
username parameters, 835-837

custom Profile providers, creating, 1327-1331

custom sorting interface (GridView control), 497, 499, 501

custom validation controls, creating, 167
AjaxValidator control, 170, 173-174
AjaxValidator.js file, 173-174
AjaxValidator.cs file, 171
ShowAjaxValidator.aspx page, 174
BaseValidator class, 167
LengthValidator control, 168-170
LengthValidator.cs file, 168
ShowLengthValidator.aspx page, 169

Custom value, 1291

customErrors configuration section, 1516

CustomExpression, querying, 712

Customize Line Images, 1071

customizing
CreateUserWizard control, 1178
HttpApplication class, 1483
interfaces, 684, 686
Login controls, 1158
paging interface (DetailsView control), 580-583
PagerSettings class, 581
ShowPagerSettings.aspx page, 580-581
ShowPagerTemplate.aspx page, 582-583

CustomLinkButton control, 1610
CustomLinkButton.cs file, 1606
ShowCustomLinkButton.aspx page, 1608

CustomLinkButton.cs file, 1606

customProviderData attribute, 1298, 1309

CustomSorting.aspx page, 500-501

CustomTextBox control
CustomTextBox.cs file, 1602, 1604
ShowCustomTextBox.aspx page, 1605

CustomTextBox.cs file, 1602, 1604

CustomValidator control, 122, 148, 154
client-side validation functions, 156, 158
ClientValidationFunction property, 154
ControlToValidate property, 154
Display property, 127-128
empty form fields, validating, 158-159
implementing on OrderForm page, 122, 124-125
IsValid property, 126-127
ServerValidateEventArgs class, 155
string lengths, checking, 154, 156, 158
Text property, 154
timed tests, creating, 160

D

data access
- binding controls, 348, 351
- database-driven web application, 375-376
- DataBound controls, 337
  - hierarchical DataBound controls, 345, 347-348
  - List controls. *See* list controls
  - one-way databinding expressions, 362-364, 366
  - tabular DataBound controls, 341-342, 345
  - templates, 359-360, 362
  - two-way databinding expressions, 366-368
- DataSource controls. *See* DataSource controls
- programmatic databinding, 356, 358
- SQL Server 2005 Express, 369-370
  - connecting to, 372
  - Database Explorer, 370
  - keywords, 373
  - Local databases, 338, 373-374
  - Management Studio Express, 370
  - Server databases, 371-372
  - SQLCMD, 370-371

data access components, 843-845
- AsyncDataLayer, 913-914
- asynchronous ASP.NET pages, 914, 917
- asynchronous database commands, 911-912, 914
- building
  - Movie1.cs file, 845-846
  - ShowMovie1.aspx page, 847-848
- CLR integration, 918
- Command object, 859
  - executing commands with parameters, 863-864
  - executing commands without parameters, 860, 862
  - executing stored procedures, 865-872
  - returning resultsets, 876, 878-881
  - returning single values, 874, 876
- Connection object, 848-849
  - Connection pooling, 856-858
  - provider statistics, retrieving, 849-856
- data access layers, creating with user-defined types, 924, 926-927
- DataAdapter object, 889
  - batch updates, performing, 892-893, 896
  - displaying data with, 890, 892
  - populating DataTables with, 889-890
- DataLayer1
  - DataLayer1.cs file, 882
  - ShowDataLayer1.aspx page, 885
- DataReader object, 881
  - Multiple Active Results Sets (MARS), 888
  - multiple resultsets, returning, 882, 884
  - Read( ) method, 881
- DataSet object, 908, 911
- DataTable object, 896
  - building programmatically, 896, 898
  - DataRow states and versions, 901, 903, 906
  - DataRows, selecting, 898, 901
- DataView object, 906
- DBDataLayer
  - DBDataLayer.cs file, 924-925
  - ShowDBDataLayer.aspx page, 926-927
- Movie1
  - Movie1.cs file, 845-846
  - ShowMovie1.aspx page, 847-848
- Movie2
  - Movie2.cs file, 850-851
  - ShowMovie2.aspx page, 852
- Movie3
  - Movie3.cs file, 860, 862
  - ShowMovie3.aspx, 862
- Movie6
  - Movie6.cs file, 877-878
  - ShowMovie6.aspx page, 878-879
- Movie7
  - Movie7.cs file, 879
  - ShowMovie7.aspx page, 880

Movie8
Movie8.cs file, 890
ShowMovie8.aspx page, 891-892
Movie9
Movie9.cs file, 902
ShowMovie9.aspx page, 904, 906
stored procedures, creating, 928
CREATE PROCEDURE command, 931
executing from ASP.NET pages, 931-932
stored procedure assembly, creating, 928, 930
stored procedure assembly, registering, 930
user-defined types, creating with .NET Framework, 919
CREATE TYPE command, 924
user-defined type assembly, creating, 920, 923
user-defined type assembly, registering, 923

Data Access layer, 765, 769
creating with user-defined types, 924-927

Data Caching, 1336, 1387
absolute expiration policy, 1390, 1392
adding items to cache, 1390
cache configuration, 1397
cache item priorities, 1396
Cache object, 1387, 1389
dependencies, 1394-1396
Polling SQL cache dependencies, 1406-1407
Push SQL cache dependencies, 1415-1416
sliding expiration policy, 1393-1394

data context, applying WCF Data Services with, 1007-1009

data controls, 19

data keys, GridView control, 492-495

data points, 690

data source modes, changing, 397-398

data source paging, 503, 686, 688, 798-805
GetPagedMovies stored procedure, 803
GetPagedMovies2005 stored procedure, 805
MoviesDSPaging component, 800-802
ShowDSPaging.aspx page, 798-799

data source sorting, 808-809, 811-812
EmployeesDSSorting component, 810
GetSortedEmployees stored procedure, 811-812
ShowDSSorting.aspx page, 808-809

data sources, Connection objects, 848-849
Connection pooling, 856, 858
provider statistics, retrieving, 849-856

data types
user-defined types, creating data access layers with, 924, 926-927
user-defined types, creating with .NET Framework, 919
CREATE TYPE command, 924
user-defined type assembly, creating, 920, 923
user-defined type assembly, registering, 923
validating, 146

DataAdapter object, 888-889
batch updates, performing, 892-893, 896
displaying data with, 890, 892
populating DataTables with, 889

DataAlternateTextField property (ImageField), 543

DataAlternateTextFormatString property (ImageField), 543

Database Explorer (SQL Server), 370

database-driven web application, 375-376

databases. *See also* ADO.NET; data types
asynchronous ASP.NET pages, 914, 917
asynchronous database commands, 911-912, 914
binding Menu controls to, 1035, 1040
binding TreeView controls to, 1061, 1064
caching with SqlDataSource control, 435, 437
CLR integration, 918
command execution
canceling, 403, 406
data source mode, changing, 397-398
database rows, filtering, 395-397
error handling, 398, 400, 403
inline SQL statements, 389, 391-392
stored procedures, 392, 395

configuring
- for Polling SQL cache dependencies, 1400-1402
- for Push SQL cache dependencies, 1409-1410

connecting with SqlDataSource control, 380
- connection strings, encrypting, 388-389
- connection strings, storing in Web.Config file, 386-387
- ODBC, 384
- OLE DB, 384
- Oracle, 384, 386
- SQL Server, 380, 382-384

data filtering, 813-816

data source paging, 798-805
- GetPagedMovies stored procedure, 803
- GetPagedMovies2005 stored procedure, 805
- MoviesDSPaging component, 800-802
- ShowDSPaging.aspx page, 798-799

data source sorting, 808-809, 811-812
- EmployeesDSSorting component, 810
- GetSortedEmployees stored procedure, 811-812
- ShowDSSorting.aspx page, 808-809

data, storing in Session state, 1281, 1284

database-driven web application, 375-376

deploying, 1571-1572

dta source connections. *See* SqlConnection object

editing, 675, 678-679, 681

records
- deleting, 860, 862
- updating, 860, 862
- viewing, 664

rendering calendars from, 202, 206-207

rows, filtering, 395-397

saving files to, 185, 188-189
- FileHandler.aspx page, 188-189
- FileUploadDatabase.aspx page, 185, 187
- large files, 190, 193

sorting, 672, 674

SQL Server Express, 1408
- Local databases, 338, 373-374
- Server databases, 371-372

standard commands, 964-990

storing advertisements in, 212-213

user interface paging, 795, 797-798

user interface sorting, 805, 807-808

DataBind( ) method, 358, 1004, 1652, 1670

DataBinder.Eval( ) method, 839, 1005

databinding. *See* binding

DataBinding event, 547, 598, 633

DataBinding event handler, 640

databinding expressions
- one-way, 362-364, 366
- two-way, 366-368

DataBound controls, 337
- binding controls, 348, 351
- binding to DataSource controls, 351, 353
- DataList, 640
  - displaying data with, 640, 642-643
  - editing data with, 651, 656
  - formatting data with, 657, 659
  - multi-column tables, creating, 644-645
  - properties, 657
  - selecting data with, 648-651
  - templates, 645-646, 648
- hierarchical DataBound controls, 345-348
  - Menu, 346
  - Movies.xml page, 347-348
  - ShowHierarchicalDataBound.aspx page, 346-347
  - TreeView, 346
- list controls, 338, 341, 439
  - automatic postbacks, 452, 454
  - binding, 341
  - binding to data sources, 442-443, 446
  - BulletedList, 338, 470-471, 473-474
  - CheckBoxList, 338, 467, 469-470
  - DropDownList, 338, 458, 460
  - EasyListBox, 460
  - Items collection, 454, 458
  - list items, appending, 450, 452
  - list items, declaring, 439, 441

list items, reordering, 454, 458

ListBox, 338, 463, 465, 467

ListControl class, 439

ListItem class, 439, 441

MultiSelectList (custom control), 475, 480-482, 484

RadioButtonList, 338, 461-463

selected list items, determining, 446, 448-450

ShowListControls.aspx page, 339

one-way databinding expressions, 362-366

Repeater control, 623

displaying data with, 624-625, 627

events, 633, 640

templates, 627-633

tabular DataBound controls, 341-342, 345

DataList, 341

DetailsView. *See* DetailsView control

FormView, 342

GridView, 341

Repeater, 342

ShowTabularDataBound.aspx page, 343, 345

templated databound controls, 1666

base classes, 1667

DivView, 1667, 1670-1671

templates, 359-360, 362

two-way databinding expressions, 366-368

DataBound event, 547, 598

DataBuildProvider, 1458, 1461-1462, 1464

App_Code\Movie.data, 1458

DataBuildProvider.vb, 1463

movie.data.72cecc2a.vb, 1459

ShowDataBuildProvider.aspx, 1462

DataBuildProvider.cs file, 1463

DataColumn.Expression property, 397, 817

DataContext.ExecuteCommand( ) method, 983

DataContext.GetCommand( ) method, 990

|DataDirectory| keyword, 374

DataField property (BoundField), 525

DataFormatString property (BoundField), 525

DataGrid control, 342, 485. *See also* GridView control

DataImageUrlField property (ImageField), 543

DataImageUrlFormatString property (ImageField), 543

DataItem property

GridViewRow object, 549

IDataItemContainer interface, 1656

DataItemIndex property

GridViewRow object, 549

IDataItemContainer interface, 1656

DataKeyField property, 651

DataKeyNames property, 492, 512, 598, 979

DataLayer1 component

DataLayer1.cs file, 882

ShowDataLayer1.aspx page, 885

DataLayer1.cs file, 882

DataList control, 341, 640

displaying data with, 640, 642-643

RepeatLayout property, 643

ShowDataList.aspx page, 640, 642

ShowFlowDataList.aspx page, 643

editing data with, 651, 656

formatting data with, 657, 659

multi-column tables, creating, 644-645

properties, 657

selecting data with, 648-651

templates, 645-646, 648

AlternatingItemTemplate, 645

EditItemTemplate, 646

FooterTemplate, 646

HeaderTemplate, 646

ItemTemplate, 645

SelectedItemTemplate, 646

SeparatorTemplate, 645

ShowDataListTemplates.aspx page, 646, 648

DataManipulator property, 694

DataMember property (AdRotator control), 207

DataNavigateUrlFields property (HyperLinkField), 539

DataNavigateUrlFormatString property (HyperLinkField), 539

DataPager control, 681-683
data source paging, 686, 688
interfaces, customizing, 684, 686
DataPagerDataSource.aspx page, 686
DataPagerListView.aspx page, 682
DataPagerTemplate.aspx page, 684
DataReader class, 844
DataReader object, 397, 881
binding to ObjectDataSource control, 774-776
Multiple Active Results Sets (MARS), 888
multiple resultsets, returning, 882, 884
Read( ) method, 881
DataRows
selecting, 898, 901
states and versions, 901, 903, 906
DataSet object, 397, 889, 908, 911, 1281
binding to ObjectDataSource control, 776-778
DataSource Caching, 1336, 1375-1376
absolute cache expiration policy, 1376-1377
DataSource control key dependencies, 1383, 1385
ObjectDataSource control, 1380, 1382
Polling SQL cache dependencies, 1405-1406
Push SQL cache dependencies, 1412, 1414
sliding cache expiration policy, 1378-1379
XmlDataSource control, 1382-1383
DataSource controls, 351
AccessDataSource, 351
ASP.NET parameters, 353, 355
binding DataBound controls to, 351, 353
binding List controls to
declarative databinding, 442-443
programmatic databinding, 443, 446
ObjectDataSource, 351, 771
binding to components, 772, 774
binding to DataReaders, 774-776
binding to DataSets, 776-778
binding to Web Services, 779-780, 783
concurrency, 828, 832
custom ObjectDataSource controls, 832-833, 835
custom parameter objects, creating, 835-837, 839, 841
data filtering, 813-816
data source paging, 798-805
data source sorting, 808-809, 811-812
error handling, 822, 825
events, 817-822, 826-828
parameters, 783, 785, 789-790, 793-795, 818, 820, 822
properties, 772
user interface paging, 795, 797-798
user interface sorting, 805, 807-808
SiteMapDataSource, 351
SqlDataSource. *See* SqlDataSource control
XmlDataSource, 351
DataSource property (AdRotator control), 207
DataSourceAbsoluteCache.aspx page, 1376
DataSourceID property (AdRotator control), 207
DataSourceKeyDependency.aspx page, 1384-1385
DataSourceMode property, 1376
DataSourceSlidingCache.aspx page, 1378-1379
DataTable object, 889, 896
building programmatically, 896, 898
DataRow states and versions, 901, 903, 906
DataRows, selecting, 898, 901
populating with SqlDataAdapter object, 889-890
DataTextField property
ButtonField, 533
HyperLinkField, 539
DataTextFormatString property
ButtonField, 533
HyperLinkField, 539
DataView object, 889, 906, 1282
DataView.RowFilter property, 397, 817
DayNameFormat property (Calendar control), 197
DayRender event, 197, 207

DbCommand class, 845

DbConnection class, 845

DBDataLayer component

DBDataLayer.cs file, 924-925

ShowDBDataLayer.aspx page, 926-927

DBDataLayer.cs file, 924-925

DbDataReader class, 845

DBMovie class, 920

DBMovie.cs file, 920

DbType property, 953

Deactivate event

View control, 220

WizardStep, 227

debugging

Ajax, 1675-1677

ASP.NET pages, 48, 50, 52

checking if enabled, 1524-1525, 1527

disabling, 50

jQuery, 1770

LINQ to SQL, 987-988, 990

declarative databinding, 356, 442-443

DeclarativeDataBinding.aspx page, 442-443

declaring

constructors, 733-734

controls

HTML controls, 21-22

standard controls, 20

fields, 728-731

interfaces, 742-743

methods, 726-728

namespaces, 736-738

partial classes, 738

properties, 728-731

Decrypt( ) method, 1217

decryptionKey attribute, 1211, 1294, 1296

decryptionKey, generating, 1234

default buttons, specifying, 100-101

default content (Master Pages), 242

ContentPlaceHolder controls, 242

DefaultContent.aspx file, 245

DefaultMaster.master file, 242, 244

default namespaces, 14

default parameters, routes, 1143-1144

Default Skins, creating, 270

ShowSkin.aspx file, 271-272

Simple\TextBox.skin file, 270

default templates, creating, 1652-1656

Default value, 901

Default.aspx page, 1085, 1125-1126, 1130, 1132

DefaultButton property (Panel control), 114

DefaultContent.aspx file, 245

DefaultEvent design-time attribute, 1631

DefaultMaster.master file, 242, 244

DefaultMode property, 594

DefaultModel.RegisterContext method, 1494

DefaultProperty design-time attribute, 1631

defaultUrl configuration option (Forms authentication), 1207

DefaultUrl property (FormsAuthentication class), 1216

defaultValue attribute, 1298

DefaultValue design-time attribute, 1632

DefaultValue parameter, 1548-1549

DefaultValue property (Parameter object), 406

DefaultView property, 906

defining chart areas, 690

delete button fields, creating, 560, 562-563

Delete command, 428, 1492

Delete( ) method, 981

DeleteAllCookies.aspx page, 1274

DeleteButtonField class, 560, 562-563

DeleteButtonField.cs file, 561-562

DeleteCommand object, 892

DeleteCookie( ) method, 1257-1258

DeleteCookie.aspx page, 1273-1274

Deleted event, 398, 817

Deleted value, 901

DeleteImageUrl property (CommandField), 530

DeleteInactiveProfiles( ) method, 1323

DeleteMethod property (ObjectDataSource control), 772

DeleteParameters collection, 783

DeleteProfile( ) method, 1323

DeleteProfiles( ) method, 1323

DeleteRole( ) method, 1258

DeleteText property (CommandField), 530

DeleteUser( ) method, 1227

deleting

- cookies, 1273-1275
- data
  - DetailsView control, 597-598
  - FormView control, 619, 621-622
- inactive profiles, 1324
- LINQ to SQL, 981, 983
- records, 860, 862
- user locks, 1236

Deleting event, 403, 817

Demo folder, 1508

DemoController class, 1508

dependencies

- Data Caching, 1394-1396
- DataSource control key dependencies, 1383, 1385
- key dependencies, 1355
- Page Output Caching, 1351, 1353
- Partial Page Caching, 1372
- Polling SQL cache dependencies, 1399-1400
  - configuring applications for, 1402-1403
  - configuring databases for, 1400-1402
  - Data Caching, 1406-1407
  - DataSource Caching, 1405-1406
  - Page Output Caching, 1403-1404
- Push SQL cache dependencies, 1408-1409
  - configuring applications for, 1410
  - configuring databases for, 1409-1410
  - Data Caching, 1415-1416
  - DataSource Caching, 1412, 1414
  - Page Output Caching, 1411-1412
- SQL cache, 1399-1400

Dependency property (ControlCachePolicy class), 1370

deploying

- databases, 1571-1572
- encrypted web configuration files, 1561-1564

deployment configuration section, 1516

descending sorts, 496

description attribute, 1013

Description design-time attribute, 1632

Description property, 937, 1093

DescriptionUrl property, 106

- Image control, 105
- ImageButton control, 93
- ImageMap control, 110

design-time attributes, applying to controls, 1631-1632, 1634-1635

DesignerActionList, 1644

DesignerSerializationVisibility design-time attribute, 1632

DestinationPageUrl property, 1155

Detached value, 901

Detail Form, 1491

Details page, viewing, 1341

Details.aspx page, 535, 1340-1341

DetailsView control, 342, 567, 594, 1354

- deleting data with, 597-598
- displaying data with, 567, 569, 571
  - ShowDetailsView.aspx page, 567, 569
  - ShowEmployee.aspx page, 569-570
- displaying empty data with, 573-574, 576
  - ShowEmptyDataTemplate.aspx page, 574, 576
  - ShowEmptyDataText.aspx page, 573-574
- events, 598-599, 601
- fields, 571, 573
  - BoundField, 571
  - ButtonField, 571
  - CheckBoxField, 571
  - CommandField, 571
  - HyperLinkField, 571
  - ImageField, 571
  - ShowFields.aspx page, 571, 573
  - TemplateField, 571

formatting, 601-604
inserting data with, 593-596
ShowInsert.aspx page, 593
ShowInsertMode.aspx page, 594, 596
paging through data with, 576, 578
AJAX, 578-579
paging interface, customizing, 580-583
ShowPaging.aspx page, 576, 578
updating data with, 584-593
concurrency issues, 591-593
MasterDetailEdit.aspx page, 586, 588
ShowUpdate.aspx page, 584-585
templates, 588, 590-591

deviceFilters configuration section, 1516

Dhawan, Priya, 889

different language components, mixing in App_Code folder, 725-726

Direction property
Panel control, 114
Parameter object, 406

directives, 8. *See also names of specific directives*
%@ Import % directive, 13
%@ OutputCache % directive, 1337, 1369
CacheProfile attribute, 1361
Location attribute, 1349, 1351
VaryByControl attribute, 1342, 1344
VaryByCustom attribute, 1346-1349
VaryByHeader attribute, 1345-1346
VaryByParam attribute, 1338, 1342
%@ Page % directive, setting cultures, 1421

@Director parameter, 512

Director property, 977

Directories property, 1134

DirectoryExists( ) method, 1133

DisableCreatedUser property (CreateUserWizard control), 1172

disableExpiration attribute, 1397

disableMemoryCollection attribute, 1397

DisableNotifications( ) method, 1400

DisablePageTheme.aspx file, 280-281

DisableTableForNotification( ) method, 1400

DisableViewState.aspx page, 32-33

disabling
debugging, 50
Themes, 277, 279
ShowEnableTheming.aspx file, 277-278
Simple4\Calendar.skin file, 279
validation, 138-139
View State, 32-33

DisappearAfter property (Menu control), 1040

disconnected data access, 888-889
DataAdapter object, 889
batch updates, performing, 892-893, 896
displaying data with, 890, 892
populating DataTables with, 889-890
DataSet object, 908, 911
DataTable object, 896
building programmatically, 896, 898
DataRow states and versions, 901, 903, 906
DataRows, selecting, 898, 901
DataView object, 906

Discrete Animation, 1764

Display property (validation controls), 127-128

DisplayAfter property, 1731

displayCalendar( ) method, 199

DisplayCancelButton property (Wizard control), 226

DisplayIndex property (IDataItemContainer interface), 1656

displaying data
Chart Control, 690-694
DataList control, 640, 642-643
RepeatLayout property, 643
ShowDataList.aspx page, 640, 642
ShowFlowDataList.aspx page, 643
DetailsView control, 567, 569, 571
empty data, 573-574, 576
ShowDetailsView.aspx page, 567, 569
ShowEmployee.aspx page, 569-570
FormView control, 605-606

GridView control, 486-487, 489
empty data, 514-518
ShowMovies.aspx page, 486-487
ShowShoppingList.aspx page, 488
line charts, 692
mean value, 698
Repeater control, 624-625, 627
ShowRepeater.aspx file, 624-625
ShowRepeaterPhotos.aspx file, 626-627

DisplayMode property
BulletedList control, 473
ValidationSummary control, 165

DisplaySideBar property (Wizard control), 226

Distinct( ) method, 951

Divide( ) method, 984

DivView control, 1667
binding to SqlDataSource control, 1670-1671
CreateChildControls( ) method, 1669
DataBind( ) method, 1670
DivView.cs file, 1668

dll files. *See* assemblies

dlstMovies_Edit( ) method, 656

documents, saving to database tables, 185, 188-189
FileHandler.aspx page, 188-189
FileUploadDatabase.aspx page, 185, 187

domain attribute, 1258

domain configuration option (Forms authentication), 1207

Domain property (HttpCookie class), 1272-1273

domains, Forms authentication across, 1214-1216

DoSomething.aspx view, 1509

Downstream value, 1349

DpapiProtectedConfigurationProvider, 1557, 1561

DragHandleID property, 1750

DragPanel control, 1738, 1750-1751, 1754

drill-down reports, 709-710

DROP Assembly command, 923

drop shadows, rendering, 1583-1584

DropDown control, 1738

DropDownList control, 47, 325, 338, 410, 458, 460, 1423, 1425

DropShadow control, 1738

DropShadow.cs file, 1583

dsmgmt.exe tool, 1241

Duration attribute (%@ OutputCache % directive), 1369

Duration property (ControlCachePolicy class), 1370

dynamic compilation, 34-37, 724-725

dynamic data
building, 1492, 1494, 1496
overview of, 1491-1492
templates, 1496-1497, 1500

dynamic queries, 983-984, 986-987

dynamic SQL commands, 965

dynamically applying Skins, 293, 295, 297
ShowDynamicCSS.aspx, 295, 297
ShowDynamicSkin.aspx, 294-295

dynamically applying Themes, 291, 293
DynamicTheme.aspx file, 291, 293
Web.Config file, 293

dynamically loading Master Pages, 263
DynamicContent.aspx file, 263-264
DynamicMasterPage.cs file, 266
multiple content pages, 266-267
Web.Config file, 266-267

dynamically loading User controls, 320-321
%@ Reference %> directive, 322, 325, 327
caching, 1373, 1375
LoadControl( ) method, 320
multi-page wizards, creating, 327-331, 336
ShowFeaturedProduct.aspx example, 320-321

DynamicBottomSeparatorImageUrl property (Menu control), 1040

DynamicContent.aspx file, 263-264

DynamicEnableDefaultPopOutImage property (Menu control), 1040

DynamicFilter control, 713

DynamicFilterExpression, querying, 713

DynamicHorizontalOffset property (Menu control), 1040

DynamicHoverStyle object, 1041

DynamicImage.ashx page, 434-435

DynamicItemFormatString property (Menu control), 1040

DynamicItemTemplate, 1047

DynamicLayout property, 1731

DynamicMasterPage.cs file, 266

DynamicMenuItemStyle object, 1041

DynamicPopOutImageTextFormatString property (Menu control), 1040

DynamicPopOutImageUrl property (Menu control), 1040

DynamicPopulate control, 1738

DynamicSelectedStyle object, 1041

DynamicTopSeparatorImageUrl property (Menu control), 1040

DynamicVerticalOffset property (Menu control), 1040

E

EasyListBox control, 460

Edit Form, 1492

EditDataList.aspx page, 652, 656

EditGrid.aspx page, 510-511

EditImageUrl property (CommandField), 530

editing

- data
  - DataList control, 651, 656
  - FormView control, 612-616
  - GridView control, 510-512
- database data, 675, 678-679, 681
- Master Pages
  - controls, 261-263
  - HeaderContent.aspx file, 256
  - MetaContent.aspx file, 257
  - page headers, 256-258
  - properties, 258-259, 261
  - titles, 255-256
- movies, 367
- templates, 588, 590-591

EditItemStyle property (DataList control), 657

EditItemTemplate property, 546, 662, 1663

EditItemTemplate template, 646

EditListView.aspx page, 675

Editor design-time attribute, 1632

EditorBrowsable design-time attribute, 1632

EditRepeater.aspx page, 633, 640

EditRowStyle property (DetailsView control), 601

EditText property (CommandField), 530

EDM (Entity Data Model), 1492

EffectivePrivateBytesLimit property (Cache object), 1387

elements. ***See names of specific elements***

email

- addresses, validating, 151, 153
- encryption, 1166
- sending
  - ChangePassword control, 1188, 1190
  - CreateUserWizard control, 1166-1169
  - SmtpClient class, 11

Email ID, 1180

Email property (MembershipUser class), 1231

EmbeddedObjects property (MailDefinition class), 1167

Employee Directory application, 375-376

Employee object, passing as parameter, 790, 793-795

Employee.cs file, 778

Employee.cs file, 745-746

EmployeeData component

- EmployeeData.cs file, 791, 793
- UpdateEmployees.aspx page, 793

EmployeeData.cs file, 791, 793

Employees.cs file, 741

Employees/Employees.sitemap file, 1102

EmployeesDSSorting component, 810

EmployeesDSSorting.vb file, 810

EmployeesUISorting.cs file, 807

- **empty data, displaying**
  - DetailsView control, 573-574, 576
    - ShowEmptyDataTemplate.aspx page, 574, 576
    - ShowEmptyDataText.aspx page, 573-574
  - GridView control, 514, 516, 518
- **empty form fields, validating, 158-159**
- **EmptyDataRowStyle property (DetailsView control), 601**
- **EmptyDataTemplate, 518, 662, 664**
- **EmptyDataText property, 516**
- **EmptyItemTemplate, 662, 669**
- **Enable Action, 1764**
- **EnableAction animation, 1763**
- **EnableCaching property, 1377**
- **enableCrossAppRedirects attribute, 1214**
- **enableCrossAppRedirects configuration option (Forms authentication), 1208**
- **EnableCrossAppRedirects property (FormsAuthentication class), 1216**
- **Enabled property**
  - Button control, 85
  - CheckBox control, 78
  - FileUpload control, 180
  - HyperLink control, 119
  - ImageButton control, 93
  - LinkButton control, 88
  - ListItem control, 441
  - RadioButton control, 81
  - SiteMap class, 1090
  - TextBox control, 70
- **enableNotifications tool, 1402**
- **EnableNotifications( ) method, 1400**
- **enableSearchMethods( ) method, 1244**
- **EnableTableForNotifications( ) method, 1401**
- **EnableViewState property (GridView control), 522**
- **enabling**
  - automatic postbacks, 452, 454
  - cookieless anonymous identifiers, 1303
  - roles, 1248
  - Security Trimming, 1096
  - SQL caches, 1114
  - SQL Server Session, 1294
  - ViewState, GridView control, 521-522
- **Encode value (Mode property), 67**
- **EncryptConfig.aspx page, 1559**
- **encrypted configuration sections, 1556-1557**
  - creating programmatically, 1558, 1561
  - creating with aspnet_regiis command, 1557-1558
  - deploying, 1561-1564
  - DpapiProtectedConfigurationProvider, 1557
  - RsaProtectedConfigurationProvider, 1557
- **EncryptedXml class, 1557**
- **encryption**
  - connection strings, 388-389
  - email, 1166
  - encrypted configuration sections, 1556-1557
    - creating programmatically, 1558, 1561
    - creating with aspnet_regiis command, 1557-1558
    - deploying, 1561-1564
    - DpapiProtectedConfigurationProvider, 1557
    - RsaProtectedConfigurationProvider, 1557
  - Forms authentication, 1210, 1212, 1214
  - passwords, 1232-1233
- **EndExecuteNonQuery( ) method, 912**
- **EndExecuteReader( ) method, 912**
- **EndExecuteXmlReader( ) method, 912**
- **EndGetMovies( ) method, 912**
- **endHandler parameter, 916**
- **EndProcessRequest( ) method, 1479-1480**
- **EndRequestEventArgs, 1704**
- **Enlist, 857**
- **EnsureChildControls( ) method, 1588**
- **entering confirmation codes, 1175**
- **entities**
  - adding, 1006
  - templates, 1497, 1500
  - updating, 1006
  - validation, 996

Entities\App_Code\Movie.cs file, 952, 958

Entities\EditLinqDataSource.aspx page, 962

Entities\ListMoviesByBoxOffice.aspx page, 957

Entities\PartialMovie.aspx page, 959

Entities\SearchMovies.aspx page, 954

Entities\ShowLinqDataSource.aspx page, 960

Entity Data Model (EDM), 1492

Entity Lists, 1491

EntityBase class, 992

EntityBase file, 991

EntityCallOutValidator file, 991

EntityDataSource control, 994

EntityDataSource file, 991

EntityValidator file, 991

EnumerateCache.aspx page, 1388

Equal( ) method, 984

error handling, ObjectDataSource control, 822, 825

error messages, displaying summary of, 163, 165

ErrorMessage ID, 1180

ErrorMessage property, 162

errors

- command execution errors, 398, 400, 403
- displaying summary of, 163, 165
- UpdatePanel controls, 1714-1717, 1720
- validation errors, highlighting, 128
  - ShowSetFocusOnError.aspx page, 130
  - ShowValidators.aspx page, 131
  - ValidationImage.aspx page, 128

Eval( ) method, 363-364, 606, 729, 1006, 1660

EvaluateExpression( ) method, 1469

EvaluateIsValid( ) method, 167

EvaluationContext property (Configuration class), 1536

event handling, Session state, 1285-1287

events. *See also names of specific events*

- Button control, 86
- Calendar control, 197
- DataBinding, 598
- DataBound, 598
- DetailsView control, 598-599, 601
- exposing from User controls, 307, 309, 311
- GridView control, 546-547
- handling, Click events, 22-23, 25-28
- ImageButton control, 93
- ItemCommand, 598
- ItemCreated, 598
- ItemDeleted, 598
- ItemDeleting, 598
- ItemInserted, 598
- ItemInserting, 598
- ItemUpdated, 598
- ItemUpdating, 598
- LinkButton control, 89
- ModeChanged, 598
- ModeChanging, 598
- MultiView control, 220
- ObjectDataSource control, 817-818
  - Deleted, 817
  - Deleting, 817
  - error handling, 822, 825
  - Filtering, 817
  - Inserted, 817
  - Inserting, 817
  - ObjectCreated, 817
  - ObjectCreating, 817, 826-828
  - ObjectDisposing, 817
  - parameters, adding and modifying, 818, 820, 822
  - Selected, 817
  - Selecting, 817
  - Updated, 817
  - Updating, 817
- page events, handling, 44-46
- PageIndexChanged, 598
- PageIndexChanging, 598
- Page_InitComplete( ), 46
- postback events, handling, 1606-1616
  - CustomLinkButton example, 1606, 1608
  - event arguments, passing, 1610, 1613
  - IPostBackEventHandler interface, 1606, 1608
  - postback options, 1613, 1616
  - RaisePostBackEvent( ) method, 1606

Repeater control, 633, 640
DataBinding, 633
EditRepeater.aspx page, 633, 640
ItemCommand, 633
ItemCreated, 633
ItemDataBound, 633
SelectedIndexChanged, 454
SiteMap class, 1091
Wizard control, 227

ExceptionHandled property, 400

exceptions, NullReference, 1369

ExecuteAndSend( ) method, 928

executeInParallel parameter, 916

ExecuteInsert.aspx page, 429-430

ExecuteNonQuery( ) method, 860

ExecuteQuery( ) method, 983

ExecuteReader( ) method, 876, 878, 880-881

ExecuteRegisteredAsyncTasks( ) method, 916

ExecuteScalar( ) method, 874, 876

executing
asynchronous database commands, 911-912, 914
commands with parameters, 863-864
commands without parameters, 860, 862
database commands
canceling command execution, 403, 406
data source mode, changing, 397-398
database rows, filtering, 395-397
error handling, 398, 400, 403
inline SQL statements, 389, 391-392
stored procedures, 392, 395
dynamic SQL statements, 983
SqlDataSource commands, 426
ADO.NET parameters, 426, 428
Delete command, 428
Insert command, 428, 430
Select command, 428-435
Update command, 428
stored procedures, 865-872
from ASP.NET pages, 931-932
GetBoxOfficeTotals, 870
GetMovieCount, 868
Movie4 component listing, 865-866
Movie5 component listing, 870
MovieSelect, 867
MovieUpdate, 867
ShowMovie4.aspx page, 867-868
ShowMovie5.aspx page, 872
ShowMovieCount.aspx page, 869-870

execution life cycles, UpdatePanel control, 1697, 1699-1700, 1703-1704, 1706-1707

ExecutionTime statistic, 851-854

Expand( ) method, 1004

ExpandDepth property (TreeView control), 1070

Expanded property (TreeNode object), 1073

ExpandImageToolTip property (TreeView control), 1070

ExpandImageUrl property (TreeView control), 1070

Expires property, 1269, 1272

expiring
Data Caching
absolute expiration policy, 1390, 1392
sliding expiration policy, 1393-1394
DataSource caching
absolute cache expiration policy, 1376-1377
sliding cache expiration policy, 1378-1379
Forms authentication, 1210
Page Output Cache, 1353-1355, 1357

explicit localization expressions, 1438-1441

ExplicitShowMovies.aspx page, 785-786

ExponentialMovingAverage formula, 702

exposing
User control events, 307, 309, 311
User control properties, 304-306
PropertyRandomImage.ascx file, 304-305
ShowDeclarative.ascx file, 305
ShowProgrammatic.ascx file, 306

Expression property, 953

ExpressionBuilders, 1464-1465
AppSettingsExpressionBuilder class, 1464
ConnectionStringsExpressionBuilder class, 1465

LookupExpressionBuilder class, 1465, 1467, 1469
Lookup.config file, 1467
LookupExpressionBuilder.cs file, 1465
ShowLookupExpressionBuilder.aspx page, 1467-1468
Web.Config file, 1467
methods, 1469
ResourceExpressionBuilder class, 1465

expressions
lambda, 944-947
one-way databinding expressions, 362-366
queries, 983
two-way databinding expressions, 366-368

extending GridView control, 555
delete button fields, 560, 562-563
long text fields, 556, 559-560
validated fields, 563, 565-566

Extensible Output Caching, 1361

extension methods, 948-949

external files, placing configuration settings in, 1522-1523

external Web services, calling, 1776-1779

ExtractValues( ) method, 1660, 1663

F

Fade animation, 1763

FadeIn animation, 1763

FadeOut animation, 1763

FailureText ID, 1192

FavoriteMovie.aspx page, 440-441

Feedback.aspx page, 1683

Fiddler tool, 1336, 1675-1676

Field( ) method, 984

FieldHeaderStyle property (DetailsView control), 601

FieldHelloWorld component, 728-729

FieldHelloWorld.cs file, 728

fields
checking string lengths, 154, 156, 158
comparing against another form field, 150
comparing against fixed values, 148
configuring with CreateUserWizard control, 1163, 1165-1166
DataPager control, 682
declaring, 728-731
DetailsView control, 571, 573
BoundField, 571
ButtonField, 571
CheckBoxField, 571
CommandField, 571
HyperLinkField, 571
ImageField, 571
ShowFields.aspx page, 571, 573
TemplateField, 571
empty form fields, validating, 158-159
GridView control, 522-523
BoundField class, 523-524, 526
ButtonField class, 530, 533
CheckBoxField class, 526-527
CommandField class, 527, 529-530
HyperLinkField class, 534-536, 538-539
ImageField class, 539, 543
TemplateField class, 543, 545-546
shared fields, 733
validating length of, 168-170. *See also* validation

Fields property, 681

FieldTemplates folder, 1494

File class, 10

file dependencies
Page Output Caching, 1351, 1353
Partial Page Caching, 1372

File menu commands, New Web Site, 5

FileBytes property (FileUpload control), 180

FileContent property (FileUpload control), 180

FileExists( ) method, 1133

FileHandler.ashx page, 188-189

FileName property (FileUpload control), 180

FilePath property (Configuration class), 1536

files
- authorizing by location, 1224-1225
- generating source code from. *See* BuildProviders
- jQuery, creating, 1771-1773
- large files, uploading, 189-190, 193, 195
  - buffering thresholds, 189
  - FileUploadLarge.ashx page, 194
  - FileUploadLarge.aspx page, 190, 193
- placing configuration settings in, 1522-1523
- saving
  - to database, 185, 188-190, 193
  - to filesystem, 181, 184
- web configuration, 13, 1096
- Web.sitemap, 1097
- XML files, storing advertisements in, 208, 210, 212

Files property, 1134

FileService class, 1744

FileService.asmx file, 1743

filesystem, saving files to, 181, 184

FileUpload control, 180
- files
  - large files, uploading, 189-195
  - saving to databases, 185, 188-190, 193
  - saving to filesystems, 181, 184
- HttpPostedFile class, 180
- methods, 180
- properties, 180

FileUploadDatabase.aspx page, 185, 187

FileUploadFile.aspx page, 181, 183

FileUploadLarge.ashx page, 194

FileUploadLarge.aspx page, 190, 193

Fill( ) method, 892, 1416

FilterChart.aspx page, 697

FilteredTextBox control, 1738, 1754, 1756

FilterExpression property, 397

filtering, 813
- Chart Control, 694, 696, 698
- data, ObjectDataSource control, 813-816
- rows, 395-397

Filtering event, 403, 817

FilterMovies component, 815-816

FilterMovies.cs file, 815-816

FilterParameters collection, 783

filters
- QueryExtender control, 711-719
- transition, 627

FilterTopN method, 696

FinancialFormula method, 701, 704

FindContent.aspx file, 262-263

FindControl( ) method, 39, 98, 261-263, 555

FindInactiveProfilesByUserName( ) method, 1323

FindMaster.master file, 261-262

FindProfilesByUserName( ) method, 1323

FindUsersByEmail( ) method, 1227

FindUsersByName( ) method, 1227

FindUsersInRole( ) method, 1258

FinishButtonClick event, 227

FinishDestinationPageUrl property (Wizard control), 226

FinishNavigationTemplate, 226

Firebug, 1676

FirstHalf.cs file, 738

FirstPage.aspx page, 7, 35, 37

FirstPageCodeBehind.aspx page, 40-42

FirstPageCodeBehind.aspx.cs file, 41

FirstPageImageUrl property (PagerSettings class), 506, 581, 608

FirstPageText property (PagerSettings class), 506, 581, 608

fixed values, comparing form fields against, 148

Float.css file, 287-288

floating layouts, 287-288

Focus( ) method, 180
- Button control, 86
- CheckBox control, 78
- ImageButton control, 93
- ImageMap control, 111
- LinkButton control, 89

RadioButton control, 82
TextBox control, 70, 74, 76

FolderA\Web.Config file, 254

folders
App_Code, 725-726
App_GlobalResources, 1447
App_Themes, 270
Bin, adding assemblies to, 753

Font property (Label control), 61

fonts, listing installed, 357

FooterStyle property
BoundField, 525
DataList control, 657
DetailsView control, 601

FooterTemplate property (TemplateField), 546

FooterTemplate templates, 628, 646

FooterText property
BoundField, 525
DetailsView control, 602

Forecasting formula, 702

ForeColor property (Label control), 61

foreign language support, 1419-1420
culture names, 1420
Culture property, 1420-1421
CultureInfo class, 1434
comparing and sorting string values, 1437-1438
formatting string values, 1435-1436
methods, 1434
cultures, detecting automatically, 1428, 1431-1432
SelectCultureAuto.aspx page, 1431-1432
ShowAcceptLanguages.aspx page, 1430
cultures, setting in web configuration files, 1432-1433
cultures, setting manually, 1421-1423, 1425, 1428
%@ Page % directive, 1421-1422
DropDownList control, 1423, 1425
profiles, 1425, 1427
web configuration files, 1427-1428
effect on ASP.NET controls, 1433-1434
global resources
creating, 1446, 1448
retrieving programmatically, 1448-1449
strongly typed localization expressions, 1449-1450
local resources, 1438
explicit localization expressions, 1438-1441
implicit localization expressions, 1441-1443
page properties, 1443-1444
retrieving programmatically, 1444-1445
Localize control, 1450-1451
neutral cultures, 1420
specific cultures, 1420
UICulture property, 1420-1421

form data, submitting, 84
Button control, 84, 86
client scripts, 94-95
Command event, 101, 103
cross-page posts, 96, 98, 100
default buttons, specifying, 100-101
ImageButton control, 89-93
LinkButton control, 86, 88-89

form tag, 9

Format property, 920

FormatDataList.aspx page, 658-659

FormatDetailsView.aspx page, 602-604

FormatGrid.aspx page, 520-521

FormatLabel.aspx page, 62-63

FormatMovieTitles.aspx file, 365-366

FormatString property (LoginName control), 1186

formatting
Chart Control, backgrounds and plotting areas, 704, 706, 708
data, DataList control, 657, 659
DetailsView control, 601-604
FormView control, 604
deleting data with, 619, 621-622
displaying data with, 605-606
editing data with, 612-616
inserting data with, 616-617, 619
paging through data with, 607-609, 612

GridView control, 519-521
labels, 62-63
Menu control, 1040-1046
general properties, 1040-1041
MenuDesktop.aspx page, 1045-1046
MenuItem class properties, 1044
MenuLevelStyles.aspx page, 1042-1043
Style objects, 1041-1042
SiteMapPath control, 1016, 1018-1020
string values, 1435-1436
three-dimensional charts, 708
TreeView controls, 1070-1077
general properties, 1070
Style objects, 1073, 1075, 1077
TreeNode object properties, 1073
TreeViewImageSet.aspx page, 1071, 1073
TreeViewLevelStyles.aspx page, 1076-1077
TreeViewStyles.aspx page, 1074-1075

FormatTitle( ) method, 365-366

FormParameter objects, 353, 417-419

forms, 9

fields
checking string lengths, 154-158
comparing against another form field, 150
comparing against fixed values, 148
configuring with CreateUserWizard control, 1163, 1165-1166
empty form fields, validating, 158-159
validating length of, 168-170
labels, 63, 65
login forms, generating with Login control
custom authentication, 1160-1161
hiding Login control from authenticated users, 1155, 1157
Login.aspx page, 1150-1151
Master Pages, 1155, 1157
properties, 1152, 1154
redirecting users, 1155
templates, 1157, 1159-1160
multi-part forms, displaying, 223, 226
MVC Framework, 1507, 1509
registration, 1152
submitting, 84
Button control, 84, 86
client scripts, 94-95
Command event, 101, 103
cross-page posts, 96, 98, 100
default buttons, specifying, 100-101
ImageButton control, 89-93
LinkButton control, 86, 88-89
user registration forms, creating. *See* CreateUserWizard control
validating. *See* validation controls

Forms authentication, 1206-1207

across applications, 1210-1212, 1214
across domains, 1214-1216
configuring, 1207-1208
cookieless Forms authentication, 1209-1210
enabling, 1148-1149
encryption and validation options, 1211-1212, 1214
FormsAuthentication class, 1216-1220
FormsLogin.aspx page, 1219
methods, 1217
properties, 1216
Web.config file, 1217
sliding expiration, 1210
User class, 1220-1221

FormsAuthentication class, 1216-1218, 1220

FormsLogin.aspx page, 1219
methods, 1217
properties, 1216
Web.config file, 1217

FormsCookieName property (FormsAuthentication class), 1216

FormsCookiePath property (FormsAuthentication class), 1216

FormsLogin.aspx page, 1219

formulas, Chart Control, 698, 700-701, 704

FormView control, 342, 604, 977

deleting data with, 619, 621-622
displaying data with, 605-606
editing data with, 612-616

LinkButton, 615
ShowFormViewEditing.aspx page, 612, 614-615
inserting data with, 616-617, 619
InsertItemTemplate, 619
LinkButton, 619
ShowFormViewInserting.aspx page, 616-618
paging through data with, 607-609, 612
PagerSettings class, 608-609
ShowFormViewPagerTemplate.aspx page, 609, 611
ShowFormViewPaging.aspx page, 607-608

forward slash (/), 1124

FrameDetails.aspx page, 538-539

FrameMaster.aspx page, 537-538

Framework Class Library, 9, 12. *See also* components
assemblies, 14, 17
private, 15
referencing, 16-17
shared, 15
namespaces, 12-14
viewing classes in, 10

Friends, 744

from clause, 949

From property (MailDefinition class), 1167

Full value, 1541

fully rendered controls, building, 1576, 1579-1581, 1584, 1587
FullyRenderedControl.cs file, 1577
FullyRenderedWebControl.cs file, 1578
HtmlTextWriter class, 1580-1584
drop shadows, rendering, 1583-1584
HTML color table, rendering, 1581-1582
methods, 1580-1581
ShowFullyRenderedControl.aspx page, 1577-1578
ShowFullyRenderedWebControl.aspx page, 1579-1580
WebControl TagKey property, 1584-1587

FullyRenderedControl.cs file, 1577

FullyRenderedWebControl.cs file, 1578

functions
anonymous, 1775
varying output cache by, 1347, 1349

G

GAC (Global Assembly Cache), 15, 754-757

GameBrowser.aspx file, 717

GameSearchTextBox control, 719

GenerateCode( ) method, 1455, 1464

GenerateEmptyAlternateText property, 106
Image control, 105
ImageButton control, 93
ImageMap control, 111

GenerateKeys.aspx page, 1212-1215, 1234

GeneratePassword( ) method, 1227

Generic HTTP Handlers, creating, 1470-1473
ImageTextHandler.ashx file, 1470-1471
ShowImageTextHandler.aspx page, 1473

generics, 941-944

GenreDropDownList control, 719

GenreID property, 719

Get( ) function, 730

Get( ) method, 993, 1387

GetActiveView( ) method, 220

GetAll( ) method, 845, 850, 926

GetAllCookies.aspx page, 1271-1272

GetAllInactiveProfiles( ) method, 1323

GetAllNodes( ) method, 1093

GetAllProfiles( ) method, 1323

GetAllRoles( ) method, 1258

GetAllUsers( ) method, 1227-1228

GetAuthCookie( ) method, 1217

GetBoxOffice( ) method, 870

GetBoxOfficeTotals stored procedure, 870

GetBoxOfficeTotals.sql file, 870

GetBytes( ) method, 1213

GetCacheDependency( ) method, 1133

GetCacheKey( ) method, 1133
GetChildControlType( ) method, 1626
GetChildRows( ) method, 911
GetCodeExpression( ) method, 1469
GetControlValidationValue( ) method, 167
GetCookie.aspx page, 1270
GetCookieValues.aspx page, 1278
GetCultureInfo( ) method, 1434
GetCultureInfoByIetfLanguageTag( ) method, 1434
GetCultures( ) method, 1425, 1434
GetDataCode( ) method, 1464
GetDataSourceView( ) method, 1093
GetDirectory( ) method, 1133-1135
GetDynamicSort( ) method, 987
getElementById method, 1773
GetEmployees( ) method, 807
GetEnumerator( ) method, 1004, 1006, 1387
GetEvalData( ) method, 1469
GetExtension( ) method, 1135
GetFile( ) method, 1133-1134
GetFileHash( ) method, 1133
GetFileName( ) method, 1135
GetFileNameWithoutExtension( ) method, 1132
GetGlobalResourceObject( ) method, 1448-1449
GetHierarchicalDataSourceView( ) method, 1093
GetHistory( ) method, 227
GetListFromCommand( ) method, 942-943
GetLocalResourceObject( ) method, 1444-1445
GetMessage( ) method, 738
GetMovieCount stored procedure, 868
GetMovieCount( ) method, 802
GetMovieCount.sql file, 868
GetMovieData( ) method, 882
GetMovies( ) method, 1382
GetMoviesDataSet( ) method, 797
GetMoviesFromDB( ) method, 1392
GetNumberOfInactiveProfiles( ) method, 1323
GetNumberOfProfiles( ) method, 1323
GetNumberOfUsersOnline( ) method, 1227-1228
GetPagedMovies stored procedure, 803
GetPagedMovies.sql file, 803
GetPagedMovies2005 stored procedure, 805
GetPagedMovies2005.sql file, 805
GetPassword( ) method, 1231
GetPostBackClientHyperLink( ) method, 1610, 1612
GetPostBackEventReference( ) method, 1608
GetQuote( ) method, 734, 1780
GetRandomRow( ) method, 928
GetRandomRows( ) method, 928
GetRedirectUrl( ) method, 1217
GetRolesForUser( ) method, 1258
GetRootNodeCore( ) method, 1106
GetSection( ) method, 1524-1527, 1537
GetSectionGroup( ) method, 1537
GetSortedEmployees stored procedure, 811-812
GetSortedEmployees.sql file, 811-812
GetStepType( ) method, 227
GetSuggestions( ) method, 1743
GetTablesEnabledForNotifications( ) method, 1401
GetTableT, 956
getters, 937
GetUniqueID( ) method, 1395
GetUser( ) method, 1227
GetUserNameByEmail( ) method, 1227
GetUsersInRole( ) method, 1258
GetVaryByCustomString( ) method, 1347
GetWebApplication( ) method, 1525
GetWebApplicationSection( ) method, 1524-1527
.gif image files, authorization, 1225
Global Assembly Cache (GAC), 15, 754-757
global resources
  creating, 1446, 1448
  retrieving programmatically, 1448-1449
  strongly typed localization expressions, 1449-1450

global Themes, 290-291

Global.asax file, 1091, 1285, 1483, 1485

Global.asax.cs file, 1139

globalization configuration section, 1516

Glow.cs file, 1585

Google SiteMap files, generating, 1116-1120

- PublicSiteMap.ashx page, 1117, 1119
- PublicSiteMap.ashx results, 1120

GradientPanel control, 1639

- GradientPanel.cs file, 1637
- ShowGradientPanel.aspx page, 1639

GradientPanel.cs file, 1637

granting access to key containers, 1563

Graphics class, 10

grdMovieCategories_RowCommand( ) method, 533

grdMovieCategories_RowDataBound( ) method, 555

grdMovies_DataBound( ) method, 509

GridLines property

- DataList control, 657
- DetailsView control, 601
- GridView control, 521

GridView control, 341, 485, 719, 958, 1612

- column summaries, 550, 552
- concurrency issues, 512-514
- data keys, 492-495
- displaying data with, 486-487, 489
  - ShowMovies.aspx page, 486-487
  - ShowShoppingList.aspx page, 488
- displaying empty data, 514, 516, 518
- editing data with, 510-512
- events, 546-547
- fields, 522-523
  - BoundField class, 523-524, 526
  - ButtonField class, 530, 533
  - CheckBoxField class, 526-527
  - CommandField class, 527, 529-530
  - DeleteButtonField, 560, 562-563
  - HyperLinkField class, 534-536, 538-539
  - ImageField class, 539, 543
  - LongTextField, 556, 559-560
  - TemplateField class, 543, 545-546
  - ValidatedField, 563-566
- formatting, 519-521
- highlighting rows, 547, 549-550
- nested Master/Details forms, 552, 555
- paging through data, 501-509
  - AJAX, 503-504
  - AllowPaging property, 501, 503
  - custom paging interface, 504, 506, 509
  - PageGrid.aspx page, 502-503
  - PagerSettings class, 504, 506
  - PagerTemplate, 507, 509
- selecting data with, 489-491
- sorting data with, 495-497, 499, 501
  - AJAX, 496-497
  - AllowSorting property, 495-496
  - customized sorting interface, 497-501
- statistics, displaying, 855-856
- ViewState, 521-522

GridViewHandleError.aspx page, 401-402

GridViewRow object, 549-550

group clause, 949

GroupingText property (Panel control), 114-115

GroupName property (RadioButton control), 81

groups

- Profile groups, 1301-1302
- validation groups, 133, 138

GroupSeparatorTemplate, 662

GroupTemplate, 662, 666, 668-669

Guestbook component, 820

Guestbook.cs file, 820

guestbooks, displaying, 819-820, 822

GUID, 1140, 1303

H

hand-writing custom algorithms, 698

HandleError.aspx page, 399-400

HandleErrors.aspx page, 822-823

handling
- events
  - Click, 22-28
  - page events, 44-46
- method errors, 824
- session events, 1285-1287

HasChanged property (CacheDependency class), 1395

HasChildNodes property (SiteMapNode class), 1093

HasFile property
- Configuration class, 1536
- FileUpload control, 180

HashPasswordForStoringInConfigFile( ) method, 1217

HasKeys property (HttpCookie class), 1272

head tag, 257

HeaderContent.aspx file, 256

HeaderImageUrl property (BoundField), 525

headers
- Cache-Control, 1349
- Cache-Control HTTP, 1351
- COOKIE, 1210, 1264
- HeaderContent.aspx file, 256
- MetaContent.aspx file, 257
- Set-Cookie, 1264
- varying output cache by, 1345-1346

HeaderStyle property
- BoundField, 525
- DataList control, 657
- DetailsView control, 601

HeaderTemplate property (TemplateField), 226, 546

HeaderTemplate template, 628, 646

HeaderText property
- BoundField, 525
- DetailsView control, 601
- ValidationSummary control, 165
- Wizard control, 226

healthMonitoring configuration section, 1516

HelloJquery.aspx page, 1773

HelloWorld component, 722, 724

HelloWorld.cs file, 722

Hide Action, 1764

hiding
- Login control from authenticated users, 1155-1160
- Site Map nodes, 1100

hierarchical DataBound controls, 345, 347-348
- Menu, 346
- Movies.xml page, 347-348
- ShowHierarchicalDataBound.aspx page, 346-347
- TreeView, 346

highlighting rows, GridView control, 547-550

highlighting validation errors, 128
- ShowSetFocusOnError.aspx page, 130
- ShowValidators.aspx page, 131
- ValidationImage.aspx page, 128

HighlightRows.aspx page, 548-549

HorizontalAlign property (Panel control), 114

Host property (SmtpClient class), 10

hostingEnvironment configuration section, 1516

HotSpot classes, 106

HotSpotMode property (ImageMap control), 111

HotSpots property (ImageMap control), 111

HoverMenu control, 1738

HoverNodeStyle object, 1073

href attribute, 1612

HTML controls, declaring, 20-22

HTML table of colors, rendering, 1581-1582

HtmlControls.aspx page, 21

HTMLEditor control, 1738

HtmlEncode property (BoundField), 525

HtmlEncode( ) method, 948

HtmlForm class, 39

HtmlMeta controls, 258

HtmlTextWriter class, 1580-1581, 1583-1584
- drop shadows, rendering, 1583-1584
- HTML color table, rendering, 1581-1582
- methods, 1580-1581

HtmlTextWriterAttribute, 1581

HtmlTextWriterStyle, 1581

HtmlTextWriterTag, 1581

HTTP Handler, 692, 1470

anonymous HTTP Handlers, 1478-1481

IHttpAsyncHandler interface, 1478

RSSHandler.cs file, 1479

ShowRSSHandler.aspx page, 1481

Web.Config file, 1480

Generic Handlers, creating, 1470-1473

ImageTextHandler.ashx file, 1470-1471

ShowImageTextHandler.aspx page, 1473

IHttpHandler interface, 1475-1476

IHttpHandler interface, implementing, 1473

ImageHandler.cs file, 1473, 1475

ImageUpload.aspx page, 1476

Web.Config file, 1475-1476

HTTP Modules, authentication, 1207

HTTP Runtime, 1453

BuildProviders, 1454

DataBuildProvider, 1458-1464

SimpleBuildProvider, 1454, 1456-1458

ExpressionBuilders, 1464-1465

AppSettingsExpressionBuilder class, 1464

ConnectionStringsExpressionBuilder class, 1465

LookupExpressionBuilder class, 1465, 1467, 1469

methods, 1469

ResourceExpressionBuilder class, 1465

HTTP Handlers, 1470

anonymous HTTP Handlers, 1478-1481

Generic Handlers, creating, 1470-1473

IHttpHandler interface, 1473-1476

HttpApplication class, 1482-1483

custom HTTP Modules, creating, 1487-1489

Global.asax files, 1483, 1485

HttpApplication class, 1482-1483

custom HTTP Modules, creating, 1487-1489

QueryStringAuthenticationModule.cs file, 1487

Web.Config file, 1488

Global.asax files, 1483, 1485

HttpBrowserCapabilities class, 1349

HttpCachePolicy class, 1358-1360

HttpContext object, 747

HttpCookie class, 1272-1273

httpCookies configuration section, 1516

httpHandlers configuration section, 1516

httpModules configuration section, 1516

httpModules element (web.config file), 1129

HttpOnly property (HttpCookie class), 1272-1273

HttpPostedFile class, 180

httpRuntime configuration section, 1516

HttpSessionState class, 1284-1285

hybrid controls, building, 1590-1595

Login.cs file, 1590

LoginStandards.cs file, 1593

HyperLink control, 118-119

HyperLinkField class, 522, 534-536, 538-539

HyperLinkField fields, 571

hyperlinks, Master Pages, 251-253

ImageContent.aspx file, 253

MasterPages\ImageMaster.master file, 252-253

I

IBindableTemplate interface, 1660

ICallbackEventHandler interface, 173

ICreatable interface, 943

Id property, 937, 973

IDataItemContainer interface, 1656-1657

identity configuration section, 1516

Identity object, 1221

IduCount statistics, 854

IduRows statistics, 854

IEnumerableT, 949-950, 964

IHttpAsyncHandler interface, 1478

IHttpHandler interface, 1475-1476

implementing, 1473

ImageHandler.cs file, 1473, 1475

ImageUpload.aspx page, 1476

Web.Config file, 1475-1476

IHttpModule interface, 1129
IIS (Internet Information Services), 1567
IIS Site/Application Name option, 1567
Image control, 104, 106
image maps
  ImageButton control, 91-92
  ImageMap control, 106, 108, 110-111
    Click event, 111
    Focus( ) method, 111
    hot spots, 106
    navigation, 107
    postback, 108, 110
    properties, 110
ImageAlign property
  Image control, 106
  ImageButton control, 93
  ImageMap control, 111
ImageButton control, 89-93
  Click event, 93
  client scripts, 94-95
  Command event, 93, 101, 103
  cross-page posts, 96, 98, 100
  displaying on pages, 89-90
  Focus( ) method, 93
  image maps, 91-92
  properties, 93
ImageButtonTarget.aspx page, 91-92
ImageClickEventArgs class, 91
ImageContent.aspx file, 253
ImageField class, 523, 539, 543
ImageField fields, 571
ImageFolderPath property, 305
ImageHandler.cs file, 1473, 1475
ImageMap control, 106, 108, 110-111
  Click event, 111
  Focus( ) method, 111
  hot spots, 106
  navigation, 107
  postback, 108, 110
  properties, 110
ImageMapNavigate.aspx page, 107
ImageMapPostBack.aspx page, 108, 110
ImageRotator control
  ImageRotator.cs file, 1623
  ShowImageRotator.aspx page, 1625
ImageRotator.cs file, 1623
images
  alternative text, 90
  authorization, 1225
  Image control, 104, 106
  ImageMap control, 106, 108, 110-111
    Click event, 111
    Focus( ) method, 111
    hot spots, 106
    navigation, 107
    postback, 108, 110
    properties, 110
  Master Pages, 251-253
    ImageContent.aspx file, 253
    MasterPages\ImageMaster.master file, 252-253
  uploadin FileUpload control, 181, 184
ImageSet property (TreeView control), 1070
ImageSorting.aspx page, 498-499
ImageTextHandler.ashx file, 1470-1471
ImageToolTip property (TreeNode object), 1073
ImageUpload.aspx page, 1476
ImageUrl property, 251
  HyperLink control, 119
  Image control, 106
  ImageButton control, 93
  ImageMap control, 111
  MenuItem class, 1044
  TreeNode object, 1073
ImageUrlField property (AdRotator control), 207
img tag, 252
impersonation, enabling, 1542
implicit localization expressions, 1441-1443
importing
  applications, 1568
  namespaces, 13
  System.Collections.Generic namespace, 941

How can we make this index more useful? Email us at indexes@samspublishing.com

impressions (ads), tracking, 213, 215, 217, 219
- AdHandler.ashx page, 216
- AdRotatorStats.aspx page, 218
- AdRotatorTrack.aspx page, 214

inactive profiles, deleting, 1324
Include All Databases option, 1566
Include all IIS Settings option, 1566
Index.aspx page, 1508
inheritance, 739-741
Inheritance.cs file, 739-740
inheriting Profiles from custom classes, 1308-1309
Init( ) method, 1129, 1485
Initialize( ) method, 1114
InitializeCell( ) method, 562
InitializeCulture( ) method, 1427
InitializeDataCell( ) method, 559
InitializeRequestEventArgs, 1703
InitializeRoutes method, 1140
InitialValue property (RequiredFieldValidator control), 141, 143
initilizers, 937-939
inline SQL statements, executing, 389, 391-392
InProc value, 1291
input. *See* user input, accepting
InputStream property (FileUpload control), 180
Insert command, 428, 430
INSERT statement, 973
Insert( ) method, 926, 973, 1387, 1390
InsertCommand object, 892
Inserted event, 398, 817
Inserted event handler, 824
InsertEmployee( ) method, 790
InsertErrors.aspx page, 599-600
InsertImageUrl property (CommandField), 530
inserting
- LINQ to SQL, 973-974, 976
- records, 679

inserting data
- DetailsView control, 593-596
  - ShowInsert.aspx page, 593
  - ShowInsertMode.aspx page, 594, 596
- FormView control, 616-617, 619
  - InsertItemTemplate, 619
  - LinkButton, 619
  - ShowFormViewInserting.aspx page, 616-618

Inserting event, 403, 817
InsertItemTemplate class, 619, 662
InsertItemTemplate property (TemplateField), 546
InsertListView.aspx page, 679
InsertMethod property (ObjectDataSource control), 772
InsertMovie component, 824-825
InsertMovie.cs file, 824-825
InsertOnSubmit( ) method, 973
InsertParameters collection, 783
InsertRowStyle property (DetailsView control), 601
InsertText property (CommandField), 530
InsertVisible property (BoundField), 525
installation, ASP.NET Framework, 55-56
InstallCommon.sql file, 1251
InstallRoles.sql file, 1251
InstantiateIn( ) method, 1648, 1660
IntegerValidator, 1548
Intellisense, 744, 746
interfaces
- customizing, 684, 686
- declaring, 742-743
- IBindableTemplate, 1660
- ICallbackEventHandler, 173
- IDataItemContainer, 1656
- IHttpAsyncHandler, 1478
- IHttpHandler, 1473, 1475-1476
  - ImageHandler.cs file, 1473, 1475
  - ImageUpload.aspx page, 1476
  - Web.Config file, 1475-1476
- IHttpModule, 1129
- IPageHeader, 256
- IPostbackDataHandler, 1602
- IPostBackEventHandler, 1606, 1608

ITemplate, implementing, 1648, 1650-1652
IWizardStep, 328
paging, 503, 608

Internet Explore, cookies, 1266

Interpolated Animation, 1764

into clause, 949

intrinsic objects in components, 746-749

IPageHeader interface, 256

IPostbackDataHandler interface, 1602, 1606

IPostBackEventHandler interface, 1606, 1608

IQueryable interface, 956

IQueryableT, 966

IsAbsolute( ) method, 1135

IsAccessibleToUser( ) method, 1094

IsAppRelative( ) method, 1135

IsApproved property (MembershipUser class), 1231

IsAuthenticated property (Identity object), 1221

IsBodyHtml property (MailDefinition class), 1167

IsByteOrdered property, 920

IsCookieless property (HttpSessionState class), 1284

IsDbGenerated property, 953

IsDefaultCollection parameter, 1548

IsDescendantOf( ) method, 1094

IsDirectory property, 1134

IsDiscriminator property, 953

IsFixedLength property, 920

IsInPartialRendering property, 1686

IsInRole( ) method, 995, 1220

IsKey parameter, 1548

IsLockedOut property (MembershipUser class), 1231

IsNewSession property (HttpSessionState class), 1284

IsNothing( ) method, 1270

IsolateApps modifier, 1211

IsOnline property (MembershipUser class), 1231

IsPostBack property, 46-48

IsPrimaryKey property, 953-954

IsReadOnly property (HttpSessionState class), 1284

IsRequired parameter, 1548

IsUserInRole( ) method, 1258

IsUserName( ) method, 995

IsValid property (ServerValidateEventArgs class), 155

IsVersion property, 953-954

ItemCommand event, 598, 633

ItemCommand event handler, 640

itemContainer element, 664, 668

ItemCreated event, 598, 633

ItemDataBound event, 633

ItemDeleted event, 598

ItemDeleting event, 598

ItemInserted event, 598

ItemInserting event, 598

ITemplate interface, implementing, 1648-1652

ItemRotator control
ItemRotator.cs file, 1621
ShowItemRotator.aspx page, 1622-1623

ItemRotator.cs file, 1621

Items collection (List controls), 454, 458

Items property (SiteMapNode class), 1093

Items to Deploy option, 1566

ItemSeparatorTemplate, 662

ItemStyle property
BoundField, 525
DataList control, 657

ItemTemplate property (TemplateField), 546

ItemTemplate template, 606, 627, 645, 662-666

ItemUpdated event, 598

ItemUpdating event, 598

ItemWrap property (Menu control), 1040

IWizardStep interface, 328

J–K

- **JavaScript**
  - UpdatePanel control, 1694-1697
  - validation controls and, 126
- **JavaScript Object Notation (JSON), 1779**
- **JITTER, 17**
- **join clause, 950**
- **joining records, 970**
- **jQuery**
  - $( ) method, 1773-1774
  - $(document).ready( ) method, 1773-1774
  - files, creating, 1771-1773
  - libraries, 1770-1771
  - overview of, 1769
  - selectors, 1774-1776
- **jqueryClick.aspx page, 1775**
- **jqueryClick.js file, 1775**
- **JSON (JavaScript Object Notation), 1779**

- **key containers**
  - creating, 1562
  - exporting contents to XML files, 1563
  - granting access to, 1563
  - web.config file, 1562
- **key dependencies, 1355, 1383, 1385**
- **Key property (SiteMapNode class), 1093**
- **KeyPair.snk file, 754-755**
- **KeyPair.snk public key, 755**
- **Keys property (HttpSessionState class), 1284**
- **KeywordFilter property (AdRotator control), 207-208**
- **keywords, |DataDirectory|, 374**
- **Kill class, 1004**

L

- **Label control, 60**
  - accessibility guidelines, 65
  - displaying time with, 60-61
  - form field labels, 63, 65
  - formatting, 62-63
  - properties, 61-62
- **label for tag, 65**
- **label tag, 64**
- **Label.skin file, 275**
- **LabelForm.aspx page, 63-64**
- **lambda expressions, 944-947**
- **Lambda( ) method, 984**
- **Language Integrated Query. *See* LINQ**
- **LanguageChanges\AnonymousMethod.aspx page, 945**
- **LanguageChanges\App_Code\AutomaticProperties.cs file, 936**
- **LanguageChanges\App_Code\Product.cs file, 937**
- **LanguageChanges\App_Code\Product.vb file, 938**
- **LanguageChanges\LambdaExpression.aspx page, 946**
- **LanguageChanges\MyExtensions.cs file, 948**
- **LanguageChanges\NormalMethod.aspx page, 944**
- **languages, localizing applications for, 1419-1420**
  - culture names, 1420
  - Culture property, 1420-1421
  - CultureInfo class, 1434
    - comparing and sorting string values, 1437-1438
    - formatting string values, 1435-1436
    - methods, 1434
  - cultures, detecting automatically, 1428, 1431-1432
    - SelectCultureAuto.aspx page, 1431-1432
    - ShowAcceptLanguages.aspx page, 1430
  - cultures, setting in web configuration files, 1432-1433
  - cultures, setting manually, 1421-1423, 1425, 1428
    - %@ Page % directive, 1421-1422
    - DropDownList control, 1423, 1425
    - profiles, 1425, 1427
    - web configuration files, 1427-1428

effect on ASP.NET controls, 1433-1434
global resources
creating, 1446, 1448
retrieving programmatically, 1448-1449
strongly typed localization expressions, 1449-1450
local resources, 1438
explicit localization expressions, 1438-1441
implicit localization expressions, 1441-1443
page properties, 1443-1444
retrieving programmatically, 1444-1445
Localize control, 1450-1451
neutral cultures, 1420
specific cultures, 1420
UICulture property, 1420-1421

large files, uploading, 189-190, 193, 195
buffering thresholds, 189
FileUploadLarge.ashx page, 194
FileUploadLarge.aspx page, 190, 193

LastActivityDate property (MembershipUser class), 1231

LastLockoutDate property (MembershipUser class), 1231

LastLoginDate property (MembershipUser class), 1231

LastPageImageUrl property (PagerSettings class), 506, 581, 609

LastPageText property (PagerSettings class), 506, 581, 609

LastPasswordChangedDate property (MembershipUser class), 1231

layers (applications)
Business Logic layer, 761, 765
Data Access layer, 765, 769
multi-tier applications, building, 757-758
User Interface layer, 758-761

layouts, changing with Cascading Style Sheets, 285, 287-288, 290
absolute layouts, 288, 290
Absolute.css file, 288, 290
Float.css file, 287-288
floating layouts, 287-288
ShowLayout.aspx file, 285, 287

LayoutTemplate, 662-664, 666, 1158

@Id parameter, 512

Leaf value, 1054

LeafNodeStyle object, 1073

legends on charts, adding, 706

Length Animation, 1764

LengthValidator control, 167
LengthValidator.cs file, 168
ShowLengthValidator.aspx page, 169

LengthValidator.cs file, 168

let clause, 950

LevelMenuItemStyles property (Menu control), 1042

LevelSelectedStyles property (Menu control), 1042

LevelStyles property, 1075

LevelSubMenuStyles property (Menu control), 1042

libraries
component libraries, 749
compiling with vbc command-line compiler, 750-751
compiling with Visual Basic .NET Express, 751
compiling with Visual Studio .NET 2005, 752
referencing, 753-755, 757
Framework Class Library, 9, 12. *See also* components
assemblies, 14, 17
namespaces, 12-14
viewing classes in, 10
jQuery, 1770-1771

life cycles, execution, 1697-1707

LIKE Select, 967

limitations, CompareValidator, 148

line charts, 692, 694

LineImagesFolder property (TreeView control), 1070

LinkButton control, 86, 88-89
Click event, 89
client scripts, 94-95
Command event, 89, 101, 103

- cross-page posts, 96, 98, 100
- displaying on pages, 86-87
- events, 89
- Focus( ) method, 89
- properties, 88-89

linking to other pages with routes, 1142-1143. *See also* URLs

links, Menu control, 1022, 1024

LINQ (Language Integrated Query), 949-951
- custom LINQ to SQL base class, 991-995, 999-1000
- SQL, 951-952
  - associations, 959-960
  - building entities, 952-959
  - LinqDataSource control, 960, 962-963
  - standard database commands, 964-990

LINQ to SQL Debug Visualizer, 988

LinqDataSource control, 960, 962-963, 1375

list controls, 338, 341
- automatic postbacks, 452, 454
- binding, 341
- binding to data sources
  - declarative databinding, 442-443
  - programmatic databinding, 443, 446
- BulletedList, 338, 470-474
  - BulletStyle property, 471
  - DisplayMode property, 473
  - ShowBulletedList.aspx page, 470-471
  - ShowBulletedListHyperLinks.aspx page, 474
  - ShowBulletedListImage.aspx page, 472-473
- CheckBoxList, 338, 467, 469-470
- DropDownList, 338, 458, 460
- EasyListBox, 460
- Items collection, 454, 458
- list items
  - appending, 450, 452
  - declaring, 439, 441
  - reordering, 454, 458
- ListBox, 338, 463, 465, 467
  - selecting multiple items from, 465, 467
  - selecting single items from, 463, 465
- ListControl class, 439
- ListItem class, 439, 441
- MultiSelectList (custom control), 475
  - MultiSelectList.js, 481-482
  - MultiSelectList.vb, 475, 480
  - ShowMultiSelectList.aspx page, 482, 484
- RadioButtonList, 338, 461-463
- selected list items, determining, 446-450
  - SelectedIndex property, 446
  - SelectedItem property, 446
  - SelectedValue property, 446
  - SelectMovie.aspx page, 447-448
  - ShowMoviesByCategory.aspx page, 448-450
- ShowListControls.aspx page, 339

ListBox control, 338, 454, 463-467
- selecting multiple items from, 465, 467
- selecting single items from, 463, 465

ListControl class, 439

listing users, 1227-1228

ListItem class, 439, 441

ListPicker.aspx page, 455, 458

ListSearch control, 1738

ListUsers.aspx page, 1227-1228

ListView control, 342, 661
- databases
  - editing, 675, 678-679, 681
  - sorting, 672, 674
- GroupTemplate, 666, 668-669
- ItemTemplate, 662-664, 666
- LayoutTemplate, 662-664, 666
- rows, selecting, 669, 671-672

Literal control, 65, 68
- Mode property, 67-68
- modifying browser titles with, 65-66

Load Balance Timeout, 857

LoadControl( ) method, 320

LoadControlState( ) method, 1601

loading
- Master Pages dynamically, 263
  - DynamicContent.aspx file, 263-264
  - DynamicMasterPage.cs file, 266

multiple content pages, 266-267
Web.Config file, 266-267
User controls, 320-321
%@ Reference %> directive, 322, 325, 327
LoadControl( ) method, 320
multi-page wizards, creating, 327-329, 331, 336
ShowFeaturedProduct.aspx example, 320-321

LoadPostData( ) method, 1602, 1604

LoadStep( ) method, 328, 330

LoadWizardStep( ) method, 335

Local databases (SQL Server), 338, 373-374

local resources, 1438
explicit localization expressions, 1438-1441
implicit localization expressions, 1441-1443
page properties, 1443-1444
retrieving programmatically, 1444-1445

Local Security Authority. *See* LSA (Local Security Authority)

LocalComponent.cs file, 1446

LocalizablePage.aspx page, 1439-1440

LocalizablePageImplicit.aspx page, 1442

Localize control, 1450-1451

localizing applications, 1419-1420
culture names, 1420
Culture property, 1420-1421
CultureInfo class, 1434
comparing and sorting string values, 1437-1438
formatting string values, 1435-1436
methods, 1434
cultures, detecting automatically, 1428, 1431-1432
SelectCultureAuto.aspx page, 1431-1432
ShowAcceptLanguages.aspx page, 1430
cultures, setting in web configuration files, 1432-1433
cultures, setting manually, 1421-1423, 1425, 1428
%@ Page % directive, 1421-1422
DropDownList control, 1423, 1425
profiles, 1425, 1427
web configuration files, 1427-1428
effect on ASP.NET controls, 1433-1434
global resources
creating, 1446, 1448
retrieving programmatically, 1448-1449
strongly typed localization expressions, 1449-1450
local resources, 1438
explicit localization expressions, 1438-1441
implicit localization expressions, 1441-1443
page properties, 1443-1444
retrieving programmatically, 1444-1445
Localize control, 1450-1451
neutral cultures, 1420
specific cultures, 1420
UICulture property, 1420-1421

Location attribute (%@ OutputCache % directive), 1349, 1351

location element (web.config file), 1517-1518

Locations property (Configuration class), 1536

lockAllAttributesExcept attribute, 1519

lockAllElementsExcept attribute, 1519

lockAttributes attribute, 1520

lockElements attribute, 1520

locking
compilation sections, 1518
configuration settings, 1518, 1520

locking out users, 1235

lockItem attribute, 1520

LoggedOut event, 1185

logging LINQ to SQL queries, 988

LoggingOut event, 1185

Login control, 1152
adding to Master Pages, 1155, 1157
custom authentication, 1160-1161
hiding from authenticated users, 1155, 1157
Login.aspx page, 1150-1151
Login.cs file, 1590
LoginStandards.cs file, 1593

properties, 1152, 1154
redirecting users, 1155
templates, 1157, 1159-1160

login controls, 19, 1147-1150, 1152. *See also* Forms authentication
ChangePassword, 1186-1192
emails, sending, 1188, 1190
ShowChangePassword.aspx page, 1186, 1188
templates, 1190, 1192
CreateUserWizard, 1162
adding steps to, 1181, 1183
automatically generating passwords, 1171-1174, 1177
configuring form fields, 1163-1166
properties, 1162-1163
redirecting users, 1169-1170
sending email messages, 1166-1169
templates, 1177, 1179, 1181
Forms authentication, enabling, 1148-1149
Login, 1152
adding to Master Pages, 1155, 1157
custom authentication, 1160-1161
hiding from authenticated users, 1155, 1157
Login.aspx page, 1150-1151
properties, 1152, 1154
redirecting users, 1155
templates, 1157, 1159-1160
LoginName, 1185-1186
LoginStatus, 1183, 1185
LoginView, 1199
roles, 1201, 1203
ShowLoginView.aspx sample page, 1200-1201
PasswordRecovery, 1193-1199
security question and answer, 1196
sending original passwords, 1195-1196
ShowPasswordRecovery.aspx page, 1193-1194
templates, 1197, 1199
Secret.aspx sample page, 1148
UsersOnline
ShowUsersOnline.aspx page, 1229-1230
UsersOnline.cs file, 1229

login forms, generating with Login control
custom authentication, 1160-1161
hiding Login control from authenticated users, 1155, 1157
Login.aspx page, 1150-1151
Master Pages, 1155, 1157
properties, 1152, 1154
redirecting users, 1155
templates, 1157, 1159-1160

Login.aspx page, 1150-1151

Login.cs file, 1590

LoginContent.aspx page, 1157

LoginCreatedUser property (CreateUserWizard control), 1172

LoginCustom.aspx page, 1161

LoginImageUrl property (LoginStatus control), 1184

LoginMaster.master page, 1155-1156

LoginName control, 1185-1186

LoginReturn.aspx page, 1169

LoginStandards.cs file, 1593

LoginStatus control, 1183, 1185

LoginTemplate.aspx page, 1158, 1160

LoginText property (LoginStatus control), 1184

loginUrl configuration option (Forms authentication), 1208

LoginUrl property (FormsAuthentication class), 1216

LoginView control, 1199
roles, 1201, 1203
ShowLoginView.aspx sample page, 1200-1201

LoginViewRoles.aspx page, 1201, 1203

LogoutAction property (LoginStatus control), 1184

LogoutImageUrl property (LoginStatus control), 1184

LogoutPageUrl property (LoginStatus control), 1184

LogoutText property (LoginStatus control), 1184

long text fields, creating, 556, 559-560
LongTextField.cs file, 556, 559
ShowLongTextField.aspx page, 559-560

LongTextField class, 556, 559-560
LongTextField.cs file, 556, 559
ShowLongTextField.aspx page, 559-560
LongTextField.cs file, 557, 559
LongValidator, 1548
Lookup.config file, 1467
LookupExpressionBuilder class, 1465-1469
Lookup.config file, 1467
LookupExpressionBuilder.cs file, 1465
ShowLookupExpressionBuilder.aspx page, 1467-1468
Web.Config file, 1467
LookupExpressionBuilder.cs file, 1465
lost passwords, retrieving, 1194
LSA (Local Security Authority), 1211

M

Machine.config file, 1511
Machine.config.comments file, 1512
Machine.config.default file, 1512
machineKey configuration section, 1516
machineKey element (web.config file), 1211-1212, 1214
MailDefinition class, 1167, 1190
MailDefinition property (CreateUserWizard control), 1166
maintaining state. *See* state, maintaining
MaintainScrollPositionOnPostback attribute (%@ Page %> directive), 511, 656
MakeRelative( ) method, 1135
Management Studio Express (SQL Server), 370
ManageProfiles.aspx page, 1324-1325
mappedUrl attribute, 1124
mapping URLs, 1139-1142
maps, site. *See* Site Maps
MARS (Multiple Active Results Sets), 886, 888
Mask property, 1758
MaskedEdit control, 1738, 1756, 1758-1759
MaskType property, 1758
MassIndex formula, 702
Master Pages, 237
compared to Themes, 269
default content, creating, 242
ContentPlaceHolder controls, 242
DefaultContent.aspx file, 245
DefaultMaster.master file, 242, 244
hyperlinks, 251-253
ImageContent.aspx file, 253
MasterPages\ImageMaster.master file, 252-253
images, 251-253
ImageContent.aspx file, 253
MasterPages\ImageMaster.master file, 252-253
loading dynamically, 263
DynamicContent.aspx file, 263-264
DynamicMasterPage.cs file, 266
multiple content pages, 266-267
Web.Config file, 266-267
Login controls, adding, 1155, 1157
modifying controls in, 261
FindContent.aspx file, 262-263
FindMaster.master file, 261-262
nesting, 245
Products.aspx file, 249-250
SectionProducts.master file, 247
SectionServices.master file, 248-249
Services.aspx file, 250
Site.master file, 246-247
page headers, 256-258
HeaderContent.aspx file, 256
MetaContent.aspx file, 257
properties, 258
PropertyContent.aspx file, 259, 261
PropertyMaster.master file, 258-259
registering in web configuration, 254-255
simple Master page example
SimpleContent.aspx file, 241-242
SimpleMaster.master file, 238-239
titles, 255-256
viewing, 1340
Master.aspx page, 534, 1338-1339

MasterDetail.aspx page, 669

MasterDetailEdit.aspx page, 586, 588

MasterDetails.aspx page, 1343-1344

MasterPageFile attribute (%@ Page %> directive), 241

MasterPages\ImageMaster.master file, 252-253

Max Pool Size, 857

Max( ) method, 951

MaxByteSize property, 920

maxCachedResults attribute, 1258

MaxDataBindDepth property (TreeView control), 1070

MaximumDynamicDisplayLevels property (Menu control), 1040

MaximumRows property, 681

MaximumValue property (RangeValidator control), 144

maxInvalidPasswordAttempts attribute (web.config file), 1235

maxInvalidPasswordAttempts control, 1235

MaxLength property (TextBox control), 70

maxRequestLength setting, 189

mean value, 698

Membership, 1226

- ActiveDirectoryMembershipProvider, 1239, 1242-1244
- custom membership providers, 1244-1246
- Membership class, 1227-1230
  - custom Login controls, creating, 1228-1230
  - methods, 1227
  - registered users, listing, 1227
  - ValidatingPassword event, 1227
- MembershipUser class, 1230-1231
- passwords
  - encrypting and hashing, 1232-1233
  - password policy, 1234
- SqlMembershipProvider, 1237-1239
- users, locking out, 1235

Membership class, 1227-1230

- custom Login controls, creating, 1228-1230
- methods, 1227
- registered users, listing, 1227
- ValidatingPassword event, 1227

membership configuration section, 1516

Membership provider, 1152

Membership.xml file, 1246

MembershipProvider class, 741

MembershipProvider component, 741

MembershipUser class, 1230-1231

MembershipUser.IsApproved property, 1177

memory, cache

- caching database data with SqlDataSource control, 435, 437
- GAC (Global Assembly Cache), 15
- Global Assembly Cache, 754-755, 757
- SQL cache dependency, configuring, 1114

Menu control, 346, 1020

- adding items to, 1021-1022, 1024
- binding to database data, 1035, 1040
- binding to Site Maps, 1027, 1029, 1031
  - UsingMenu/MenuNavigate.aspx, 1029, 1031
  - UsingMenu/MenuSiteMap.aspx, 1028-1029
- binding to SiteMapDataSource controls, 1087-1088
- binding to XML files, 1031-1034
  - Menu.xml, 1031
  - MenuComplex.xml, 1033
  - MenuXML.aspx, 1032
  - MenuXMLComplex.aspx, 1034
- combining with MultiView control, 1024, 1027
- formatting, 1040-1042, 1044, 1046
  - general properties, 1040-1041
  - MenuDesktop.aspx page, 1045-1046
  - MenuItem class properties, 1044
  - MenuLevelStyles.aspx page, 1042-1043
  - Style objects, 1041-1042
- link buttons, 1022, 1024
- tabbed pages, 1024, 1027
- templates, 1046, 1049

Menu.xml file, 1031

MenuComplex.xml file, 1033

MenuDatabase.aspx page, 1036

MenuDesktop.aspx page, 1045-1046

MenuHyperLink.aspx page, 1021-1022

MenuItem class, properties, 1044

MenuItemBinding subtags, 1033

MenuLevelStyles.aspx page, 1042-1043

MenuLinkButton.aspx page, 1023-1024

MenuTabStrip.aspx page, 1025, 1027

MenuTemplates.aspx page, 1047, 1049

MenuXML.aspx page, 1032

MenuXMLComplex.aspx page, 1034

merging Site Maps, 1101-1102

- Employees/Employees.sitemap file, 1102
- Web.sitemap file, 1101-1102

Message property, 730

meta tag, 257

MetaContent.aspx file, 257

metaDescription attribute, 1103

method syntax, 950

MethodExpression, querying, 713

methods

- access modifiers, 744
- constructors
  - declaring, 733-734
  - overloading, 734-736
  - shared constructors, 734
- declaring, 726-728
- extension, 948-949
- overloading, 734-736
- Page_Load( ), 8
- shared methods, 726, 728
- signatures, 735

Microsoft Intermediate Language (MSIL), 17

Microsoft Management Console (MMC) Snap-In, 1514

Microsoft SQL Server. *See* SQL Server

MigrateAnonymous event, 1307

migrating anonymous profiles, 1307-1308

mike.simple.72cecc2a.cs file, 1457

Min Pool Size, 858

Min( ) method, 951

Minimal value, 1541

MinimumPrefixLength property, 1742

MinimumValue property (RangeValidator control), 144

minRequiredNonAlphanumericCharacters attribute (web.config file), 1234

MinRuntimeVersion parameter, 232

MMC (Microsoft Management Console) Snap-In, 1514

mobileControls configuration section, 1516

ModalPopup control, 1738

mode attribute, 1149

Mode property

- HttpSessionState class, 1284
- Literal control, 67-68
- PagerSettings class, 506, 581, 609

ModeChanged event, 598

ModeChanging event, 598

Model View Controller. *See* MVC (Model View Controller)

models, MVC Framework, 1502

Models folder, 1504

modes, Session state, 1291

Modified value, 901, 1541

modifiers, access, 743

modifying

- configuration settings, 1513
- passwords, requirements, 1234. *See also* ChangePassword control

modules, UrlRemapper, 1127-1132

- IHttpModule interface, 1129
- Products/Default.aspx page, 1130, 1132
- UrlMappings.config file, 1129
- UrlRemapper.cs file, 1127, 1129
- web configuration file, 1129

Move Animation, 1764

MoveTo( ) method, 227

Movie component, 1382

Movie.cs file, 943

Movie.data file, 1458

movie.data.72cecc2a.cs file, 1459

Movie.cs file, 1382

Movie1 component
Movie1.cs file, 845-846
ShowMovie1.aspx page, 847-848
Movie1.cs file, 845-846
Movie2 component
Movie2.cs file, 850-851
ShowMovie2.aspx page, 852
Movie2.cs file, 850-851
Movie3 component
Movie3.cs file, 860, 862
ShowMovie3.aspx, 862
Movie3.cs file, 860, 862
Movie4 component
Movie4.cs file, 865-866
ShowMovie4.aspx, 867-868
Movie4.cs file, 865-866
Movie5 component
Movie5.cs file, 870
ShowMovie5.aspx, 872
Movie5.cs file, 870
Movie6 component
Movie6.cs file, 877-878
ShowMovie6.aspx page, 878-879
Movie6.cs file, 877-878
Movie7 component
Movie7.cs file, 879
ShowMovie7.aspx page, 880
Movie7.cs file, 879
Movie8 component
Movie8.cs file, 890
ShowMovie8.aspx page, 891-892
Movie8.cs file, 890
Movie9 component
Movie9.cs file, 902
ShowMovie9.aspx page, 904, 906
Movie9.cs file, 902
MovieCategories.cs file, 789
MovieCollection component
binding to ObjectDataSource control, 773
MovieCollection.cs file, 772
MovieCollection.cs file, 772
MovieDataReader component
binding to ObjectDataSource control, 775
MovieDataReader.cs file, 774-775
MovieDataReader.cs file, 774-775
MovieDataSet component
binding to ObjectDataSource control, 777-778
MovieDataSet.cs file, 776
MovieDataSet.cs file, 776
MovieDataSource control, 832-833, 835
MovieDataSource.cs file, 832-833
MovieFileDependency.ascx page, 1372
MovieList.aspx page, 1353-1354
MovieListKeyDependency.aspx page, 1356
movies, editing, 367
Movies component, 783, 785-786
ExplicitShowMovies.aspx page, 785-786
Movies.cs file, 783
ShowMovies.aspx page, 785
Movies.ascx page, 1367
Movies.cs file, 783
Movies.xml file, 347-348, 1058
MoviesByCategory component, 826-827
MoviesByCategory.cs file, 826-827
MoviesComplex.xml file, 1059
MoviesDSPaging component, 800-802
MoviesDSPaging.cs file, 800-802
MovieSelect stored procedure, 867
MovieStoredProcedures.sql file, 867
MovieUIPaging.cs file, 797-798
MovieUpdate stored procedure, 867
MovieView control
MovieView.cs file, 1633
ShowMovieView.aspx page, 1635
MovieView.cs file, 1633
MovingAverage formula, 702
-moz style rules, 1584
Mozilla Firefox browser, cookies, 1265
MSIL (Microsoft Intermediate Language), 17
multi-column tables, creating with DataList control, 644-645

multi-page wizards, creating, 327-331, 336

IWizardStep interface, 328

LoadStep( ) method, 328

LoadWizardStep( ) method, 335

NextStep( ) method, 328

Wizard.aspx, 332, 335

WizardSteps\Step1.ascx, 328

WizardSteps\Step2.ascx, 330

WizardSteps\Step3.ascx, 331

multi-part forms, displaying, 223, 226. *See also* wizards

multi-tier applications, 757-758

multi-valued cookies, 1275, 1278-1279

creating, 1275

reading, 1278-1279

MultiColumnDataList.aspx page, 644-645

MultiHandleSlider control, 1738

MultiLine (TextMode property), 69

Multiple Active Results Sets (MARS), 886, 888

multiple content pages, loading Master Pages for, 266-267

multiple langage support. *See* foreign language support

multiple resultsets, returning, 882, 884

multiple segments, URLs, mapping, 1141-1142

Multiply( ) method, 984

MultiSelectList control, 458, 467, 475

MultiSelectList.js, 481-482

MultiSelectList.vb, 475, 480

ShowMultiSelectList.aspx page, 482, 484

MultiView control, 220

ActiveViewChanged event, 220

combining with Menu control, 1024, 1027

methods, 220

multi-part forms, displaying, 223, 226

properties, 220

tabbed page views, displaying, 220, 223

MultiViewForm.aspx page, 224

MultiViewTabs.aspx page, 221

MustInherit classes, 739-741

MustInherit keyword, 741

MutuallyExclusiveCheckBox control, 1738

MVC (Model View Controller) Framework, 1501

accepting form input, 1507, 1509

controllers, 1502

creating, 1502, 1504-1505

models, 1502

overview of, 1501

pages, 1505-1507

URL routing, 1139

views, 1502

MyBase keyword, 741

MyDatabase.designer.cs file, 958

MyKills.aspx page, 1005

myScript.js file, 1772

N

name attribute, 1298

name configuration option (Forms authentication), 1208

Name parameter, 1548

Name property, 920, 953, 1134

HttpCookie class, 1272

Identity object, 1221

Parameter object, 406

WizardStep control, 227

Named Skins, creating, 273-274

ShowNamedSkin.aspx file, 273-274

Simple2\TextBox.skin file, 273

SkinID property, 273

names, Themes, 270

Namespace statement, 737

Namespaced component, 737

Namespaced.cs file, 737

NamespaceDeclared property (Configuration class), 1536

namespaces, 12-14

declaring, 736-738

default namespaces, 14

importing, 13

naming conventions, 737

System.Data.Odbc, 844
System.Data.OleDb, 844
System.Data.OracleClient, 844
System.Data.SqlClient, 844
System.Data.SqlServerCe, 844
System.IO, 12

naming conventions for namespaces, 737

NavigateUrl property
HyperLink control, 119
HyperLinkField, 539
TreeNode object, 1073

NavigateUrlField property (AdRotator control), 207

navigation, 1123, 1492
ImageMap control, 107
remapping URLs
web configuration files, 1124, 1126
wildcard matches (UrlRemapper module), 1127-1132
Site Maps, 1083
AutoSiteMapProvider, 1106, 1109-1110
custom attributes, 1103-1105
Google SiteMap files, generating, 1116-1120
merging, 1101-1102
Security Trimming, 1096-1098, 1100
SiteMap class, 1090-1093
SiteMapDataSource control, 1084-1090
SiteMapNode class, 1093-1095
SqlSiteMapProvider, 1111, 1114, 1116
VirtualPathProvider class, 1132
limitations, 1133
methods, 1133-1134
properties, 1134
registering, 1135
VirtualPathUtility class, 1135

navigation controls, 19, 1011, 1016
Menu, 1020
adding items to, 1021-1022, 1024
binding to database data, 1035, 1040
binding to Site Maps, 1027-1031
binding to XML files, 1031-1034
combining with MultiView control, 1024, 1027
formatting, 1040-1046
link buttons, 1022, 1024
tabbed pages, 1024, 1027
templates, 1046, 1049
Site Maps, 1011
binding Menu controls to, 1027-1031
binding TreeView controls to, 1057-1058
example of, 1012
siteMapNode elements, 1013
SiteMapPath, 1013
declaring in pages, 1013, 1015
formatting, 1016, 1018-1020
properties, 1015-1016
templates, 1018-1020
SqlHierarchicalDataSource, 1077
classes, 1080-1081
ShowMenu.aspx page, 1078-1079
ShowTreeView.aspx page, 1079
TreeView, 1049
adding nodes to, 1049, 1051, 1054
AJAX, 1066-1067, 1070
binding to database data, 1061, 1064
binding to Site Maps, 1057-1058
binding to XML files, 1058-1059, 1061
check boxes, 1054, 1057
formatting, 1070-1077
Populate On Demand feature, 1064-1066

nested Master/Details forms, GridView control, 552, 555

NestedMasterDetail.aspx page, 553, 555

NestedUpdatePanels.aspx page, 1689

nesting
Master Pages, 245
Products.aspx file, 249-250
SectionProducts.master file, 247
SectionServices.master file, 248-249
Services.aspx file, 250
Site.master file, 246-247
UpdatePanel control, 1688, 1692

.NET Framework
ASP.NET pages
code-behind pages, 39, 41-43
control trees, 38-39

creating, 5, 8-9
debugging, 48, 50, 52
dynamic compilation, 34-35, 37
page events, handling, 44-46
page execution lifestyle, 44
Page.IsPostBack property, 46, 48
tracing, 52-55
CLR (Common Language Runtime), 17-18
Framework Class Library, 9, 12
assemblies, 14, 16-17
namespaces, 12-14
viewing classes in, 10

.NET Passport authentication, 1206

NETWORK SERVICE accounts, 1410

NetworkServerTime statistics, 854

neutral cultures, 1420

new features, 936, 938-939
anonymous types, 940
automatic properties, 936-937
extension methods, 948-949
generics, 941-944
initilizers, 937-939
lambda expressions, 944-947
LINQ, 949-951
type inference, 939-940

New Web Site command (File menu), 5

New Web Site dialog box, 5

NewImageUrl property (CommandField), 530

NewPassword ID, 1192

NewsRotator control, 1364

NewsRotator.cs file, 1364

NewText property (CommandField), 530

Next( ) method, 734

NextButtonClick event, 227

NextMonthText property (Calendar control), 197

NextPageImageUrl property (PagerSettings class), 506, 581, 609

NextPageText property (PagerSettings class), 506, 581, 609

NextPrevFormat property (Calendar control), 197

NextPreviousPagerField, 682

NextResult( ) method, 884

NextSibling property (SiteMapNode class), 1093

NextStep( ) method, 328

NextView command (MultiView control), 223

no-compile ASP.NET pages, 1469

NoBot control, 1738

NodeIndent property (TreeView control), 1070

nodes
adding to Site Maps, 1091, 1093
About.asax file, 1092-1093
Global.asax file, 1091
hiding in Site Maps, 1100

NodeStyle object, 1016, 1073

NodeTemplate, 1018

NodeWrap property (TreeView control), 1070

NoExpandImageUrl property (TreeView control), 1070

None value, 1054, 1349

NotifyParentProperty design-time attribute, 1632

NullDisplayText property (BoundField), 525

NullImageUrl property (ImageField), 543

NullReference exception, 1369

numberOfVisits property, 1301, 1304

NumericPagerField, 682

NumericUpDown control, 1738

O

Object-Relational Mapping (ORM), 711

ObjectCreated event, 817

ObjectCreating event, 817, 826-828

ObjectDataSource control, 351, 380, 771, 956, 993
binding to components, 772, 774
binding to DataReaders, 774-776
binding to DataSets, 776-778
binding to Web Services, 779-780, 783
concurrency, 828, 832
custom ObjectDataSource controls, 832-833, 835

custom parameter objects, creating, 835-841
  page property parameters, 838-841
  username parameters, 835-837
data filtering, 813-816
data source paging, 798-805
  GetPagedMovies stored procedure, 803
  GetPagedMovies2005 stored procedure, 805
  MoviesDSPaging component, 800-802
  ShowDSPaging.aspx page, 798-799
data source sorting, 808–812
  EmployeesDSSorting component, 810
  GetSortedEmployees stored procedure, 811-812
  ShowDSSorting.aspx page, 808-809
DataSource Caching, 1380, 1382
error handling, 822, 825
events, 817-818
  Deleted, 817
  Deleting, 817
  Filtering, 817
  Inserted, 817
  Inserting, 817
  ObjectCreated, 817
  ObjectCreating, 817, 826-828
  ObjectDisposing, 817
  parameters, adding and modifying, 818, 820, 822
  Selected, 817
  Selecting, 817
  Updated, 817
  Updating, 817
parameters, 783
  adding and modifying, 818, 820, 822
  DeleteParameters, 783
  FilterParameters, 783
  InsertParameters, 783
  Movies component, 783, 785-786
  passing objects as, 790, 793-795
  SelectParameters, 783
  types of, 787, 789
  UpdateParameters, 783
properties, 772
user interface paging, 795, 797-798
user interface sorting, 805, 807-808

ObjectDisposing event, 817

objects

Cache, 437, 1387, 1389
Command, 859
  executing commands with parameters, 863-864
  executing stored procedures, 865-872
  returning resultsets, 876, 878-881
  returning single values, 874, 876
commands, executing without parameters, 860, 862
Connection, 848-849
  Connection pooling, 856-858
  provider statistics, retrieving, 849-856
ControlParameter, 410-412, 414
  properties, 410
  ShowControlParameter.aspx page, 410-411
  ShowPageControlParameter.aspx page, 413-414
CookieParameter, 415, 417
  properties, 415
  Vote.aspx page, 415
DataAdapter, 889
  batch updates, performing, 892-893, 896
  displaying data with, 890, 892
  populating DataTables with, 889-890
DataReader, 397, 881
  Multiple Active Results Sets (MARS), 888
  multiple resultsets, returning, 882, 884
  Read( ) method, 881
DataSet, 397, 908, 911
DataTable, 896
  building programmatically, 896, 898
  DataRow states and versions, 901, 903, 906
  DataRows, selecting, 898, 901
  populating with SqlDataAdapter object, 889-890
DataView, 906

definition of, 722
DeleteCommand, 892
Employee, passing as parameter, 790, 793-795
FormParameter, 417, 419
HtppContext, 747
Identity, 1221
InsertCommand, 892
Parameter, 407-409
properties, 406-407
ShowDetailsView.aspx page, 407
ShowDetailsViewExplicit.aspx page, 408-409
passing as parameters, 790, 793-795
Principal, 1220
ProfileParameter, 420, 422
properties, 420
ShowProfileParameter.aspx page, 421-422
Web.config file, 420
Profiles
accessing from components, 1321-1322
anonymous profiles, migrating, 1307-1308
anonymous users, supporting, 1303-1307
complex profile properties, 1310-1311, 1314-1316
custom Profile providers, 1327-1331
defining, 1297-1298
displaying, 1298, 1301
inheriting from custom classes, 1308-1309
Profile groups, 1301-1302
Profile provider configuration, 1326-1327
ProfileManager class, 1323, 1325
saving automatically, 1317, 1320
QueryStringParameter, 422-424
properties, 422
ShowQueryStringParameterMaster.aspx page, 422-423
ShowQueryStringParamterDetails.aspx page, 424
SelectCommand, 892
Session, 1284
SessionParameter, 424, 426
properties, 424
ShowSessionParameter.aspx page, 425-426
SqlParameter, 863
UpdateCommand, 892
ODBC, connecting to (SqlDataSource control), 384
Off value, 1291
OldValuesParameterFormatString attribute, 979
OldValuesParameterFormatString property (SqlDataSource control), 513
OLE DB, connecting to (SqlDataSource control), 384
OnClick event, 1761
OnClientClick property, 95, 622
Button control, 85
LinkButton control, 88
OnDataBound event, 694
one-click web applications, 1572-1573
OnError parameter, 232
OnHoverOut event, 1761
OnHoverOver event, 1761
OnLoad event, 1761
OnMouseOut event, 1761
OnMouseOver event, 1761
OnPreInit( ) method, 267, 293
OnPreRender( ) method, 173, 481
OnSaleProduct.cs file, 740
Opacity Action, 1764
Open( ) method, 1134
OpenFile( ) method, 1133
opening configuration files
on remote servers, 1533-1534, 1536
on same server, 1529-1530, 1533
OpenMachineConfiguration( ) method, 1524, 1529
OpenMappedMachineConfiguration( ) method, 1524
OpenMappedWebConfiguration( ) method, 1524

OpenWebConfiguration( ) method, 1524, 1533

Operator property (CompareValidator control), 146

Option Infer option, 940

Options parameter, 1548

Or( ) method, 984

Oracle databases, connecting to (SqlDataSource control), 384, 386

orderby clause, 950

OrderBy( ) method, 966

OrderByDescending( ) method, 966

OrderByExpression, querying, 714

OrderForm.aspx page, 122, 124-125

Orientation property (Menu control), 1040

Original value, 901

ORM (Object-Relational Mapping), 711

output cache

- specifying location of, 1349, 1351
- User Control output cache, sharing, 1370
- varying
  - by browser, 1346-1347
  - by control, 1342, 1344
  - by custom functions, 1347, 1349
  - by header, 1345-1346
  - by parameter, 1338-1339, 1341-1342

output parameters, 872

OutputCache directive, 240, 1367

OutputCacheFileDependency.aspx page, 1352-1353

OutputCacheProfile.aspx page, 1361

overloading

- constructors, 734-736
- methods, 734-736

overriding

- locked configuration sections, 1519
- Master Pages, 255
- Skin properties, 276-277

OverwriteChanges, 828

P

Package/Publish Web Properties screen, 1567

packaging web applications, 1565-1566, 1568

page execution lifestyle, 44

page headers, Master Pages, 256-258

- HeaderContent.aspx file, 256
- MetaContent.aspx file, 257

Page InitComplete event, 1597

page layouts, changing with Cascading Style Sheets, 285-290

- absolute layouts, 288, 290
- Absolute.css file, 288, 290
- Float.css file, 287-288
- floating layouts, 287-288
- ShowLayout.aspx file, 285, 287

Page Load event, 44

Page Output Caching, 1336-1337

- %@ OutputCache % directive, 1337
  - CacheProfile attribute, 1361
  - Location attribute, 1349, 1351
  - VaryByControl attribute, 1342, 1344
  - VaryByCustom attribute, 1346-1349
  - VaryByHeader attribute, 1345-1346
  - VaryByParam attribute, 1338, 1342
- CachePageOutput.aspx sample page, 1337
- expiring programmatically, 1353-1357
- file dependencies, 1351, 1353
- HttpCachePolicy class, 1358-1360
- Polling SQL cache dependencies, 1403-1404
- profiles, 1360-1361
- Push SQL cache dependencies, 1411-1412
- specifying cache location, 1349, 1351
- varying output cache
  - by browser, 1346-1347
  - by control, 1342, 1344
  - by custom functions, 1347, 1349
  - by header, 1345-1346
  - by parameter, 1338-1339, 1341-1342

Page PreInit event, 293

page property parameters, creating, 838-841

page render blocks, 9

page views

- multi-part forms, displaying, 223, 226
- tabbed page views, displaying, 220, 223

Page.Eval( ) method, 363

Page.IsPostBack property, 47, 358

Page.IsValid property, 126-127

Page.ResolveUrl( ) method, 252, 1124, 1354

PageButtonCount property (PagerSettings class), 506, 581, 609

PagedControlId property, 681

PageExplicit.aspx page, 1443-1444

PageGrid.aspx page, 502-503

PageGridPreviousNext.aspx page, 505-506

PageImplicit.aspx page, 1444

PageIndexChanged event, 547, 598

PageIndexChanging event, 547, 598

PageLoadedEventArgs, 1703

PageLoadedEventArgs.updated property, 1707

PageLoadingEventArgs, 1703

PagePropertyParameter class, 838-839, 841

PagePropertyParameter.cs file, 838-839

Pager control, 1612

- Pager.vb, 1610
- ShowPager.aspx, 1613

Pager.cs file, 1610

PageRequestManager.beginRequest event, 1700, 1707

PageRequestManager.endRequest event, 1700

PageRequestManager.initializeRequest event, 1700

PageRequestManager.pageLoaded event, 1700, 1707

PageRequestManager.pageLoading event, 1700

PagerSettings class, 504, 506, 608-609

PagerStyle property (DetailsView control), 601

PagerTemplate class, 507, 509, 609, 611

pages

- ASP.NET pages, 914, 917
- code-behind pages, 39, 41-43
  - FirstPageCodeBehind.aspx page, 40-41
  - FirstPageCodeBehind.aspx.cs file, 41
  - partial classes, 42
  - presentation pages, 41-43
  - when to use, 43
- control trees, 38-39
- creating, 5, 8-9
- debugging, 48, 50, 52
- dynamic compilation, 34-35, 37
- Master Pages. *See* Master Pages
- MVC Framework, 1505-1507
- no-compile ASP.NET pages, 1469
- page events, handling, 44-46
- page execution lifestyle, 44
- Page.IsPostBack property, 46, 48
- Trace.axd, 54
- tracing, 52-55

pages configuration section, 1516

PageSize property, 681

PageTemplateGrid.aspx page, 507, 509

PageTemplates folder, 1494

PageTrace.aspx page, 52-53

Page_Init( ) method, 945

Page_InitComplete( ) event, 46

Page_Load for MyKills.aspx.cs file, 1003

Page_Load( ) event handler, 724, 1100, 1170, 1415

Page_Load( ) method, 8, 21, 258, 668, 1373, 1652, 1665

Page_PreInit( ) event handler, 264

Page_PreRender( ) handler, 1317

paging

- data source, 686-688, 798-805
  - GetPagedMovies stored procedure, 803
  - GetPagedMovies2005 stored procedure, 805
  - MoviesDSPaging component, 800-802
  - ShowDSPaging.aspx page, 798-799

interfaces, customizing, 684, 686
records, 808, 968
user interface paging, 795, 797-798

paging through data
DetailsView control, 576, 578
AJAX, 578-579
paging interface, customizing, 580-583
ShowPaging.aspx page, 576, 578
FormView control, 607-609, 612
PagerSettings class, 608-609
ShowFormViewPagerTemplate.aspx page, 609, 611
ShowFormViewPaging.aspx page, 607-608
GridView control, 501-506, 509
AJAX, 503-504
AllowPaging property, 501, 503
custom paging interface, 504, 506, 509
PageGrid.aspx page, 502-503
PagerSettings class, 504, 506
PagerTemplate, 507, 509

PagingBulletedList control, 1738

Palette property, 704

Panel control, 101, 111, 114, 116, 118
GroupingText property, 114-115
hiding/displaying controls with, 111, 113
properties, 114
properties list, 114
ScrollBars property, 116, 118

Panel controls, 1347

PanelGroupingText.aspx page, 114-115

PanelScrollBars.aspx page, 116

Parallel Animation, 1763

Parameter object, 353, 406, 407-409
ControlParameter, 410-412, 414
properties, 410
ShowControlParameter.aspx page, 410-411
ShowPageControlParameter.aspx page, 413-414
CookieParameter, 415, 417
properties, 415
Vote.aspx page, 415
FormParameter, 417, 419
Parameter, 407-409
properties, 406-407
ShowDetailsView.aspx page, 407
ShowDetailsViewExplicit.aspx page, 408-409
ProfileParameter, 420, 422
properties, 420
ShowProfileParameter.aspx page, 421-422
Web.config file, 420
properties, 406-407
QueryStringParameter, 422-424
properties, 422
ShowQueryStringParameterMaster.aspx page, 422-423
ShowQueryStringParameterDetails.aspx page, 424
SessionParameter, 424, 426
properties, 424
ShowSessionParameter.aspx page, 425-426
ShowDetailsView.aspx page, 407
ShowDetailsViewExplicit.aspx page, 408-409

Parameter parameters (ObjectDataSource control), 787

Parameter( ) method, 984

ParameterDataSourceExpression class, 712

parameters
default, routes, 1143-1144
ObjectDataSource control, 783
adding and modifying, 818, 820, 822
custom parameter objects, creating, 835-841
DeleteParameters, 783
FilterParameters, 783
InsertParameters, 783
Movies component, 783, 785-786
parameter types, 787, 789
passing objects as parameters, 790, 793-795
SelectParameters, 783
UpdateParameters, 783

URLs, mapping, 1140-1141
varying output cache by, 1338-1342
parameters.xml file, 1567
Parent value, 1054
ParentLevelsDisplay property (SiteMapPath), 1015
ParentNode property (SiteMapNode class), 1093
ParentNodeStyle object, 1073
Parse( ) method, 923
ParseChildren attribute, 1618-1619, 1622-1625
ContentRotator control example
ContentRotator.cs file, 1618
ShowContentRotator.aspx page, 1620-1621
ImageRotator control example
ImageRotator.cs file, 1623
ShowImageRotator.aspx page, 1625
ItemRotator control example
ItemRotator.cs file, 1621
ShowItemRotator.aspx page, 1622-1623
ParseExpression( ) method, 1469
partial classes, 42, 738
Partial keyword, 738
Partial Page Caching, 1336, 1361-1362
caching dynamically loaded User Controls, 1373, 1375
ControlCachePolicy class, 1370-1371
file dependencies, 1372
post-cache substitution, 1362-1364, 1366
sharing User Control output cache, 1370
User Controls, 1366-1367, 1369
passing
objects as parameters, 790, 793-795
postback event arguments, 1610, 1613
PassThrough value (Mode property), 67
Password (TextMode property), 69
Password ID, 1180
password-protecting Login controls, 1148
passwordAttemptWindow attribute (web.config file), 1235
PasswordConfirmation.htm file, 1173
passwordFormat attribute (web.config file), 1195, 1232, 1245
PasswordQuestion property (MembershipUser class), 1231
PasswordRecovery control, 1193-1199
security question and answer, 1196
sending original passwords, 1195-1196
ShowPasswordRecovery.aspx page, 1193-1194
templates, 1197, 1199
PasswordRecovery.txt file, 1199
PasswordRecoveryTemplate.aspx page, 1197, 1199
passwords
changing with ChangePassword control, 1186-1192
emails, sending, 1188, 1190
ShowChangePassword.aspx page, 1186, 1188
templates, 1190, 1192
email, 1166
encrypting and hashing, 1232-1233
generating automatically with CreateUserWizard control, 1171-1174, 1177
CodeConfirmation.htm file, 1175
ConfirmCode.aspx page, 1176
CreateUserWizardCodeConfirmation.aspx page, 1173
CreateUserWizardPasswordConfirmation.aspx page, 1172
PasswordConfirmation.htm file, 1173
lost, retrieving, 1194
password policy, 1234
PasswordRecovery control, 1193-1199
security question and answer, 1196
sending original passwords, 1195-1196
ShowPasswordRecovery.aspx page, 1193-1194
templates, 1197, 1199
PasswordStrength control, 1739
path attribute, 1475
path configuration option (Forms authentication), 1208

Path property (HttpCookie class), 1272-1273

PathDirection property (SiteMapPath), 1015

paths

- applying configuration settings to, 1517-1518
- VirtualPathProvider class, 1132
  - limitations, 1133
  - methods, 1133-1134
  - properties, 1134
  - registering, 1135
  - VirtualPathUtility class, 1135

PathSeparator property (SiteMapPath), 1016

PathSeparatorStyle object, 1016

PathSeparatorTemplate, 1018

patterns, 1491

- MVC. *See* MVC Framework
- routes, 1144-1145

-pef option (Protect Encrypt Filepath), 389

percentagePhysicalMemoryUsedLimit attribute, 1397

performance

- debugging, 50
- UpdatePanel controls, 1720, 1723-1724

PerformValidation property (PostBackOptions class), 1614

PersistChildren design-time attribute, 1631

PersistenceMode attribute, 1632, 1650

persistent cookies, 1264, 1268-1269

photo galleries, viewing, 667

PhotoGallery.aspx page, 431, 433, 667

pie charts, 694

PlaceHolder control, 320

plots, areas, 690

plotting areas, Chart control, 704, 706, 708

points, 694

policies, passwords, 1234

Polling SQL cache dependencies, 1399-1400

- configuring applications for, 1402-1403
- configuring databases for, 1400-1402
- Data Caching, 1406-1407
- DataSource Caching, 1405-1406
- Page Output Caching, 1403-1404

PollingSQLDataCache.aspx page, 1406-1407

PollingSQLDataSourceCache.aspx page, 1405-1406

PollingSQLOutputCache.aspx page, 1403-1404

PolygonHotSpot class, 106

Pooling, 858

pooling SqlConnections, 856-858

pop-up calendars, creating, 199

PopOutImageUrl property (MenuItem class), 1044

Populate On Demand feature, TreeView controls, 1064, 1066

PopulateMenu( ) method, 1039

PopulateNodesFromClient property, 1067

PopulateTopNodes( ) method, 1069

populating

- DataTables with SqlDataAdapter object, 889-890
- TreeView controls, 1064, 1066

popup alert boxes, displaying, 165-166

PopupControl control, 1739

Port property (SmtpClient class), 10

Position property (PagerSettings class), 506, 581, 609

PositiveTimeSpanValidator, 1548

post-cache substitution, 1336, 1362-1366

PostBackOptions class, 1613, 1616

postbacks, 1776

- asynchronous, 1707-1714
- definition of, 1602
- postback data, handling, 1602, 1604, 1606
  - CustomTextBox example, 1602-1605
  - IPostbackDataHandler interface, 1602
  - LoadPostData( ) method, 1602, 1604
  - RaisePostDataChangedEvent( ) method, 1602, 1604
  - RegisterRequiresPostBack( ) method, 1604
- postback events, handling, 1606-1616
  - CustomLinkButton example, 1606, 1608
  - event arguments, passing, 1610, 1613
  - IPostBackEventHandler interface, 1606, 1608

postback options, 1613, 1616
RaisePostBackEvent( ) method, 1606

PostBackUrl property
Button control, 85
ImageButton control, 93
LinkButton control, 88

PostedFile property (FileUpload control), 180

Preferences component, 747, 749

Preferences.cs file, 747

PreInit event, 295

PreparedExecs statistics, 854

Prepares statistics, 854

PreRender event, 1317

PreRenderComplete event, 1597

presentation pages, 41-43

Previous property (VirtualPathProvider class), 1134

PreviousButtonClick event, 227

PreviousPageImageUrl property (PagerSettings class), 506, 581, 609

PreviousPageText property (PagerSettings class), 506, 581, 609

PreviousSibling property (SiteMapNode class), 1093

PrevMonthText property (Calendar control), 197

PrevView command (MultiView control), 223

Principal object, 1220-1221

priorities, Data Caching, 1396

priority element, 1117

Priority property (MailDefinition class), 1167

Private access, 744

private assemblies, 15

privateBytesLimit attribute, 1397

privateBytesPollTime attribute, 1397

procedures. *See* stored procedures

processModel configuration section, 1516

ProcessRequest( ) method, 37, 189, 435, 1483

Product class, 938

Product control
implementing on pages, 1659
Product.cs file, 1657

Product.cs file, 1657

ProductConstructor component, 736

ProductConstructor.cs file, 736

ProductForm control
implementing on pages, 1663
ProductForm.cs file, 1661

ProductForm.cs file, 1661, 1663

Products.aspx file, 249-250, 758, 760-761

Products.cs file, 742

Products/Default.aspx page, 1089, 1130, 1132

Products/FirstProduct.aspx page, 1094-1095

ProductsByTag.aspx.cs, 1142

ProductSelectAll( ) method, 769

profile configuration section, 1516

Profile properties, 266, 1297

Profile.IsDirty property, 1317

Profile.Save( ) method, 1318

ProfileCommon object, 1321

ProfileComponent.cs, 1322

ProfileComponent.cs file, 1321

ProfileManager class, 1323, 1325

ProfileParameter object, 353, 420, 422, 787
properties, 420
ShowProfileParameter.aspx page, 421-422
Web.config file, 420

ProfileProvider attribute, 1309

profiles
accessing from components, 1321-1322
anonymous profiles, migrating, 1307-1308
anonymous users, supporting, 1303-1307
complex profile properties, 1310-1311, 1314-1316
culture profiles, 1425, 1427
custom Profile providers, 1327-1329, 1331
defining, 1297-1298
displaying, 1298, 1301
inactive, deleting, 1324
inheriting from custom classes, 1308-1309
Page Output Caching, 1360-1361
Profile groups, 1301-1302
Profile provider configuration, 1326-1327

ProfileManager class, 1323, 1325
saving automatically, 1317, 1320
ProgramGlobal.aspx page, 1449
ProgramGlobalTyped.aspx page, 1450
ProgramLocal.aspx page, 1444-1445
programmatic databinding, 356, 358, 443, 446, 488
ProgrammaticDataBinding.aspx page, 444-445
ProgramOutputCache.aspx page, 1359
properties
access modifiers, 744
ActiveDirectoryMembershipProvider, 1243-1244
AdRotator control, 207
automatic, 936-937
Button control, 85
Cache object, 1387
CacheDependency class, 1394
Calendar control, 197-199
CheckBox control, 78
CompareValidator control, 146
Configuration class, 1536-1537
ControlCachePolicy class, 1370
ControlParameter object, 410
CookieParameter object, 415
cookies, 1272-1273
CreateUserWizard control, 1162-1163
DataView object, 906
declaring, 728-731
exposing from User controls, 304-306
PropertyRandomImage.ascx file, 304-305
ShowDeclarative.ascx file, 305
ShowProgrammatic.ascx file, 306
FileUpload control, 180
FormsAuthentication class, 1216
HttpCachePolicy class, 1358
HttpSessionState class, 1284
HyperLink control, 119
IDataItemContainer interface, 1656
Identity object, 1221
Image control, 105
ImageButton control, 93
ImageMap control, 110-111
Label control, 61-62
LinkButton control, 88-89
Login control, 1152, 1154
LoginName control, 1186
LoginStatus control, 1184-1185
MailDefinition class, 1167
Master Pages, 258
PropertyContent.aspx file, 259, 261
PropertyMaster.master file, 258-259
MembershipUser class, 1230-1231
Menu control, 1040-1042
general properties, 1040-1041
Style objects, 1041-1042
MenuItem class, 1044
MultiView control, 220
ObjectDataSource control, 772
Panel control, 114
Parameter object, 406-407
PostBackOptions class, 1614
ProfileParameter object, 420
Profiles, complex profile properties, 1310-1311, 1314-1316
QueryStringParameter object, 422
RadioButton control, 81
RegularExpressionValidator control, 154
RequiredFieldValidator control, 139
ServerValidateEventArgs class, 155
SessionParameter object, 424
shared properties, 733
SiteMap class, 1090
SiteMapDataSource control, 1086-1090
ShowStartingNode, 1086
StartFromCurrentNode, 1086-1090
StartingNodeOffset, 1087
StartingNodeUrl, 1087-1088
SiteMapNode class, 1093
SiteMapPath control, 1015-1016
SmtpClient class, 10
TextBox control, 70
TreeNode object, 1073
ValidationSummary control, 165
VirtualPathProvider class, 1134

WebConfigurationManager class, 1523
Wizard control, 226

Property( ) method, 984

PropertyContent.aspx file, 259, 261

PropertyExpression, 715, 719

PropertyHelloWorld component, 730-731

PropertyHelloWorld.cs file, 729

PropertyMaster.master file, 258-259

PropertyName property

ControlParameter object, 410
ProfileParameter object, 420

PropertyOrField( ) method, 984

PropertyRandomImage control, 304

PropertyRandomImage User control, 305

PropertyRandomImage.ascx file, 304-305

Proposed value, 901

Protected access, 743-744

Protected Friends, 744

protection configuration option (Forms authentication), 1208

ProtectSection( ) method, 1558, 1561

provider attribute, 1298

Provider property

SiteMap class, 1090
SiteMapNode class, 1093

provider statistics, retrieving, 849-856

displaying in GridView control, 855-856
list of statistics, 854
Movie2 component, 850, 852
ResetStatistics( ) method, 853
RetrieveStatistics( ) method, 853

ProviderName property, 386, 1231

providers, creating Site Maps providers

AutoSiteMapProvider, 1106, 1109-1110
SqlSiteMapProvider, 1111, 1114, 1116

Providers property, 1090, 1552

ProviderSettingsCollection class, 1553

ProviderSpecific value, 1314

ProviderUserKey property (MembershipUser class), 1231

provisioning new websites, 1543, 1546

ProvisionSite.aspx page, 1544

Public access, 743-744

PublicSiteMap.ashx page, 1117, 1119

PublicSiteMap.ashx results, 1120

Publish/Package SQL screen, 1571

Pulse Animation, 1763

Push SQL cache dependencies, 1408-1409

configuring applications for, 1410
configuring databases for, 1409-1410
Data Caching, 1415-1416
DataSource Caching, 1412, 1414
Page Output Caching, 1411-1412

PushSQLDataCache.aspx page, 1415-1416

PushSQLDataSourceCache.aspx page, 1413-1414

PushSQLOutputCache.aspx page, 1411-1412

Q

queries

associations, 959-960
dynamic, 983-987
entities, building, 952-959
expressions, 983
jQuery
$( ) method, 1773-1774
$(document).ready( ) method, 1773-1774
creating files, 1771-1773
libraries, 1770-1771
overview of, 1769
selectors, 1774-1776
LinqDataSource control, 960, 962-963
SQL, cache dependencies, 1408

query syntax, 950

QueryExtender control, 711-719

querying, 712

ControlFilterExpression, 712
CustomExpression, 712
DynamicFilterExpression, 713

MethodExpression, 713
OrderByExpression, 714
PropertyExpression, 715
RangeExpression, 715
SearchExpression, 716-717

QueryStringAuthenticate.aspx page, 1215-1216

QueryStringAuthenticationModule.cs file, 1487

QueryStringField property (SqlDataSource control), 422

QueryStringParameter object, 353, 422-424, 787
properties, 422
ShowQueryStringParameterMaster.aspx page, 422-423
ShowQueryStringParameterDetails.aspx page, 424

Question ID, 1180

Quotation web services, 1776

quotations, displaying at random (RandomQuotation.ascx), 318-319

QuotationService.asmx file, 1777

Quote component, 734

Quote.cs file, 734

R

RadioButton control, 79, 82-83, 111
automatic postback, 82-83
CheckedChanged event, 82
displaying on pages, 79
Focus( ) method, 82
properties, 81

RadioButtonAutoPostBack.aspx page, 82-83

RadioButtonList control, 83, 338, 461-463

RaisePostBackEvent( ) method, 1606, 1610

RaisePostDataChangedEvent( ) method, 1602, 1604

Random class, 10

random quotations, displaying, 734

Random.Next( ) method, 734

RandomDataLayer component
RandomDataLayer.cs file, 931
ShowRandomDataLayer.aspx page, 933

RandomDataLayer.cs file, 931

RandomImage example
RandomImage.ascx, 300-301
ShowRandomImage.ascx, 301-302

RandomQuotation control
RandomQuotation.ascx, 318-319
ShowRandomQuotation.ascx, 319

RandomRows class, 929

RandomRows.cs file, 929

RangeExpression, querying, 715

RangeValidator control, 121, 143-145
Display property, 127-128
implementing on OrderForm page, 122, 124-125
IsValid property, 126-127

RansomDataSource, 715

Rating control, 1739

Read( ) method, 847, 881, 923

read-only properties, 731

reading
configuration sections from current application, 1524-1525, 1527
ShowConfigApp.aspx page, 1524-1525
ShowConfigPath.aspx page, 1528
ShowConfigRelative.aspx page, 1525, 1527
cookies, 1269-1270, 1272
multi-valued cookies, 1278-1279

readOnly attribute, 1298

ReadOnly property
BoundField, 526
SiteMapNode class, 1093
TextBox control, 70

rebasing, 252

records
caching, 970
databases, viewing, 664
deleting, 860, 862
DetailsView control, 597-598
FormView control, 619, 621-622

displaying. *See* displaying data
editing, FormView control, 612-616
inserting, 679
DetailsView control, 593-596
FormView control, 616-617, 619
joining, 970
paging, 808, 968
DetailsView control, 576-583
FormView control, 607-609, 612
sorting, 806
updating, 860, 862
DetailsView control, 584, 586, 588, 591-593

recovering passwords. *See* PasswordRecovery control

RectangleHotSpot class, 106

/recurse option, 751

RedirectFromLoginPage( ) method, 1217, 1220

redirecting users
CreateUserWizard control, 1169-1170
Login control, 1155

RedirectToLoginPage( ) method, 1217

referencing
assemblies, 16-17
component libraries
assemblies, adding to Bin folder, 753
assemblies, adding to Global Assembly Cache, 754-755, 757
services, 1002-1004

regenerateExpiredSessionId attribute (sessionState element), 1290

RegexStringValidator, 1548-1549

Register declaration, 692

Register.aspx page, 1151

RegisterArrayDeclaration( ) method, 1696

RegisterAsyncTask( ) method, 914, 916

RegisterClientScriptBlock( ) method, 1696

RegisterClientScriptInclude( ) method, 1696

RegisterClientScriptResource( ) method, 1696

registered usernames, displaying, 1185-1186

RegisterExpandoAttribute( ) method, 1696

registering
custom configuration sections, 1549
Master Pages in web configuration, 254-255
SimpleBuildProvider, 1456
stored procedures, 930
Themes in web configuration file, 280-281
DisablePageTheme.aspx example, 280-281
Web.Config file example, 280
User controls in web configuration file, 303
user-defined type assembly, 923
VirtualPathProvider class, 1135

RegisterOnSubmitStatement( ) method, 1696

RegisterRequiresPostBack( ) method, 1604

RegisterStartupScript( ) method, 1696

RegisterVirtualPathProvider( ) method, 1135

registration, 1166

registration forms, 1152. *See also* CreateUserWizard control

regular expressions, 1144

RegularExpression control, 148

RegularExpressionValidator control, 122, 151, 153

RejectChanges( ) method, 902

relative URLs, 252

remapping home pages, 1126

remapping URLs
web configuration files, 1124, 1126
current URL, determining, 1125
Default.aspx page, 1125-1126
urlMappings element, 1124
Web.Config code listing, 1124
wildcard matches (UrlRemapper module), 1127-1132
IHttpModule interface, 1129
Products/Default.aspx page, 1130, 1132
UrlMappings.config file, 1129
UrlRemapper.cs file, 1127, 1129
web configuration file, 1129

remote servers, opening configuration files on, 1533-1534, 1536

Remove( ) method, 1284, 1387

RemoveLock.aspx page, 1236

RemoveOutputCacheItem( ) method, 1353, 1355

RemoveTrailingSlash( ) method, 1135

RemoveUserFromRole( ) method, 1258

RemoveUserFromRoles( ) method, 1258

RemoveUsersFromRole( ) method, 1258

RemoveUsersFromRoles( ) method, 1258

removing user locks, 1236

Render( ) method, 1577

RenderBeginTag( ) method, 1580, 1582-1583

RenderContents( ) method, 481, 1578, 1593, 1612

RenderCurrentNodeAsLink property (SiteMapPath), 1016

RenderEndTag( ) method, 1580

rendering properties (TextBox control), 70

RenderMode property, 1686

RenewTicketIfOld( ) method, 1217

reordering list items, 454, 458

ReorderList control, 1739

RepeatColumns property
- CheckBoxList control, 469
- DataList control, 644
- RadioButtonList control, 462

RepeatDirection property
- CheckBoxList control, 469
- DataList control, 644
- RadioButtonList control, 462

Repeater control, 342, 351, 623
- displaying data with, 624-625, 627
  - ShowRepeater.aspx file, 624-625
  - ShowRepeaterPhotos.aspx file, 626-627
- events, 633, 640
  - DataBinding, 633
  - EditRepeater.aspx page, 633, 640
  - ItemCommand, 633
  - ItemCreated, 633
  - ItemDataBound, 633
- templates, 627-633
  - AlternatingItemTemplate, 627
  - FooterTemplate, 628
  - HeaderTemplate, 628
  - ItemTemplate, 627
  - SeparatorTemplate, 628, 630, 633

RepeatLayout property
- CheckBoxList control, 469
- DataList control, 643
- RadioButtonList control, 462

reports, drill-down, 709-710

Request.AppRelativeCurrentExecutionFilePath property, 1125

Request.Cookies collection, 1272

Request.Path property, 1125

Request.QueryString object, 1141

Request.RawUrl property, 1125

Request.UserLanguages property, 1428

requestLengthDiskThreshold setting, 189

RequiredFieldValidator control, 121, 139
- ControlToValidate property, 139
- Display property, 127-128
- implementing on OrderForm page, 122, 124-125
- InitialValue property, 141, 143
- IsValid property, 126-127
- ShowInitialValue.aspx page, 142
- ShowRequiredFieldValidator.aspx page, 140
- Text property, 139

RequiredTextBox.cs file, 1587

RequireEmail property (CreateUserWizard control), 1165-1166

RequiresJavaScriptProtocol property (PostBackOptions class), 1614

RequireSSL property (FormsAuthentication class), 1217

requiresSSL configuration option (Forms authentication), 1208

ResetPassword( ) method, 1231

ResetStatistics( ) method, 853

ResizableControl control, 1739

Resize Animation, 1764

ResourceExpressionBuilder class, 1465

ResourceKey property (SiteMapNode class), 1093

Response.RemoveOutputCacheItem( ) method, 1354

resultsets

- MARS, 886, 888
- multiple resultsets, returning, 882, 884
- returning, 876, 878-881

RetrieveStatistics( ) method, 851, 853

retrieving lost passwords, 1194

ReturnUrl parameter, 1155, 1170

RewritePath( ) method, 1129

rich controls, 19, 179

- AdRotator
  - AdCreated event, 207
  - displaying advertisements, 207-208
  - properties, 207
  - storing advertisements in database tables, 212-213
  - storing advertisements in XML files, 208, 210, 212
  - tracking impressions and transfers, 213-219
- Calendar
  - calendars, displaying, 197
  - displaying calendars with, 195-196
  - events, 197
  - multiple dates, selecting, 198-199
  - pop-up calendars, creating, 199
  - properties, 197
  - rendering calendars from databases, 202, 206-207
- FileUpload, 180
  - HttpPostedFile class, 180
  - methods, 180
  - properties, 180
  - saving files to database, 185, 188-190, 193
  - saving files to filesystem, 181, 184
  - uploading large files, 189-190, 193, 195
- MultiView, 220
  - ActiveViewChanged event, 220
  - methods, 220
  - multi-part forms, displaying, 223, 226
  - properties, 220
  - tabbed page views, displaying, 220, 223
- Wizard, 226
  - displaying wizards with (ShowWizard.aspx), 228, 231
  - events, 227
  - methods, 227
  - properties, 226
  - templates, 226-227
  - WizardStep control, 227

Role Manager, 1246-1247

- AuthorizationStoreRoleProvider, 1254-1256
- caching roles in cookies, 1256-1258
- Roles class, 1258
- SqlRoleProvider, 1247-1248, 1250-1252
- WindowsTokenRoleProvider, 1252-1254

RoleExists( ) method, 1258

RoleGroup, 1203

roleManager configuration section, 1516

roles

- authorization by role, 1223
- LoginView control, 1201, 1203
- Role Manager, 1246-1247
  - AuthorizationStoreRoleProvider, 1254-1256
  - caching roles in cookies, 1256-1258
  - Roles class, 1258
  - SqlRoleProvider, 1247-1248, 1250-1252
  - WindowsTokenRoleProvider, 1252-1254

Roles class, 1258

Roles property (SiteMapNode class), 1093

Root value, 1054

RootNode property

- SiteMap class, 1090
- SiteMapNode class, 1093

RootNodeStyle object, 1016, 1074

RootNodeTemplate, 1018

RootSectionGroup property (Configuration class), 1537

RoundedCorners control, 1739

route expressions, 1139

How can we make this index more useful? Email us at indexes@samspublishing.com

RouteData dictionary, 1141, 1144

routes

default parameters, 1143-1144

security, 1146

RouteUrlExpressionBuilder class, 1143

RouteValueDictionary, 1145

routing URL (uniform resource locator), 1137-1139, 1143-1146

linking to other pages with routes, 1142-1143

mapping, 1139-1142

scenarios, 1139

RowCancelingEdit event, 547

RowCommand event, 547

RowCreated event, 547

RowDataBound event, 547

RowDeleted event, 547

RowDeleting event, 547

RowFilter property, 906

RowIndex property (GridViewRow object), 550

rows

DataList control, 648

DataRows

selecting, 898, 901

states and versions, 901, 903, 906

filtering, 395-397

highlighting, GridView control, 547, 549-550

selecting, 669, 671-672, 965

Rows property (TextBox control), 70

RowState property (GridViewRow object), 550

RowStateFilter property, 906

RowStyle property (DetailsView control), 601

RowType property (GridViewRow object), 550

RowUpdated event, 547

RowUpdating event, 547

ROW_NUMBER( ) function, 804

RsaProtectedConfigurationProvider, 1557, 1561-1562, 1564

RSSHandler.cs file, 1479

Runtime. *See* HTTP Runtime

S

SatisfiesExpression( ) method, 995

SatisfiesRequired( ) method, 995

SatisfiesType( ) method, 995

Save( ) method, 993, 1537, 1539, 1541

SaveAs( ) method, 180, 1537, 1539

SaveChanges( ) method, 1006

SaveControlState( ) method, 1601

SaveProduct( ) method, 735, 743

saving

files

to database, 185, 188-190, 193

to filesystem, 181, 184

Profiles automatically, 1317, 1320

SayMessage( ) method, 724, 726-727

Scaffold view, 1496

Scale animation, 1763-1764

ScollDownImageUrl property (Menu control), 1040

Script Action, 1764

script tag, 8

ScriptManager.IsInAsyncPostBack, 1700

Scripts folder, 1504

ScriptService attribute, 1744

ScrollBars property (Panel control), 114-118

ScrollDownText property (Menu control), 1041

ScrollUpImageUrl property (Menu control), 1041

ScrollUpText property (Menu control), 1041

SDK Command Prompt, 1401

SeaDragon control, 1739

Search button, 955

Search Engine Optimization (SEO), 1138

SearchExpression, querying, 716-717

SecondHalf.cs file, 738

SecretFiles folder, 1149

SecretFiles\Secret.aspx page, 1148

SectionGroups property (Configuration class), 1537

SectionProducts.master file, 247

Sections property (Configuration class), 1537

SectionServices.master file, 248-249

Secure property (HttpCookie class), 1272

security, 1205-1206

ASP.NET Membership, 1226

ActiveDirectoryMembershipProvider, 1239, 1242-1244

custom membership providers, 1244-1246

Membership class, 1227-1230

MembershipUser class, 1230-1231

password policy, 1234

passwords, encrypting and hashing, 1232-1233

SqlMembershipProvider, 1237-1239

users, locking out, 1235

authorization

configuring, 1221, 1223

by file location, 1224-1225

with images and other file types, 1225

by role, 1223

connection strings, 382

cookies, 1265-1266

debugging, 50

encrypted configuration sections, 1556-1557

creating programmatically, 1558, 1561

creating with aspnet_regiis command, 1557-1558

deploying, 1561-1564

DpapiProtectedConfigurationProvider, 1557

RsaProtectedConfigurationProvider, 1557

encryption

connection strings, 388-389

email, 1166

Forms authentication, 1206-1207

across applications, 1210-1212, 1214

across domains, 1214-1216

configuring, 1207-1208

cookieless Forms authentication, 1209-1210

encryption and validation options, 1211-1212, 1214

FormsAuthentication class, 1216-1220

sliding expiration, 1210

User class, 1220-1221

.NET Passport authentication, 1206

passwords

changing with ChangePassword control, 1186-1192

encrypting and hashing, 1232-1233

generating automatically with CreateUserWizard control, 1171-1174, 1177

password policy, 1234

PasswordRecovery control, 1193-1199

Role Manager, 1246-1247

AuthorizationStoreRoleProvider, 1254-1256

caching roles in cookies, 1256-1258

Roles class, 1258

SqlRoleProvider, 1247-1252

WindowsTokenRoleProvider, 1252-1254

routes, 1146

security questions and answers, 1196

Site Map Security Trimming, 1096-1100

enabling, 1096-1097

ShowSecurityTrimming.aspx page, 1098, 1100

Web.Config file, 1096-1097

Web.sitmap file, 1097

Windows authentication, 1206

XPath Injection Attacks, 1246

Security Trimming, 1096-1098, 1100

enabling, 1096-1097

ShowSecurityTrimming.aspx page, 1098, 1100

Web.Config file, 1096-1097

Web.sitmap file, 1097

securityPolicy configuration section, 1516

select clause, 949

Select command, 428-435, 847, 851, 916, 990

Select( ) method, 435, 898, 901, 950-951, 958, 966, 992, 1461

Selectable property (MenuItem class), 1044

SelectAction property (TreeNode object), 1073

SelectByBoxOfficeTotals( ) method, 958
SelectCached( ) method, 971-972, 992
SelectCommand object, 892
SelectCount statistics, 854
SelectCountCached( ) method, 993
SelectCulture.aspx page, 1423, 1425
SelectCultureAuto.aspx page, 1431-1432
SelectCultureProfile.aspx page, 1425, 1427
SelectDataKeys.aspx page, 492-494
SelectDataList.aspx page, 649-651
Selected event, 399, 817
selected list items, determining, 446, 448-450
Selected property
  ListItem control, 441
  MenuItem class, 1044
  TreeNode object, 1073
SelectedDataKey( ) method, 491, 494
SelectedDate property (Calendar control), 197
SelectedDates property (Calendar control), 197
SelectedIndex property (List controls), 446
SelectedIndex( ) method, 491
SelectedIndexChanged event, 454, 547
SelectedIndexChanging event, 547
SelectedItem property (List controls), 446
SelectedItemStyle property (DataList control), 657
SelectedItemTemplate, 646, 662
SelectedNodeStyle object, 1074
SelectedPaged( ) method, 968
SelectedRow( ) method, 491
SelectedValue property (List controls), 446
SelectedValue( ) method, 491, 494
SelectGridView.aspx page, 489-491
SelectImageUrl property (CommandField), 530
selecting
  columns, 965
  data
    DataList control, 648-651
    GridView control, 489-491
  rows, 669, 671-672, 965
  single rows, 966
  TreeView node, 1052
Selecting event, 403, 817
SelectionChanged event, 197
SelectionChanged event handler, 199
SelectionMode property (Calendar control), 197-199
SelectMethod property, 772, 794
SelectMonthText property (Calendar control), 197
SelectMovie.aspx page, 447-448
selectors, jQuery, 1774-1776
SelectParameters collection, 783
SelectRows statistics, 854
SelectSingleNode( ) method, 1246
SelectText property (CommandField), 530
SelectWeekText property (Calendar control), 197
Send( ) method, 10, 928
SendAsync( ) method, 10
SendCompleted event, 11
SendCompleted( ) method, 10
sending email
  ChangePassword control, 1188, 1190
  CreateUserWizard control, 1166-1169
  SmtpClient class, 11
SendMail.aspx page, 11
SendResultsEnd( ) method, 928
SendResultsRow( ) method, 928
SendResultsStart( ) method, 928
SEO (Search Engine Optimization), 1138
SeparatorImageUrl property (MenuItem class), 1044
SeparatorStyle property (DataList control), 657
SeparatorTemplate templates, 628, 630, 633, 645
  banding effect, creating, 628, 630
  tab strips, creating, 630, 633
Sequence Animation, 1763
sequences, 949
serializeAs attribute, 1298, 1314
serialized shopping cart, 1314
series, 690

Server databases (SQL Server), 371-372

Server value, 1349

Server.HtmlEncode( ) static method, 948

ServerAndClient value, 1349

ServerLifecycle.aspx page, 1698

ServerRoundtrips statistics, 854

servers, Ajax

- asynchronous postbacks, 1707-1714
- debugging, 1675-1677
- execution life cycles, 1697-1707
- handling errors, 1714-1717, 1720
- JavaScript, 1694-1697
- overview of, 1674-1675
- performance, 1720, 1723-1724
- server-side versus client-side, 1675
- Timer controls, 1724-1726, 1728
- UpdatePanel control, 1677-1693
- UpdateProgress control, 1728-1729, 1731

ServerTabs control

- ServerTabs.cs file, 1626
- ShowServerTabs.aspx page, 1629-1631

ServerTabs.cs file, 1626

ServerTime component, 731

ServerTime.cs file, 732

ServerValidate event, 154

ServerValidate event handler, 154, 156

ServerValidateEventArgs class, 155

ServiceMethod property, 1742

services, references, 1002-1007

Services.aspx file, 250

Services/Default.aspx page, 1087-1088

Services/FirstService.aspx file, 1104

session cookies, 1264, 1266, 1268

Session End event, 1285, 1287

Session Start event, 1285

Session state, 1279

- adding items to, 1279-1280
- cookieless Session state, 1288-1290
- event handling, 1285-1287
- HttpSessionState class, 1284-1285
- retrieving items from, 1280-1281
- Session state stores, configuring, 1290-1291
- session timeouts, 1287-1288
- SQL Server Session state, 1294-1297
- State Server Session state, 1291-1294
- storing database data in, 1281, 1284
- user sessions, counting, 1287

Session.Abandon( ) method, 1285

SessionDataView.aspx page, 1282, 1284

SessionField property (SessionParameter object), 424

SessionGet.aspx page, 1280

SessionID property (HttpSessionState class), 1284

sessionPageState configuration section, 1516

SessionParameter object, 353, 424, 426, 787

- properties, 424
- ShowSessionParameter.aspx page, 425-426

SessionSet.aspx page, 1279

sessionState configuration section, 1517

sessionState element, 1289, 1296-1297

Set( ) function, 730

Set-Cookie header, 1264

SetActiveView( ) method, 220

SetAllowResponseInBrowserHistory( ) method, 1358

SetAuthCookie( ) method, 1217

SetCacheability( ) method, 1358

SetCookie.aspx page, 1267-1268

SetCookieValues.aspx page, 1276

SetETag( ) method, 1358

SetETagFromFileDependencies( ) method, 1358

SetExpires( ) method, 1358, 1371

SetFocus( ) method, 76

SetLastModified( ) method, 1359

SetLastModifiedFromFileDependencies( ) method, 1359

SetMaxAge( ) method, 1359

SetNoServerCaching( ) method, 1359

SetNoStore( ) method, 1359

SetNoTransform( ) method, 1359

SetOmitVaryStar( ) method, 1359
SetPersistentCookie.aspx page, 1268-1269
SetProxyMaxAge( ) method, 1359
SetRevalidation( ) method, 1359
SetSlidingExpiration( ) method, 1359, 1371
setters, 937
SettingsAllowAnonymous attribute, 1309
SetValidUntilExpires( ) method, 1359
SetVaryByCustom( ) method, 1359, 1371
shadows, rendering drop shadows, 1583-1584
shared assemblies, 15
Shared attribute (%@ OutputCache % directive), 1369
shared constructors, 734
shared fields, 733
Shared keyword, 733-734
shared methods, 726, 728
shared properties, 733
shared template changes, 1497
SharedHelloWorld component, 726-728
SharedHelloWorld.cs file, 726
SharedMovies.ascx page, 1370
sharing User Control output cache, 1370
shopping carts
  serialized shopping cart, 1314
  ShoppingCart.cs file, 1310
  ShowShoppingCart.aspx page, 1312
ShoppingCart class, 1311
ShoppingCart.cs file, 1310
ShoppingCartSection class, 1553
ShoppingCartSection.cs file, 1551-1552
ShowAbsoluteExpiration.aspx page, 1391-1392
ShowAcceptLanguages.aspx page, 1430
ShowAdminUsersSection.aspx page, 1556
ShowAdvancedCheckBox.aspx page, 1616
ShowAdventureMovies.aspx page, 827
ShowAJAX.aspx page, 578-579
ShowAjaxValidator.aspx page, 174
ShowAlert.aspx page, 1695
ShowAlertUpdatePanel.aspx page, 1696
ShowAllStatistics.aspx page, 855-856
ShowAnimationComposite.aspx page, 1761
ShowAnimationSimple.aspx page, 1759
ShowAnonymousIdentification.aspx page, 1304, 1306
ShowAppRegister.aspx file, 303
ShowAppSettings.aspx page, 1521-1522
ShowAppStartTime.aspx page, 1523
ShowArticle.aspx page, 1650
ShowArticleWithDefault.aspx page, 1655
ShowAutoComplete.aspx page, 73-74
ShowBetterProfileProvider.aspx page, 1329
ShowBoundField.aspx page, 524
ShowBulletedList.aspx page, 470-471
ShowBulletedListHyperLinks.aspx page, 474
ShowBulletedListImage.aspx page, 472-473
ShowButtonClick.aspx page, 22-23
ShowButtonField.aspx page, 531, 533
ShowCalendar.aspx page, 1433
ShowCancelButton property (CommandField), 530
ShowChangePassword.aspx page, 1186, 1188
ShowChart.aspx page, 691
ShowChartTitles.aspx page, 707
ShowCheckBox property (TreeNode object), 1073
ShowCheckBox.aspx page, 76-77
ShowCheckBoxes property (TreeView control), 1070
ShowCheckBoxField.aspx page, 526-527
ShowCheckBoxList.aspx page, 468-469
ShowColorTable.aspx page, 1583
ShowCommandField.aspx page, 528-529
ShowCompareValues.aspx page, 150
ShowConfigApp.aspx page, 1524-1525
ShowConfigContents.aspx page, 1538
ShowConfigMachine.aspx page, 1529-1530
ShowConfigModify.aspx page, 1540
ShowConfigPath.aspx page, 1528
ShowConfigRelative.aspx page, 1526-1527
ShowConfigRemote.aspx page, 1534

ShowConfigSites.aspx page, 1531

ShowConflictDetection.aspx page, 829-830

ShowContentRotator.aspx page, 1620-1621

ShowControlParameter.aspx page, 354-355, 410-411

ShowControlState.aspx page, 1601

ShowControlTree.aspx page, 38

ShowCreateUserWizard.aspx page, 1162-1163

ShowCustomLinkButton.aspx page, 1608

ShowCustomTextBox.aspx page, 1605

ShowCustomValidator.aspx page, 154

ShowCustomValidatorJS.aspx page, 156

ShowDataAdapterUpdate.aspx page, 893

ShowDataBuildProvider.aspx page, 1462

ShowDataLayer1.aspx page, 885

ShowDataList.aspx page, 362-363, 640, 642

ShowDataListTemplates.aspx page, 646, 648

ShowDataSet.aspx page, 909

ShowDataSourceMode.aspx page, 398

ShowDataTableProgram.aspx page, 897-898

ShowDataTableSelect.aspx page, 899

ShowDataTypeCheck.aspx page, 146

ShowDataView.aspx page, 907

ShowDayHeader property (Calendar control), 197

ShowDBDataLayer.aspx page, 926-927

ShowDeclarative.ascx file, 305

ShowDelete.aspx page, 597-598

ShowDeleteButton property (CommandField), 530

ShowDeleteButtonField.aspx file, 562-563

ShowDesignSection.aspx page, 1550

ShowDetailsView.aspx page, 407, 567, 569

ShowDetailsViewExplicit.aspx page, 408-409

ShowDisableValidation.aspx page, 138

ShowDivView.aspx page, 1670-1671

ShowDragPanel.aspx page, 1751

ShowDropDownList.aspx page, 459-460

ShowDSPaging.aspx page, 798-799

ShowDSSorting.aspx page, 808-809

ShowDynamicCSS.aspx file, 295, 297

ShowDynamicSkin.aspx file, 294-295

ShowDynamicUserControl.aspx page, 1374-1375

ShowEditButton property (CommandField), 530

ShowEmployee.aspx page, 569-570

ShowEmptyDataTemplate.aspx page, 517-518, 574, 576

ShowEmptyDataText.aspx page, 515-516, 573-574

ShowEnableTheming.aspx file, 277-278

ShowEntityBase\App_Code\Movie.cs file, 992, 994

ShowEntityBase\InsertMovie.aspx page, 996

ShowEntityBase\SelectPagedSortedMovies.aspx page, 993

ShowError.aspx page, 48

ShowEventArgs.aspx page, 27

ShowExecuteScalar.aspx page, 874

ShowExpandCollapse property (TreeView control), 1070

ShowExplicitSave.aspx page, 1318, 1320

ShowFeaturedProduct.aspx file, 320-321

ShowFieldHelloWorld.aspx page, 728-729

ShowFields.aspx page, 571, 573

ShowFileDependency.aspx page, 1395-1396

ShowFilteredMovies.aspx page, 814-815

ShowFilteredTextBox.aspx page, 1754

ShowFilterExpression.aspx page, 395-396

ShowFinancialAnalysis.aspx page, 702

ShowFixedValue.aspx page, 148

ShowFlowDataList.aspx page, 643

ShowFonts.aspx file, 357

ShowFooter property

DataList control, 657

GridView control, 521

ShowFormParameter.aspx page, 418-419

ShowFormView.aspx page, 367-368, 605-606

ShowFormViewDeleting.aspx page, 620-621

ShowFormViewEditing.aspx page, 612-613, 615

ShowFormViewInserting.aspx page, 616-618

ShowFormViewPagerTemplate.aspx page, 610-611
ShowFormViewPaging.aspx page, 607-608
ShowFullyRenderedControl.aspx page, 1577-1578
ShowFullyRenderedWebControl.aspx page, 1579-1580
ShowGenericMethods.aspx page, 943
ShowGlobal.aspx page, 1486
ShowGlobalPage.aspx page, 1447
ShowGradientPanel.aspx page, 1639
ShowGuestbook.aspx page, 819-820
ShowHeader property
  BoundField, 526
  DataList control, 657
  GridView control, 521
ShowHelloWorld.aspx page, 723-724
ShowHierarchicalDataBound.aspx file, 346-347
ShowHyperLink.aspx page, 118-119
showImage( ) function, 627
ShowImage.aspx page, 104-105
ShowImageButton.aspx page, 89-90
ShowImageField.aspx page, 540, 542
ShowImageRotator.aspx page, 1625
ShowImageTextHandler.aspx page, 1473
ShowInitialValue.aspx page, 142
ShowInlineCommands.aspx page, 390-392
ShowInsert( ) function, 596
ShowInsert.aspx page, 593
ShowInsertButton property (CommandField), 530
ShowInsertMode.aspx page, 594, 596
ShowIsPostBack.aspx page, 47-48
ShowItemRotator.aspx page, 1622-1623
ShowItemTemplate.aspx file, 359-360
ShowLabel.aspx page, 60-61
ShowLayout.aspx file, 285, 287
ShowLengthValidator.aspx page, 169
ShowLineChart.aspx file, 693
ShowLines property (TreeView control), 1070
ShowLinkButton.aspx page, 86-87
ShowLinks.aspx file, 361-362
ShowLINQ.aspx page, 779
ShowListBox.aspx page, 464-465
ShowListControls.aspx file, 339
ShowLiteral.aspx page, 65-66
ShowLiteralMode.aspx page, 67-68
ShowLocalConnection.aspx page, 381-382
ShowLocalizeControl.aspx page, 1451
ShowLogin.aspx page, 1153
ShowLoginName.aspx page, 1185
ShowLoginStatus.aspx page, 1184
ShowLoginView.aspx page, 1200
ShowLongTextField.aspx page, 559-560
ShowLookupExpressionBuilder.aspx page, 1467-1468
ShowMARS.aspx page, 886, 888
ShowMaskedEdit.aspx page, 1756
ShowMean.aspx, 698
ShowMeanStripLine.aspx page, 700
ShowMenu.aspx page, 1078-1079
ShowMessageBox property (ValidationSummary control), 165
ShowMovie1.aspx page, 847-848
ShowMovie2.aspx page, 852
ShowMovie3.aspx, 862
ShowMovie4.aspx, 867-868
ShowMovie5.aspx, 872
ShowMovie6.aspx page, 878-879
ShowMovie7.aspx page, 880
ShowMovie8.aspx page, 891-892
ShowMovie9.aspx page, 904, 906
ShowMovieCollection.aspx, 773
ShowMovieCount.aspx page, 394-395, 869-870
ShowMovieDataReader.aspx page, 775
ShowMovieDataSet.aspx page, 777-778
ShowMovieDataSource.aspx page, 834-835
ShowMovies.aspx page, 387, 486-487, 785
ShowMoviesByCategory.aspx page, 448-450, 787
ShowMovieView.aspx page, 1635

ShowMultipleListBox.aspx page, 465, 467
ShowMultiSelectList.aspx page, 482, 484
ShowNamedSkin.aspx file, 273-274
ShowNextPrevMonth property (Calendar control), 197
ShowObjectDataSourceCaching.aspx page, 1381
ShowPageAsyncTask.aspx page, 914, 916
ShowPageControlParameter.aspx page, 413-414
ShowPageEvents.aspx page, 45-46
ShowPageMethod.aspx page, 1780
ShowPagePropertyParameter.aspx page, 839, 841
ShowPager.aspx page, 1613
ShowPagerSettings.aspx page, 580-581
ShowPagerTemplate.aspx page, 582-583
ShowPaging.aspx page, 576, 578
ShowPanel.aspx page, 111, 113
ShowPasswordRecovery.aspx page, 1193-1194
ShowPhotos.aspx file, 349-350
ShowPreferences.aspx page, 748-749
ShowPrivateBytesLimit.aspx page, 1398
ShowProduct.aspx page, 1659
ShowProfile.aspx page, 1299, 1301
ShowProfileComponent.aspx page, 1322
ShowProfileGroups.aspx page, 1302
ShowProfileParameter.aspx page, 421-422
ShowProgrammatic.ascx file, 306
ShowQueryStringParameterMaster.aspx page, 422-423
ShowQueryStringParameterDetails.aspx page, 424
ShowRadioButton.aspx page, 79
ShowRadioButtonList.aspx page, 461-462
ShowRandomDataLayer.aspx page, 933
ShowRandomImage.aspx file, 301-302
ShowRandomQuotation.ascx, 319
ShowRandomQuotation.aspx file, 319
ShowRangeValidator.aspx page, 144
ShowRegularExpressionValidator.aspx page, 152
ShowRepeater.aspx file, 624-625
ShowRepeaterPhotos.aspx file, 626-627
ShowRepeaterTable.aspx file, 628, 630
ShowRequiredFieldValidator.aspx page, 140
ShowRequiredTextBox.aspx page, 1588
ShowRoles.aspx page, 1259
ShowRSSHandler.aspx page, 1481
ShowSecurityTrimming.aspx page, 1098, 1100
ShowSelectButton property (CommandField), 530
ShowSeparatorTemplate.aspx file, 631, 633
ShowServerTabs.aspx page, 1629-1631
ShowSessionCount.aspx page, 1287
ShowSessionParameter.aspx page, 425-426
ShowSetFocusOnError.aspx page, 130
ShowSharedHelloWorld.aspx page, 727
ShowShoppingCart.aspx page, 1312
ShowShoppingList.aspx page, 488
ShowSimpleBuildProvider.aspx page, 1457-1458
ShowSimpleCSS.aspx file, 282-283
ShowSiteProfile.aspx, 1309
ShowSkin.aspx file, 271-272
ShowSkinStyleSheetTheme.aspx file, 276-277
ShowSkinTheme.aspx file, 275-276
ShowSlidingExpiration.aspx page, 1393
ShowStartingNode property (SiteMapDataSource control), 1086
ShowSummary property (ValidationSummary control), 165
ShowSummaryPopup.aspx page, 165-166
ShowTabStrip.aspx file, 309, 311
ShowTabularDataBound.aspx file, 343, 345
ShowTemplateField.aspx page, 544, 546
ShowTextBox.aspx page, 69
ShowTitle property (Calendar control), 197
ShowToolTips property (SiteMapPath), 1016
ShowTrace.aspx page, 31-32
ShowTreeView.aspx page, 1079
ShowUIPaging.aspx page, 796-797
ShowUISorting.aspx page, 806-807

ShowUpdate.aspx page, 584-585

ShowUpdatePanelAnimation.aspx page, 1765

ShowUpdateProgress.aspx page, 1729

ShowUserControlCache.aspx page, 1368-1369

ShowUsernameParameter.aspx page, 836-837

ShowUsersOnline.aspx page, 1229-1230

ShowUserSqlConnections.aspx page, 858

ShowValidatedField.aspx file, 565-566

ShowValidateEmptyText.aspx page, 158

ShowValidationErrors( ) method, 995

ShowValidationGroups.aspx page, 134

ShowValidationSummary.aspx page, 163, 165

ShowValidators.aspx page, 131

ShowViewState.aspx page, 29, 1598

ShowWebService.aspx page, 782

ShowWebServiceMethod.aspx page, 1778

ShowWindowsRoles.aspx page, 1253-1254

ShowWizard.aspx page, 228, 233

ShowWriteSubstitution.aspx page, 1364

ShowXmlDataSourceCaching.aspx page, 1383

SideBarButtonClick event, 227

SideBarTemplate, 226

signatures, 735

SignOut( ) method, 1217

Simple2\TextBox.skin file, 273

Simple3\Label.skin file, 275

Simple4\Calendar.skin file, 279

SimpleBuildProvider, 1454, 1456-1458

CodeDom, 1455

CodeSnippetCompileUnit, 1455

implementing on pages, 1457-1458

Mike class, 1456-1457

registering, 1456

SimpleBuildProvider.cs file, 1454

Web.Config file, 1456

SimpleBuildProvider.cs file, 1454

SimpleContent.aspx file, 241-242

SimpleListView.aspx page, 662

SimpleMaster.master file, 238-239

SimplePage.aspx page, 1438

SimpleStyle.css file, 281-282

Simple\TextBox.skin file, 270

simplified databinding, supporting in templated controls, 1656-1657, 1659-1660

IDataItemContainer interface, 1656

Product control example, 1657, 1659-1660

single rows, selecting, 966

Single( ) method, 951, 967

SingleLine (TextMode property), 69

SingleOrDefault( ) method, 967

Site Maps, 1011, 1083

AutoSiteMapProvider, 1106, 1109-1110

App_Code/AutoSiteMapProvider.cs file, 1106, 1109

Web.Config file, 1110

binding Menu controls to, 1027-1031

UsingMenu/MenuNavigate.aspx, 1029, 1031

UsingMenu/MenuSiteMap.aspx, 1028-1029

binding TreeView controls to, 1057-1058

custom attributes, 1103-1105

example of, 1012

Google SiteMap files, generating, 1116-1120

PublicSiteMap.ashx page, 1117, 1119

PublicSiteMap.ashx results, 1120

merging, 1101-1102

Employees/Employees.sitemap file, 1102

Web.sitemap file, 1101-1102

Security Trimming, 1096-1098, 1100

enabling, 1096-1097

ShowSecurityTrimming.aspx page, 1098, 1100

Web.Config file, 1096-1097

Web.sitmap file, 1097

SiteMap class, 1090-1093

events, 1091

methods, 1091

nodes, adding to Site Maps, 1091, 1093

properties, 1090

SiteMapDataSource control, 1084

binding Menu controls to, 1087-1088

binding TreeView controls to, 1085-1086

Default.aspx page, 1085-1086
Products/Default.aspx page, 1089
properties, 1086, 1088, 1090
Services/Default.aspx page, 1087
Web.sitemap file, 1084-1085
SiteMapNode class, 1093-1095
methods, 1093
properties, 1093
titles, displaying, 1094-1095
siteMapNode elements, 1013
SiteMapPath control, 1013
declaring in pages, 1013, 1015
formatting, 1016, 1018-1020
properties, 1015-1016
templates, 1018-1020
SqlSiteMapProvider, 1111, 1114, 1116
App_Code\SqlSiteMapProvider.cs file, 1111, 1114
Web.Config file, 1115

Site.master file, 246-247

SiteMap class, 1090-1093
events, 1091
methods, 1091
nodes, adding to Site Maps, 1091, 1093
About.asax file, 1092-1093
Global.asax file, 1091
properties, 1090

siteMap configuration section, 1517

SiteMapDataSource control, 351, 1084
binding Menu controls to, 1087-1088
binding TreeView controls to, 1085-1086
Default.aspx page, 1085-1086
Products/Default.aspx page, 1089
properties, 1086, 1088, 1090
ShowStartingNode, 1086
StartFromCurrentNode, 1086-1090
StartingNodeOffset, 1087
StartingNodeUrl, 1087-1088
Services/Default.aspx page, 1087
Web.sitemap file, 1084-1085

SiteMapNode class, 1093-1095
methods, 1093
properties, 1093
titles, displaying, 1094-1095

siteMapNode elements, 1013

SiteMapPath control, 1013
declaring in pages, 1013, 1015
formatting, 1016, 1018-1020
properties, 1015-1016
templates, 1018-1020

SiteMapProvider property (SiteMapPath), 1016

SiteMapResolve event, 1091

SiteMapResolve( ) method, 1091

SiteProfile.cs file, 1308

Size property (Parameter object), 406

SkinID property (Named Skins), 273

Skins
applying, 275-276
ShowSkinStyleSheetTheme.aspx file, 276-277
ShowSkinTheme.aspx file, 275-276
Simple3\Label.skin file, 275
applying dynamically, 293-297
ShowDynamicCSS.aspx, 295, 297
ShowDynamicSkin.aspx, 294-295
Default Skins, creating, 270
ShowSkin.aspx file, 271-272
Simple\TextBox.skin file, 270
Named Skins, creating, 273-274
ShowNamedSkin.aspx file, 273-274
Simple2\TextBox.skin file, 273
SkinID property, 273
overriding, 276-277
StyleSheetTheme attribute, 276-277

Skip( ) method, 951, 968

SkipLinkText property
Menu control, 1041
SiteMapPath, 1016
TreeView control, 1070

Slider control, 1739

SlideShow control, 1739

sliding cache expiration policy, DataSource Caching, 1378-1379

sliding expiration
Data Caching, 1393-1394
Forms authentication, 1210

slidingExpiration configuration option (Forms authentication), 1208

SlidingExpiration property (FormsAuthentication class), 1217

SlidingUserCache.ascx page, 1371

Smart Tasks, 1640, 1644

SmartImage control, 1640, 1644

smtp element (web.config file), 1169

SmtpClient class, 10, 12

SmtpClient Send( ) method, 12

sn.exe, 754

SomeFile.dll file, 750

Sort property, 906

SortChart.aspx page, 695

Sorted event, 547

SortExpression property (BoundField), 526

SortGrid.aspx page, 495

sorting
- cached Dataviews, 907
- Chart Control, 694, 696, 698
- data, GridView control, 495-501
  - AJAX, 496-497
  - AllowSorting property, 495-496
  - customized sorting interface, 497-501
- data source sorting, 808-809, 811-812
  - EmployeesDSSorting component, 810
  - GetSortedEmployees stored procedure, 811-812
  - ShowDSSorting.aspx page, 808-809
- database data, 672, 674
- DataView, 1282
- QueryExtender control, 711-719
- records, 806
- string values, 1437-1438
- user interface sorting, 805-808

Sorting event, 547

SortListView.aspx page, 673

Sort_Command( ) event handler, 103

source code, generating automatically. *See* BuildProviders

span tag, 21, 61

specific cultures, 1420

SQL
- caches, enabling, 1114
- custom LINQ to SQL base class, 991-995, 999-1000
- LINQ, 951-952
  - associations, 959-960
  - building entities, 952-959
  - LinqDataSource control, 960, 962-963
  - standard database commands, 964-990

SQL cache dependencies, 1399
- Polling SQL cache dependencies, 1399-1400
  - configuring applications for, 1402-1403
  - configuring databases for, 1400-1402
  - Data Caching, 1406-1407
  - DataSource Caching, 1405-1406
  - Page Output Caching, 1403-1404
- Push SQL cache dependencies, 1408-1409
  - configuring applications for, 1410
  - configuring databases for, 1409-1410
  - Data Caching, 1415-1416
  - DataSource Caching, 1412, 1414
  - Page Output Caching, 1411-1412

SQL DateDiff( ) function, 968

SQL Designer, 957

SQL Server
- connecting to, 380, 382-384
- SQL cache dependency, configuring, 1114

SQL Server 2005, 195

SQL Server 2008 Express, 369-370
- connecting to, 372
- Database Explorer, 370
- keywords, 373
- Local databases, 338, 373-374
- Management Studio Express, 370
- Server databases, 371-372
- SQLCMD, 370-371

SQL Server Session state, 1294-1297

SQL Server Setup Wizard, 1251

SQL statements, executing, 389, 391-392

SqlCacheDependency class, 1394

SqlCacheDependencyAdmin class, 1400

SQLCMD tool, 370-371

SqlCommand object, 859

- executing commands with parameters, 863-864
- executing commands without parameters, 860, 862
- executing stored procedures, 865-872
  - GetBoxOfficeTotals, 870
  - GetMovieCount, 868
  - Movie4 component listing, 865-866
  - Movie5 component listing, 870
  - MovieSelect, 867
  - MovieUpdate, 867
  - ShowMovie4.aspx page, 867-868
  - ShowMovie5.aspx page, 872
  - ShowMovieCount.aspx page, 869-870
- returning resultsets, 876, 878-881
- returning single values, 874, 876

SqlConnection object, 848-849

- Connection pooling, 856-858
- provider statistics, retrieving, 849-856
  - displaying in GridView control, 855-856
  - list of statistics, 854
  - Movie2 component, 850, 852
  - ResetStatistics( ) method, 853
  - RetrieveStatistics( ) method, 853

SqlConnection pooling, 856-858

SqlConnection.Close( ) method, 857

SqlConnectionBuilder class, 382

SqlConnectionStringBuilder class, 848

SqlConnectionStringBuilder.aspx page, 383-384

SqlConnectionTime statistics, 854

SqlDataAccessLayer component, 769

SqlDataAccessLayer.cs file, 765, 769

SqlDataAdapter object, 889

- batch updates, performing, 892-893, 896
- displaying data with, 890, 892
- populating DataTables with, 889-890

SqlDataSource control, 351, 379-380, 437, 487, 694, 1376

- binding DivView control to, 1670-1671
- caching database data with, 435, 437
- ControlParameter object, 410-412, 414
  - properties, 410
  - ShowControlParameter.aspx page, 410-411
  - ShowPageControlParameter.aspx page, 413-414
- CookieParameter object, 415, 417
- database command execution
  - canceling, 403, 406
  - data source mode, changing, 397-398
  - database rows, filtering, 395-397
  - error handling, 398, 400, 403
  - inline SQL statements, 389, 391-392
  - stored procedures, 392, 395
- database connections, creating, 380
  - connection strings, encrypting, 388-389
  - connection strings, storing in Web.Config file, 386-387
  - ODBC, 384
  - OLE DB, 384
  - Oracle, 384, 386
  - SQL Server, 380, 382-384
- FormParameter object, 417, 419
- Parameter object, 407-409
  - properties, 406-407
  - ShowDetailsView.aspx page, 407
  - ShowDetailsViewExplicit.aspx page, 408-409
- ProfileParameter object, 420, 422
  - properties, 420
  - ShowProfileParameter.aspx page, 421-422
  - Web.config file, 420
- QueryStringParameter object, 422-424
  - properties, 422
  - ShowQueryStringParameterMaster.aspx page, 422-423
  - ShowQueryStringParamterDetails.aspx page, 424
- SessionParameter object, 424, 426
  - properties, 424
  - ShowSessionParameter.aspx page, 425-426

SqlDataSource command execution, 426
ADO.NET parameters, 426, 428
Delete command, 428
Insert command, 428, 430
Select command, 428-435
Update command, 428

SqlDependency attribute (%@ OutputCache % directive), 1404

SqlHierarchicalDataSource control, 1011, 1077, 1081
classes, 1080-1081
ShowMenu.aspx page, 1078-1079
ShowTreeView.aspx page, 1079

SqlHierarchicalDataSourceView class, 1081

SqlHierarchicalEnumerable class, 1081

SqlMembershipProvider, 1226, 1237-1239

SqlMethods class, 968

SqlMethods.Like( ) method, 967

SqlNode class, 1081

SqlNodePropertyDescriptor class, 1081

SqlParameter object, 863

SqlPipe class, 928

SqlProfileProvider, 1301, 1327

SqlRoleProvider, 1246-1248, 1250-1252

SQLServer value, 1291

SqlSiteMapProvider, 1111, 1114, 1116
App_Code\SqlSiteMapProvider.cs file, 1111, 1114
Web.Config file, 1115

SqlUserDefinedType attribute, 920

Src attribute (%@ Register %> directive), 302

srcGuestBook_Inserting( ) event handler, 428

srcMovies_Selecting( ) event handler, 437, 1379

srcMovies_Updating( ) method, 406

standard controls, 19
Button, 84, 86
Click event, 86
client scripts, 94-95
Command event, 86, 101, 103
cross-page posts, 96, 98, 100
displaying on pages, 84
events, 86
Focus( ) method, 86
properties, 85
CheckBox, 76-77, 79
automatic postback, 78-79
CheckedChanged event, 78
displaying in forms, 76-77
Focus( ) method, 78
properties list, 78
HyperLink, 118-119
Image, 104, 106
ImageButton, 89-93
Click event, 93
client scripts, 94-95
Command event, 93, 101, 103
cross-page posts, 96, 98, 100
displaying on pages, 89-90
Focus( ) method, 93
image maps, 91-92
properties, 93
ImageMap, 106-111
Click event, 111
Focus( ) method, 111
hot spots, 106
navigation, 107
postback, 108, 110
properties, 110
Label, 60
accessibility guidelines, 65
displaying time with, 60-61
form field labels, 63, 65
formatting, 62-63
properties, 61-62
LinkButton, 86, 88-89
Click event, 89
client scripts, 94-95
Command event, 89, 101, 103
cross-page posts, 96, 98, 100
displaying on pages, 86-87
events, 89
Focus( ) method, 89
properties, 88-89

Literal, 65, 68
Mode property, 67-68
modifying browser titles with, 65-66
Panel, 111, 114, 116, 118
GroupingText property, 114-115
hiding/displaying controls with, 111, 113
properties, 114
properties list, 114
ScrollBars property, 116, 118
RadioButton, 79, 82-83
automatic postback, 82-83
CheckedChanged event, 82
displaying on pages, 79
Focus( ) method, 82
properties, 81
TextBox, 68, 70, 72, 76
AutoComplete class, 72, 74
automatic postback, 71-72
Focus( ) method, 70, 74, 76
rendering properties, 70
TextChanged event, 71
TextMode property, 68-69

standard database commands, 964-990

StandardDeviation formula, 702

Standard\App_Code\Movie.cs file, 968, 971, 973, 977, 981, 984

Standard\DeleteMovie.aspx file, 982

Standard\InsertMovie.aspx page, 974

Standard\ShowCachedMovies.aspx page, 972

Standard\ShowDynamicSort.aspx page, 986

Standard\ShowPagedMovies.aspx page, 969

Standard\TraceWriter.cs file, 989

Standard\UpdateMovie.aspx page, 978

Standard\UpdateMovieVersion.aspx page, 980

StartFromCurrentNode property (SiteMapDataSource control), 1086-1090

StartingNodeOffset property (SiteMapDataSource control), 1087

StartingNodeUrl property (SiteMapDataSource control), 1087-1088

StartNavigationTemplate, 227

StartRowIndex property, 681

state parameter, 916

State Server Session state, 1291-1294

state, maintaining, 1263-1264

cookies, 1264-1265
creating, 1266, 1268-1269
deleting, 1273-1275
HttpCookie class, 1272-1273
multi-valued cookies, 1275, 1278-1279
persistent cookies, 1264, 1268-1269
properties, 1272-1273
reading, 1269-1270, 1272
security restrictions, 1265-1266
session cookies, 1264, 1266, 1268
Profiles
accessing from components, 1321-1322
anonymous profiles, migrating, 1307-1308
anonymous users, supporting, 1303-1307
complex profile properties, 1310-1311, 1314-1316
custom Profile providers, 1327-1331
defining, 1297-1298
displaying, 1298, 1301
inheriting from custom classes, 1308-1309
Profile groups, 1301-1302
Profile provider configuration, 1326-1327
ProfileManager class, 1323, 1325
saving automatically, 1317, 1320
Session state, 1279
adding items to, 1279-1280
cookieless Session state, 1288-1290
event handling, 1285-1287
HttpSessionState class, 1284-1285
retrieving items from, 1280-1281
Session state stores, configuring, 1290-1291
session timeouts, 1287-1288
SQL Server Session state, 1294-1297
State Server Session state, 1291-1294
storing database data in, 1281, 1284
user sessions, counting, 1287

states, DataRows, 901, 903, 906. ***See also*** **Control State; View State**

StateServer value, 1291

static methods, 726

static page methods, calling, 1780-1781

StaticBottomSeparatorImageUrl property (Menu control), 1041

StaticDisplayLevels property (Menu control), 1041

StaticEnableDefaultPopOutImage property (Menu control), 1041

StaticHoverStyle object, 1041

StaticImagePopOutFormatString property (Menu control), 1041

StaticItemFormatString property (Menu control), 1041

StaticItemTemplate, 1047

StaticMenuItemStyle object, 1041

StaticMenuStyle object, 1042

StaticPopOutImageUrl property (Menu control), 1041

StaticSelectedStyle object, 1042

StaticSubMenuIndent property (Menu control), 1041

StaticTopSeparatorImageUrl property (Menu control), 1041

statistical formulas, Chart Control, 698-701, 704

statistics on providers, retrieving, 849, 851, 854, 856

- displaying in GridView control, 855-856
- list of statistics, 854
- Movie2 component, 850, 852
- ResetStatistics( ) method, 853
- RetrieveStatistics( ) method, 853

Statistics property, 698

StatisticsEnabled property (SqlConnection object), 853

Step Into option (Debug menu), 52

Step Out option (Debug menu), 52

Step Over option (Debug menu), 52

StepNavigationTemplate, 227

StepType property (WizardStep control), 227

Storage property, 954

stored procedure return values, 868

stored procedures

- CountMoviesInCategory, 392
- creating, 928
  - CREATE PROCEDURE command, 931
  - stored procedure assembly, creating, 928, 930
  - stored procedure assembly, registering, 930
- executing, 865-872
  - from ASP.NET pages, 931-932
  - data source mode, changing, 397-398
  - database rows, filtering, 395-397
  - GetBoxOfficeTotals, 870
  - GetMovieCount, 868
  - Movie4 component listing, 865-866
  - Movie5 component listing, 870
  - MovieSelect, 867
  - MovieUpdate, 867
  - ShowMovie4.aspx page, 867-868
  - ShowMovie5.aspx page, 872
  - ShowMovieCount.aspx page, 869-870
  - SqlDataSource control, 392, 395
- GetBoxOfficeTotals, 870
- GetMovieCount, 868
- GetPagedMovies, 803
- GetPagedMovies2005, 805
- GetSortedEmployees, 811-812
- MovieSelect, 867
- MovieUpdate, 867

StoreFile( ) method, 193

StoreProduct.cs file, 735

storing

- advertisements
  - AdList.xml file, 211-212
  - AdRotatorXML.aspx page, 208
  - in database tables, 212-213
  - in XML files, 208, 210-212
- database data in Session state, 1281, 1284

String value, 1314

strings

comparing and sorting, 1437-1438

concatenating, 863

connection strings

encrypting, 388-389

security, 382

storing in Web.Config file, 386-387

formatting, 1435-1436

validating length of, 154-158

StringValidator, 1548

strip lines, 700

strong names, 754

strongly typed localization expressions, 1449-1450

Style Action, 1764

Style objects

Menu control, 1041-1042

TreeView control, 1073, 1075, 1077

Style property (Label control), 62

StyleAction animation, 1763

StyleSheetTheme attribute (%@Page %> directive), 276-277, 280

SubClassTypeValidator, 1548

Subject property (MailDefinition class), 1167

SubmitChanges( ) method, 973, 976

submitting form data, 84

Button control, 84, 86

client scripts, 94-95

Command event, 101, 103

cross-page posts, 96, 98, 100

default buttons, specifying, 100-101

ImageButton control, 89-93

LinkButton control, 86, 88-89

SUBSCRIBE QUERY NOTIFICATIONS permissions, 1410

Substitution control, 1362-1363

substitution, post-cache, 1362-1364, 1366

SubstitutionControl.aspx page, 1363

Subtract( ) method, 984

SummaryColumn.aspx page, 551-552

SumResultSets statistics, 854

SupportsCaching property (ControlCachePolicy class), 1371

SupportsEvaluate property (ExpressionBuilder class), 1469

survey form

ASPNetSurvey.aspx control, 326-327

ASPSurvey.aspx control, 325-326

WebSurvey.aspx page, 322, 325

SwitchViewByID command (MultiView control), 223

SwitchViewByIndex command (MultiView control), 224

System.Collections.Generic namespace, 941

System.Data.Common namespace, 845

System.Data.Linq.dll assembly, 952

System.Data.Linq.Mapping namespace, 953

System.Data.Linq.SqlClient.SqlMethods.Like( ) method, 967

System.Data.Odbc namespace, 844

System.Data.OleDb namespace, 844

System.Data.OracleClient namespace, 844

System.Data.Services.Client.DataServiceContext class, 1004

System.Data.SqlClient namespace, 844-845

System.Data.SqlServerCe namespace, 844

System.Design.dll assembly, 1639

System.DirectoryServices namespace, 1533

System.DirectoryServices.dll assembly, 1533

System.Drawing.Color value, 1632

System.IO namespace, 12

System.LINQ.Enumerable class, 950-951

System.Linq.Expressions.Expression class, 984

System.Linq.Queryable class, 964

System.Messaging namespace, 16

System.Net.Mail namespace, 13

System.Web.UI.Control class, 1576

System.Web.UI.WebControls.CompositeControl class, 1576

System.Web.UI.WebControls.WebControl class, 1576

systeminfo.xml file, 1567

How can we make this index more useful? Email us at indexes@samspublishing.com

T

tab strips, creating with SeparatorTemplate, 630, 633

tabbed menus, 1024, 1027

tabbed page views, displaying, 220, 223

TabClick event handler, 311

TabIndex property

- Button control, 85
- CheckBox control, 78
- ImageButton control, 93
- ImageMap control, 111
- LinkButton control, 89
- RadioButton control, 81
- TextBox control, 70

Table attribute, 954

TableListView.aspx page, 664

tables

- DataTables, 896
  - building programmatically, 896, 898
  - DataRow states and versions, 901, 903, 906
  - DataRows, selecting, 898, 901
  - populating with SqlDataAdapter object, 889-890
- drop shadows, rendering, 1583-1584
- HTML color table, rendering, 1581-1582
- multi-column tables, creating with DataList control, 644-645
- records, joining, 970
- rendering calendars from, 202, 206-207
- saving documents to, 185, 188-189
  - FileHandler.aspx page, 188-189
  - FileUploadDatabase.aspx page, 185, 187
- storing advertisements in, 212-213
- viewing, 666

Tabs control, 1739

TabStrip control

- ShowTabStrip.ascx, 309, 311
- TabClick event handler, 311
- TabStrip.ascx, 307, 309

tabular DataBound controls, 341-342, 345

- DataList, 341, 640
  - displaying data with, 640, 642-643
  - editing data with, 651, 656
  - formatting data with, 657, 659
  - multi-column tables, creating, 644-645
  - properties, 657
  - selecting data with, 648-651
  - templates, 645-646, 648
- DetailsView, 342, 567
  - deleting data with, 597-598
  - displaying data with, 567, 569-571
  - displaying empty data with, 573-576
  - events, 598-599, 601
  - fields, 571, 573
  - formatting, 601-604
  - inserting data with, 593-596
  - paging through data with, 576-583
  - ShowDetailsView.aspx page, 567, 569
  - ShowEmployee.aspx page, 569-570
  - updating data with, 584-585, 588, 591, 593
- FormView, 342, 604
  - deleting data with, 619, 621-622
  - displaying data with, 605-606
  - editing data with, 612-616
  - inserting data with, 616-617, 619
  - paging through data with, 607-609, 612
- GridView, 341, 485
  - BoundField class, 523-524, 526
  - ButtonField class, 530, 533
  - CheckBoxField class, 526-527
  - column summaries, 550, 552
  - CommandField class, 527, 529-530
  - concurrency issues, 512-514
  - data keys, 492-495
  - DeleteButtonField, 560, 562-563
  - displaying data with, 486-487, 489
  - displaying empty data, 514, 516, 518
  - editing data with, 510-512
  - events, 546-547
  - formatting, 519-521
  - highlighting rows, 547, 549-550

HyperLinkField class, 534-539
ImageField class, 539, 543
LongTextField, 556, 559-560
nested Master/Details forms, 552, 555
paging through data, 501-509
selecting data with, 489-491
sorting data with, 495-501
TemplateField class, 543, 545-546
ValidatedField, 563-566
ViewState, 521-522
Repeater, 342, 623
displaying data with, 624-625, 627
events, 633, 640
templates, 627-633
ShowTabularDataBound.aspx page, 343, 345

TagKey property, 481, 1584, 1586-1587

TagName attribute (%@ Register %> directive), 302

TagName property, 1586

TagPrefix attribute (%@ Register %> directive), 302

tags. *See names of specific tags*

Take( ) method, 951, 968

Target property
AdRotator control, 207
HyperLink control, 119
HyperLinkField, 539
ImageMap control, 111
Menu control, 1041
MenuItem class, 1044
TreeNode object, 1073
TreeView control, 1070

TargetControl property (PostBackOptions class), 1614

TargetControlID property, 1750, 1758

TelligentSystems, 804

TemplateContainer attribute, 1649

TemplateControl class, 301

templated controls, 1647-1648
default templates, creating, 1652-1655
ITemplate interface, implementing, 1648-1652
simplified databinding, supporting, 1656-1660
IDataItemContainer interface, 1656
Product control example, 1657-1660
templated databound controls, 1666
base classes, 1667
DivView, 1667, 1670-1671
two-way databinding, supporting, 1660, 1663, 1666
IBindableTemplate interface, 1660
ProductForm control example, 1660, 1663, 1666

TemplateEdit.aspx page, 589

TemplateField class, 523, 543, 545-546

TemplateField fields, 571

TemplatePagerField, 682

templates
ChangePassword control, 1190, 1192
CreateUserWizard control, 1177-1181
DataBound controls, 359-360, 362
DataList control, 645-646, 648
AlternatingItemTemplate, 645
EditItemTemplate, 646
FooterTemplate, 646
HeaderTemplate, 646
ItemTemplate, 645
SelectedItemTemplate, 646
SeparatorTemplate, 645
ShowDataListTemplates.aspx page, 646, 648
default, 1656
dynamic data applications, 1496-1497, 1500
editing data with, 588, 590-591
EmptyDataTemplate, 518
entities, 1497, 1500
Login control, 1157-1160
Menu control, 1046, 1049
PasswordRecovery control, 1197, 1199
Repeater control, 627-633
AlternatingItemTemplate, 627
FooterTemplate, 628
HeaderTemplate, 628

ItemTemplate, 627
SeparatorTemplate, 628, 630, 633
SiteMapPath control, 1018-1020
templated controls, 1647-1648
default templates, creating, 1652-1655
ITemplate interface, implementing, 1648-1652
simplified databinding, supporting, 1656-1660
two-way databinding, supporting, 1660, 1663, 1666
templated databound controls, 1666
base classes, 1667
DivView, 1667, 1670-1671
types, 1497, 1500
Wizard control, 226-227

testing cookieless sessions, 1290

text fields, creating long text fields, 556, 559-560
LongTextField.cs file, 556, 559
ShowLongTextField.aspx page, 559-560

Text property
Button control, 86
ButtonField, 533
CheckBox control, 78
CheckBoxField, 527
CompareValidator control, 146
CustomValidator control, 154
HyperLink control, 119
HyperLinkField, 539
LinkButton control, 89
ListItem control, 441
RadioButton control, 81
RangeValidator control, 144
RequiredFieldValidator control, 139

text, displaying
Label control, 60
accessibility guidelines, 65
displaying time with, 60-61
form field labels, 63, 65
formatting, 62-63
properties, 61-62
Literal control, 65, 68
Mode property, 67-68
modifying browser titles with, 65-66

TextAlign property
CheckBox control, 78
RadioButton control, 81

TextBox control, 68, 70, 72, 76
AutoComplete class, 72, 74
automatic postback, 71-72
Default Skins
ShowSkin.aspx file, 271-272
Simple\TextBox.skin file, 270
Focus( ) method, 70, 74, 76
Named Skins
ShowNamedSkin.aspx file, 273-274
Simple2\TextBox.skin file, 273
rendering properties, 70
TextChanged event, 71
TextMode property, 68-69

TextBox.skin file, 270, 273

TextBoxAutoPostBack.aspx page, 71-72

TextBoxFocus.aspx page, 75-76

TextBoxWatermark control, 1739

TextChanged event, 71

TextMode property (TextBox control), 68-69

TEXTPTR( ) function, 195

TextWriter, 988

Themes, 269
applying dynamically, 291, 293
DynamicTheme.aspx file, 291, 293
Web.Config file, 293
App_Themes folder, 270
Cascading Style Sheets
applying, 281-284
changing page layouts with, 285-290
multiple Cascading Style Sheets, 285
compared to Master Pages, 269
creating, 270
disabling, 277, 279
ShowEnableTheming.aspx file, 277-278
Simple4\Calendar.skin file, 279
global Themes, 290-291

naming, 270
registering in web configuration file, 280-281
Skins
applying, 275-277
applying dynamically, 293-295, 297
Default Skins, creating, 270-272
Named Skins, creating, 273-274
overriding, 276-277
ShowNamedSkin.aspx file, 273-274
Simple2\TextBox.skin file, 273
SkinID property, 273
StyleSheetTheme attribute, 276-277

ThenBy tag, 714

ThenBy( ) method, 966

ThenByDescending( ) method, 966

third-party cookies. *See* cookies

three-dimensional charts, 708

tilde character (~), 1124

time, displaying, 60-61

timed tests, creating, 160

TimedTest.aspx page, 160

timeout configuration option (Forms authentication), 1208

Timeout property
HttpSessionState class, 1284
Session object, 1288

TimeoutData( ) method, 916-917

timeouts (sessions), 1287-1288

Timer controls, Ajax, 1724-1726, 1728

TimerMessages.aspx page, 1727

TimerPage.aspx page, 1724

TimerQuote.aspx page, 1725

TimeService
binding to ObjectDataSource control, 782
TimeService.asmx file, 780

TimeService.asmx file, 780

TimeSpanValidator, 1548

timoutHandler parameter, 916

Title attribute (%@ Page %> directive), 255-256, 1013

@Title parameter, 512

Title property, 977
SiteMapNode class, 1093
WizardStep control, 227

TitleContent.aspx file, 255-256

TitleFormat property (Calendar control), 197

titles
charts, adding to, 706
Master Pages, 255-256
modifying with Literal control, 65-66

ToAbsolute( ) method, 1135

ToAppRelative( ) method, 1135

TodaysDate property (Calendar control), 197

ToggleButton control, 1739

ToList( ) method, 972

ToolboxData design-time attribute, 1631

ToolboxItem design-time attribute, 1632

tools
AJAX Control Toolkit, 1734, 1736
Animation control, 1759-1764
AutoComplete control, 1739-1750
controls, 1737-1739
DragPanel control, 1750-1751, 1754
FilteredtextBox control, 1754, 1756
MaskedEdit control, 1756, 1758-1759
UpdatePanelAnimation control, 1764-1768
dsmgmt.exe, 1241
enableNotifications, 1402
Fiddler, 1336, 1675-1676
Firebug, 1676
ORM, 711
Web Site Administration Tool, 1149, 1207, 1247

ToolTip property
Label control, 62
TreeNode object, 1073

ToString( ) method, 66, 923, 1435

ToStringCulture.aspx page, 1435-1436

TotalRowCount property, 681

trace configuration section, 1517

Trace TextWriter, 988

Trace.axd page, 54

How can we make this index more useful? Email us at indexes@samspublishing.com

Trace.Warn( ) method, 917

TraceWriter file, 991

tracing page execution, 30, 32, 52-55

TrackFocus property (PostBackOptions class), 1614

tracking

- advertisement impressions and transfers, 213-219
  - AdHandler.ashx page, 216
  - AdRotatorStats.aspx page, 218
  - AdRotatorTrack.aspx page, 214
- data row changes, 904

Transactions statistics, 854

transfers (ads), tracking, 213-219

- AdHandler.ashx page, 216
- AdRotatorStats.aspx page, 218
- AdRotatorTrack.aspx page, 214

Transform value (Mode property), 67

transformations, Web.config files, 1569-1571

transition filters, 627

TreeNode object properties, 1073

TreeView control, 346, 1011, 1049

- adding nodes to, 1049, 1051, 1054
  - TreeViewDeclare.aspx page, 1050-1051
  - TreeViewValue.aspx page, 1052, 1054
- AJAX, 1066-1067, 1070
- binding to database data, 1061, 1064
- binding to Site Maps, 1057-1058
- binding to SiteMapDataSource controls, 1085-1086
- binding to XML files, 1058-1059, 1061
  - Movies.xml file, 1058
  - MoviesComplex.xml file, 1059
  - TreeViewXml.aspx page, 1059
  - TreeViewXMLComplex.aspx page, 1060
- check boxes, 1054, 1057
- formatting, 1070-1077
  - general properties, 1070
  - Style objects, 1073-1077
  - TreeNode object properties, 1073
  - TreeViewImageSet.aspx page, 1071, 1073
  - TreeViewLevelStyles.aspx page, 1076-1077
  - TreeViewStyles.aspx page, 1074-1075
- Populate On Demand feature, 1064, 1066

TreeViewAJAX.aspx page, 1067

TreeViewCheckBoxes.aspx page, 1055-1056

TreeViewDatabase.aspx page, 1062

TreeViewDeclare.aspx page, 1050-1051

TreeViewImageSet.aspx page, 1072-1073

TreeViewLevelStyles.aspx page, 1076-1077

TreeViewPopulateOnDemand.aspx page, 1065-1066

TreeViewStyles.aspx page, 1074-1075

TreeViewValue.aspx page, 1052, 1054

TreeViewXml.aspx page, 1059

TreeViewXMLComplex.aspx page, 1060

triggers, UpdatePanel control, 1686-1687

Triggers property, 1686

TriggerUpdatePanel.aspx page, 1686

troubleshooting

- AJAX Control Toolkit, 1734, 1736
  - Animation control, 1759-1764
  - AutoComplete control, 1739-1750
  - controls, 1737-1739
  - DragPanel control, 1750-1751, 1754
  - FilteredTextBox control, 1754, 1756
  - MaskedEdit control, 1756-1759
  - UpdatePanelAnimation control, 1764-1768
- command execution errors, 398, 400, 403
- error messages, displaying summary of, 163, 165
- method errors, 824
- UpdatePanel controls, 1714-1724

trust configuration section, 1517

Try…Catch statements, 849

two-way databinding, supporting in templated controls, 1660, 1663, 1666

- IBindableTemplate interface, 1660
- ProductForm control example, 1660, 1663, 1666

txtEdit_DataBinding( ) method, 559

type attribute, 1298, 1475

type inference, 939-940

Type property

- CompareValidator control, 146
- Parameter object, 407
- RangeValidator control, 144

TypeConverter design-time attribute, 1632

TypeName property (ObjectDataSource control), 772

types

- anonymous, 940
- of charts, 690, 694
- templates, 1497, 1500
- of validation, 995

U

UICulture property, 1420-1421

Unchanged value, 901

underscore character (_), 730

uniform resource locators. *See* URLs

UnlockUser( ) method, 1231

UnpreparedExecs statistics, 854

Update command, 428, 978

Update( ) method, 892, 1686

UpdateCheck property, 954

UpdateCommand object, 892

Updated event, 399, 817

UpdateEmployee( ) method, 794

UpdateEmployees.aspx page, 793

UpdateImageUrl property (CommandField), 530

UpdateMethod property (ObjectDataSource control), 772

UpdateMode property, 1686

UpdateMovie( ) method, 832

UpdatePanel control, Ajax, 497, 1677, 1679, 1681-1682, 1685

- asynchronous postbacks, 1707-1714
- execution life cycles, 1697-1707
- handling errors, 1714-1717, 1720
- JavaScript, 1694-1697
- nesting, 1688, 1692
- performance, 1720, 1723-1724
- triggers, 1686-1687
- updating, 1692-1693

UpdatePanelAbort.aspx page, 1710

UpdatePanelAnimation control, 1739, 1764-1768

UpdatePanelCancel.aspx page, 1708

UpdatePanelCustomProgress.aspx page, 1704

UpdatePanelDataItem.aspx page, 1712

UpdatePanelError.aspx page, 1715

UpdatePanelErrorClient.aspx page, 1718

UpdatePanelErrorServer.aspx page, 1716

UpdatePanelPrecedence.aspx page, 1709

UpdatePanelSimple.aspx page, 1678

UpdateParameters collection, 783

UpdateProgress control, Ajax, 1728-1729, 1731

updates, performing batch updates with SqlDataAdapter object, 892-893, 896

UpdateText property (CommandField), 530

UpdateUpdatePanel.aspx page, 1693

UpdateUser( ) method, 1227

updating

- entities, 1006
- LINQ to SQL, 976, 978-980
- records, 860, 862
- UpdatePanel control, 1692-1693

updating data, DetailsView control, 584-593

- concurrency issues, 591-593
- MasterDetailEdit.aspx page, 586, 588
- ShowUpdate.aspx page, 584-585
- templates, 588, 590-591

Updating event, 403, 817

uploading

- images, FileUpload control, 181, 184
- large files, 189-190, 193, 195
  - buffering thresholds, 189
  - FileUploadLarge.ashx page, 194
  - FileUploadLarge.aspx page, 190, 193

url attribute, 1013

Url property (SiteMapNode class), 1093

How can we make this index more useful? Email us at indexes@samspublishing.com

urlMappings configuration section, 1517

urlMappings element (web.config file), 1124

UrlMappings.config file, 1129

UrlRemapper module, 1127, 1129-1130, 1132
- IHttpModule interface, 1129
- Products/Default.aspx page, 1130, 1132
- UrlMappings.config file, 1129
- UrlRemapper.cs file, 1127, 1129
- web configuration file, 1129

UrlRemapper.cs file, 1127, 1129

URLs (uniform resource locators)
- current URL, determining, 1125
- relative, 252
- remapping
  - web configuration files, 1124, 1126
  - wildcard matches (UrlRemapper module), 1127-1132
- routing, 1137-1139, 1143-1146
  - linking to other pages with routes, 1142-1143
  - mapping, 1139-1142
  - scenarios, 1139

UseAccessibleHeader property (DataList control), 657

UseCookies, 1209, 1289

UseDeviceProfile, 1209, 1289

User class, 1220-1221

User controls, 302
- AddressForm control, 311, 315, 317
  - AddressForm.ascx, 311, 315
  - Checkout.ascx example, 315, 317
- AJAX (Asynchronous JavaScript and XML), 317, 320
- ASPNetSurvey control, 326-327
- ASPSurvey control, 325-326
- creating, 300-302
- dynamically loaded User Controls, caching, 1373, 1375
- exposing events from, 307, 309, 311
- exposing properties from, 304-306
  - PropertyRandomImage.ascx file, 304-305
  - ShowDeclarative.ascx file, 305
  - ShowProgrammatic.ascx file, 306
- loading dynamically, 320-321
  - %@ Reference %> directive, 322, 325, 327
  - LoadControl( ) method, 320
  - multi-page wizards, creating, 327-331, 336
  - ShowFeaturedProduct.aspx example, 320-321
- Partial Page Caching, 1366-1369
- PropertyRandomImage control, 304
- RandomImage example
  - RandomImage.ascx, 300-301
  - ShowRandomImage.ascx, 301-302
- RandomQuotation control
  - RandomQuotation.ascx, 318-319
  - ShowRandomQuotation.ascx, 319
- registering in web configuration file, 303
- TabStrip control
  - ShowTabStrip.ascx, 309, 311
  - TabClick event handler, 311
  - TabStrip.ascx, 307, 309

user input, accepting, 68, 1507-1509
- CheckBox control, 76-77, 79
  - automatic postback, 78-79
  - CheckedChanged event, 78
  - displaying in forms, 76-77
  - Focus( ) method, 78
  - properties list, 78
- RadioButton control, 79, 82-83
  - automatic postback, 82-83
  - CheckedChanged event, 82
  - displaying on pages, 79
  - Focus( ) method, 82
  - properties, 81
- TextBox control, 68, 70, 72, 76
  - AutoComplete class, 72, 74
  - automatic postback, 71-72
  - Focus( ) method, 70, 74, 76
  - rendering properties, 70
  - TextChanged event, 71
  - TextMode property, 68-69

User Instance parameter, 374

User Interface layer, 758-761

user interface paging, 503, 608, 795-798

user interface sorting, 805, 807-808

user registration forms, creating. *See* CreateUserWizard control

user-defined types

creating data access layers with, 924-927

creating with .NET Framework, 919

CREATE TYPE command, 924

user-defined type assembly, creating, 920, 923

user-defined type assembly, registering, 923

userData behavior, 1266

UserName ID, 1180, 1192

username parameters, creating, 835-837

UserName property (MembershipUser class), 1231

UsernameParameter class, 835-837

UsernameParameter.cs file, 835

users

ASP.NET Membership, 1226

ActiveDirectoryMembershipProvider, 1239, 1242-1244

custom membership providers, 1244-1246

Membership class, 1227-1230

MembershipUser class, 1230-1231

password policy, 1234

passwords, encrypting and hashing, 1232-1233

SqlMembershipProvider, 1237-1239

users, locking out, 1235

Forms authentication, 1206-1207

across applications, 1210-1214

across domains, 1214-1216

configuring, 1207-1208

cookieless Forms authentication, 1209-1210

encryption and validation options, 1211-1212, 1214

FormsAuthentication class, 1216-1220

sliding expiration, 1210

User class, 1220-1221

hiding Login control from, 1155-1160

listing, 1227-1228

locking out, 1235

.NET Passport authentication, 1206

passwords. *See* passwords

redirecting

CreateUserWizard control, 1169-1170

Login control, 1155

registered usernames, displaying, 1185-1186

Windows authentication, 1206

UsersOnline login control

ShowUsersOnline.aspx page, 1229-1230

UsersOnline.cs file, 1229

UsersOnline.cs file, 1229

UseSubmitBehavior property (Button control), 86

UseUri, 1209, 1289

Using statement, 849

UsingMenu/MenuNavigate.aspx page, 1029, 1031

UsingMenu/MenuSiteMap.aspx page, 1028-1029

UsingSiteMapPath/DisplaySiteMapPath.aspx page, 1015

UsingSiteMapPath/SiteMapPathStyle.aspx page, 1017

UsingSiteMapPath/SiteMapPathTemplates.aspx page, 1019-1020

UsingTreeView/TreeViewSiteMap.aspx page, 1057-1058

UtcLastModified property (CacheDependency class), 1395

V

validate attribute, 1476

Validate( ) method, 991, 995, 1000

validated fields, creating, 563-566

ValidatedField class, 563-566

ValidateEmptyText property (ServerValidateEventArgs class), 155

How can we make this index more useful? Email us at indexes@samspublishing.com

ValidateUser( ) method, 1227

validating forms. *See* validation controls

ValidatingPassword event, 1227

validation, 994

- entities, 996
- types of, 995

validation controls, 19, 121

- BaseValidator class, 167
- CompareValidator, 121, 146, 148, 150-151
  - ControlToCompare property, 146
  - ControlToValidate property, 146
  - data type checks, performing, 146
  - form fields, comparing against another form field, 150
  - form fields, comparing against fixed values, 148
  - Operator property, 146
  - Text property, 146
  - Type property, 146
  - ValueToCompare property, 146
- custom validation controls, creating, 167
  - AjaxValidator control, 170, 173-174
  - LengthValidator control, 168-170
- CustomValidator, 122, 154
  - client-side validation functions, 156, 158
  - ClientValidationFunction property, 154
  - ControlToValidate property, 154
  - empty form fields, validating, 158-159
  - ServerValidateEventArgs class, 155
  - string lengths, checking, 154, 156, 158
  - Text property, 154
  - timed tests, creating, 160
- disabling validation, 138-139
- Display property, 127-128
- implementing on OrderForm page, 122, 124-125
- JavaScript and, 126
- Page.IsValid property, 126-127
- RangeValidator, 121, 143-145
- RegularExpressionValidator, 122, 151, 153
- RequiredFieldValidator, 121, 139
  - ControlToValidate property, 139
  - InitialValue property, 141, 143
  - ShowInitialValue.aspx page, 142
  - ShowRequiredFieldValidator.aspx page, 140
  - Text property, 139
- validation errors, highlighting, 128
  - ShowSetFocusOnError.aspx page, 130
  - ShowValidators.aspx page, 131
  - ValidationImage.aspx page, 128
- validation groups, 133, 138
- ValidationSummary, 122, 162, 166
  - DisplayMode property, 165
  - error message summary, displaying, 163, 165
  - HeaderText property, 165
  - popup alert boxes, displaying, 165-166
  - ShowMessageBox property, 165
  - ShowSummary property, 165

validation errors, highlighting, 128

- ShowSetFocusOnError.aspx page, 130
- ShowValidators.aspx page, 131
- ValidationImage.aspx page, 128

ValidationError file, 991

ValidationErrorCollection file, 991

ValidationException file, 991

ValidationExpression property, 153

ValidationGroup property

- ButtonField, 533
- CommandField, 530
- PostBackOptions class, 1614

ValidationImage.aspx page, 128

validationKey attribute, 1211, 1294, 1296

ValidationMethodName property, 920

ValidationSummary control, 122, 162, 166

- DisplayMode property, 165
- error message summary, displaying, 163, 165
- HeaderText property, 165
- popup alert boxes, displaying, 165-166
- ShowMessageBox property, 165
- ShowSummary property, 165

ValidationUtility.ShowValidationErrors( ) method, 999

ValidatorCallout control, 1739

validators, defining configuration properties, 1548

Value property

HttpCookie class, 1272

ListItem control, 441

ServerValidateEventArgs class, 155

Values property (HttpCookie class), 1272

ValueToCompare property (CompareValidator control), 146

VaryByBrowser.aspx page, 1346-1347

VaryByControl attribute (%@ OutputCache % directive), 1342, 1344, 1369

VaryByControl property (ControlCachePolicy class), 1371

VaryByCustom attribute (%@ OutputCache % directive), 1346-1347, 1349, 1369

VaryByCustom.aspx page, 1348

VaryByHeader attribute (%@ OutputCache % directive), 1345-1346

VaryByHeader.aspx page, 1345

VaryByHeaders property (HttpCachePolicy class), 1358

VaryByParam attribute (%@ OutputCache % directive), 1338, 1342, 1369

VaryByParams property

ControlCachePolicy class, 1371

HttpCachePolicy class, 1358

VB.NET, new features, 936, 938-939

anonymous types, 940

extension methods, 948-949

generics, 941-944

initializers, 937-939

lambda expressions, 944-947

LINQ, 949-951

type inference, 939-940

vbc compiler, compiling components, 750-751

verb attribute, 1475

Version property, 979

versions

DataRows, 901, 903, 906

SQL Server, 195

View Code, 41

View control, 220

View State, 28

disabling, 32-33

showing, 28, 30

supporting in custom controls, 1596

ShowViewState.aspx page, 1598

ViewStateControl.cs file, 1596

tracing, 30, 32

viewing

Chart Control, 690-694

databases, records, 664

Details page, 1341

execution time statistics, 852

line charts, 692

Master pages, 1340

Master/Details forms, 1343

mean value, 698

output parameters, 872

photo galleries, 667

provider statistics, 855

stored procedure return values, 868

tables, 666

Trace Information, 53

views

MVC Framework, 1502

tabbed page views, displaying, 220, 223

Views folder, 1504

Views property (MultiView control), 220

ViewState, GridView control, 521-522

__VIEWSTATE form field, 1599

ViewStateControl.cs file, 1596

VirtualPath property, 1134

VirtualPathProvider class, 1123, 1132

limitations, 1133

methods, 1133-1134

properties, 1134

registering, 1135

VirtualPathUtility class, 1135

VirtualPathUtility class, 1135

visibility attribute, 127

Visible property, 113

BoundField, 526

PagerSettings class, 506, 581, 609

How can we make this index more useful? Email us at indexes@samspublishing.com

VisibleDate property (Calendar control), 197

VisibleMonthChanged event, 197

Visual Basic .NET, vbc compiler, 750-751

Visual Basic .NET Express, compiling components, 751

Visual Studio .NET 2005, compiling components, 752

Visual Studio 2010, creating MVC Framework, 1502-1505

Visual Studio Command Prompt, 1294

Visual Web Developer, 40, 51-52, 241

Vote.aspx page, 415

W

W3C (World Wide Web Consortium), 1557

WAITFOR statement, 851

Warn( ) method, 54

WCF Data Services, 1001

- data context, 1007-1009
- overview of, 1001
- servie referenes, 1002-1007

web applications. ***See also*** **applications**

- databases, deploying, 1571-1572
- MVC Framework, 1501
  - accepting form input, 1507, 1509
  - controllers, 1502
  - creating, 1502, 1504-1505
  - models, 1502
  - pages, 1505-1507
  - views, 1502
- one-click, 1572-1573
- packaging, 1565-1566, 1568
- Web.fig file transformations, 1569-1571

web configuration files, 13, 1096, 1512

- ActiveDirectoryMembershipProvider configuration, 1242
- anonymous HTTP Handlers, registering, 1480
- appSettings element, 1520, 1522-1523
- assemblies, referencing, 17
- authorization, configuring, 1222
  - authorization by role, 1223
  - password protection, 1224
- AuthorizationStoreRoleProvider configuration, 1255
- buffering thresholds, setting, 189
- cache element, 1397
- cacheRolesInCookie attribute, 1257
- compilation element, 1520
- configuration element collections, creating, 1552, 1554
- configuration sections. *See* ASP.NET configuration sections
- configuration settings
  - applying to particular paths, 1517-1518
  - locking, 1518, 1520
  - placing in external files, 1522-1523
- cultures, setting, 1427-1428, 1432-1433
- custom application settings, 1520
- custom configuration sections, registering, 1549
- custom HTTP Modules, registering, 1488
- custom Site Map providers, configuring, 1110, 1115
- encrypted web configuration files, deploying, 1561-1564
- debugging, enabling, 50
- expressionBuilders section, 1467
- ExpressionBuilders, registering, 1467
- FileUploadLarge.ashx page, 194
- FileUploadLarge.aspx page, 190, 193
- Forms authentication
  - authentication cookies, 1208
  - AutoDetect, 1210
  - enableCrossAppRedirects attribute, 1214
  - enabling, 1148-1149, 1207
  - encryption and validation options, 1211, 1214
  - machineKey element, 1211-1212, 1214
  - sliding expiration, 1210
- httpHandlers section, 1475, 1480
- httpModules section, 1129, 1488
- Image HTTP Handler, registering, 1475-1476

impersonation, enabling, 1542
location element, 1517-1518
maxInvalidPasswordAttempts attribute, 1235
minRequiredNonAlphanumericCharacters attribute, 1234
namespaces, recognizing, 13
passwordAttemptWindow attribute, 1235
passwordFormat attribute, 1232
Profiles
  accessing from components, 1321
  anonymous user support, 1303
  autosave, 1317
  complex profile properties, 1311, 1316
  custom Profile providers, 1328
  defining, 1297, 1301
  inheriting from custom classes, 1308
  Profile provider configuration, 1326
registering Master Pages in, 254-255
registering Themes in, 280-281
  DisablePageTheme.aspx example, 280-281
  Web.Config file example, 280
remapping URLs in, 1124, 1126
  current URL, determining, 1125
  Default.aspx page, 1125-1126
  urlMappings element, 1124
  Web.Config code listing, 1124
Security Trimming, enabling, 1096-1097
Session timeout value, 1288
sessionState element, 1289
SimpleBuildProvider, registering, 1456
smtp element, 1169
SQL cache dependency, configuring, 1114
SQL Server Session state configuration, 1295-1296
SqlRoleProvider configuration, 1247-1251
State Server Session state configuration, 1292
storing connection strings in, 386-387
tracing, enabling, 54
WindowsTokenRoleProvider configuration, 1252
XmlMembershipProvider configuration, 1245

Web Deployment Package Import, 1568

web forms, 9, 1501

Web Services, binding to ObjectDataSource control, 779-780, 783

Web Site Administration Tool, 1149, 1207, 1247, 1513-1514

Web Site menu commands, Add New Item, 5

Web.Config file, 266-267, 725, 756, 1004, 1512
  ActiveDirectoryMembershipProvider configuration, 1242
  anonymous HTTP Handlers, registering, 1480
  appSettings element, 1520, 1522-1523
  assemblies, referencing, 17
  authorization, configuring, 1222
    authorization by role, 1223
    password protection, 1224
  AuthorizationStoreRoleProvider configuration, 1255
  buffering thresholds, setting, 189
  cache element, 1397
  cacheRolesInCookie attribute, 1257
  compilation element, 1520
  configuration element collections, creating, 1552, 1554
  configuration sections. *See* ASP.NET configuration sections
  configuration settings
    applying to particular paths, 1517-1518
    locking, 1518, 1520
    placing in external files, 1522-1523
  cultures, setting, 1427-1428, 1432-1433
  custom application settings, 1520
  custom configuration sections, registering, 1549
  custom HTTP Modules, registering, 1488
  custom Site Map providers, configuring, 1110, 1115
  debugging, enabling, 50
  dynamic data applications, 1495
  encrypted web configuration files, deploying, 1561-1564
  expressionBuilders section, 1467

ExpressionBuilders, registering, 1467
FileUploadLarge.ashx page, 194
FileUploadLarge.aspx page, 190, 193
Forms authentication
  authentication cookies, 1208
  AutoDetect, 1210
  enableCrossAppRedirects attribute, 1214
  enabling, 1148-1149, 1207
  encryption and validation options, 1211, 1214
  machineKey element, 1211-1212, 1214
  sliding expiration, 1210
httpHandlers section, 1475, 1480
httpModules element, 1129
httpModules section, 1488
Image HTTP Handler, registering, 1475-1476
impersonation, enabling, 1542
location element, 1517-1518
maxInvalidPasswordAttempts attribute, 1235
minRequiredNonAlphanumericCharacters attribute, 1234
namespaces, recognizing, 13
passwordAttemptWindow attribute, 1235
passwordFormat attribute, 1232
Profiles
  accessing from components, 1321
  anonymous user support, 1303
  autosave, 1317
  complex profile properties, 1311, 1316
  custom Profile providers, 1328
  defining, 1297, 1301
  inheriting from custom classes, 1308
  Profile provider configuration, 1326
registering Themes in, 280-281
  DisablePageTheme.aspx example, 280-281
  Web.Config file example, 280
registering User controls in, 303
remapping URLs in, 1124, 1126
  current URL, determining, 1125
  Default.aspx page, 1125-1126
    urlMappings element, 1124
    Web.Config code listing, 1124
Security Trimming, enabling, 1096-1097
Session timeout value, 1288
sessionState element, 1289
SimpleBuildProvider, registering, 1456
smtp element, 1169
SQL cache dependency, configuring, 1114
SQL Server Session state configuration, 1295-1296
SqlRoleProvider configuration, 1247-1251
State Server Session state configuration, 1292
storing connection strings in, 386-387
tracing, enabling, 54
transformations, 1569-1571
WindowsTokenRoleProvider configuration, 1252
XmlMembershipProvider configuration, 1245

Web.config.comments file, 1512

Web.config.default file, 1512

Web.sitemap file, 1012, 1084-1085, 1097, 1101-1102

WebConfigurationManager class, 1523-1524

WebControl class, 1576-1579
  TagKey property, 1584-1587

webControls configuration section, 1517

WebForm_DoCallback( ) method, 174

webParts configuration section, 1517

webServices configuration section, 1517

website configuration, 1511, 1513. *See also* Webconfig file
  ASP.NET configuration sections, 1515-1517
  configuration settings
    applying to particular paths, 1517-1518
    custom application settings, 1520, 1522
    locking, 1518, 1520
    placing in external files, 1522-1523
  Machine.config file, 1511
  Microsoft Management Console (MMC) Snap-In, 1514
  Web Site Administration Tool, 1513-1514
  Web.config file, 1512

Website menu commands, Add New Item, 722

websites. ***See also*** **navigation**
- provisioning, 1543, 1546
- remapping URLs
  - web configuration files, 1124, 1126
  - wildcard matches (UrlRemapper module), 1127-1132

WebSurvey.aspx page, 322, 325

where clause, 949

Where( ) method, 950-951, 956, 966

wildcards, 967
- matching URIs with (UrlRemapper module), 1127-1132
  - IHttpModule interface, 1129
  - Products/Default.aspx page, 1130, 1132
  - UrlMappings.config file, 1129
  - UrlRemapper.cs file, 1127, 1129
  - web configuration file, 1129

Windows authentication, 1206

Windows Forms, 1501

WindowsTokenRoleProvider, 1246, 1252-1254

Wizard control, 226
- displaying wizards with (ShowWizard.aspx), 228, 231
- events, 227
- methods, 227
- properties, 226
- templates, 226-227
- WizardStep control, 227

wizards. ***See also*** **multi-part forms, displaying**
- ASP.NET SQL Server Setup Wizard, 1238
- displaying, 228, 231
- multi-page wizards, creating, 327-331, 336
  - IWizardStep interface, 328
  - LoadStep( ) method, 328
  - LoadWizardStep( ) method, 335
  - NextStep( ) method, 328
  - Wizard.aspx, 332, 335
  - WizardSteps\Step1.ascx, 328
  - WizardSteps\Step2.ascx, 330
  - WizardSteps\Step3.ascx, 331

WizardStep control, 227

WizardSteps property (Wizard control), 226

WMI (Windows Management Instrumentation), 1543

Word documents, saving to database tables, 185, 188-189
- FileHandler.aspx page, 188-189
- FileUploadDatabase.aspx page, 185, 187

World Wide Web Consortium (W3C), 1557

Wrap property (TextBox control), 70

.WRITE clause, 195

Write( ) method, 923, 1577, 1580

WriteBreak( ) method, 1580

WriteSubstitution( ) method, 1363-1364

writeToDiagnosticsTrace attribute (trace element), 55

Wsdl.exe (Web Services Description Language) tool, 780

X–Z

XHTML, GridView control, 487

XHTML 1.0 Transitional standard, 9

xhtmlConformance configuration section, 1517

XML files
- binding Menu controls to, 1031-1034
  - Menu.xml, 1031
  - MenuComplex.xml, 1033
  - MenuXML.aspx, 1032
  - MenuXMLComplex.aspx, 1034
- binding TreeView controls to, 1058-1061
  - Movies.xml file, 1058
  - MoviesComplex.xml file, 1059
  - TreeViewXml.aspx page, 1059
  - TreeViewXMLComplex.aspx page, 1060
- exporting key containers to, 1563
- storing advertisements in, 208, 210, 212
  - AdList.xml file, 211-212
  - AdRotatorXML.aspx page, 208

Xml value, 1314

XmlDataSource control, 351, 1382-1383

XmlMembershipProvider, 1244

XPath Injection Attacks, 1246

How can we make this index more useful? Email us at indexes@samspublishing.com