

Index

Page numbers in *italics* refer to the appendixes, which are posted on www.wrox.com.

SYMBOLS

& (ampersand)

bit-wise operator, 132, 137

logical operator, 132

unary operator, 200

Windows Form label prefix, 589

&= (ampersand, equals sign) assignment operator, 132

&& (ampersands) logical operator, 132

* (asterisk)

arithmetic operator, 132

regular expression special character, 232

Visual Studio .NET autoversioning operator, 378

***= (asterisk, equals sign) assignment operator, 132**

***/ (asterisk, slash) comment suffix, 31**

@ (at sign) string literal prefix, 46

\ (backslash) escape sequence prefix, 44, 46, 232

\\ (backslashes) escape sequence prefix, 46

| (bar)

bit-wise operator, 132

logical operator, 132

|= (bar, equals sign) assignment operator, 132

|| (bars) logical operator, 132

{ } (brackets, curly) statement block constructor, 31

() (brackets, round)

cast operator, 132

method suffix, 87

[] (brackets, square)

array element index delimiters, 58, 132

reflection attribute delimiters, 260

regular expression alternative character delimiters, 233

^ (caret)

logical operator, 132

regular expression special character, 232

^= (caret, equals sign) assignment operator, 132

: (colon) `WriteLine` method format specifier prefix, 223

:: (colons) event cancellation string, 184

, (comma)

inheritance list separator, 111

`WriteLine` method character number prefix, 223

\$ (dollar sign) regular expression special character, 232

= (equals sign) assignment operator, 132, 1278

:= (equals sign, colon) assignment operator, 34, 132

== (equals signs) comparison operator, 48, 132, 143, 151, 152

!= (exclamation mark, equals sign) comparison operator, 132, 151, 152

! (exclamation mark) logical operator, 132

> (greater than sign) comparison operator, 132, 151

>= (greater than sign, equals sign) comparison operator, 132, 151

> (greater than signs) bit shifting operator, 132

>= (greater than signs, equals sign) assignment operator, 132

< (less than sign) comparison operator, 132

<= (less than sign, equals sign) comparison operator, 132, 151

<% (less than sign, percent sign) ASP.NET code block prefix, 876

<< (less than signs) bit shifting operator, 132

<<= (less than signs, equals sign) assignment operator, 132

- (minus sign) arithmetic operator, 132

-= (minus sign, equals sign) assignment operator, 132

-> (minus sign, greater than sign) indirection operator, 132

-- (minus signs) increment operator

-- (minus signs) increment operator, 132
(number sign) preprocessor directive prefix, 70
% (percent sign) arithmetic operator, 132
%= (percent sign, equals sign) assignment operator, 132
%> (percent sign, greater than sign) ASP.NET code block suffix, 876
. (period)
dot notation, 40
member access operator, 132
method calling operator, 1143
namespace separator, 58–59
regular expression special character, 232
+ (plus sign)
arithmetic operator, 132
regular expression special character, 232, 234–235
string concatenation operator, 132
+= (plus sign, equals sign) assignment operator, 132
++ (plus signs) increment operator, 132
?: (question mark, colon) ternary operator, 134
? (question mark) regular expression special character, 232
“ (quotation marks, double) string literal delimiters, 46
; (semicolon)
database connection string separator, 689
statement suffix, 31–32
/ (slash) arithmetic operator, 132
/* (slash, asterisk) comment prefix, 31
/= (slash, equals sign) assignment operator, 132
// (slashes) comment prefix, 31
/// (slashes) comment prefix, 68
~ (tilde)
destructor prefix, 194
logical operator, 132
_ (underscore) field name prefix, 76

A

Abort method, 443
AcceptChanges method, 823
AcceptSocket method
QuoteServer class, 1098
TcpListener class, 1083, 1107
AcceptTcpClient method (TcpListener class), 1083
access modifier, 97, 1140–1141, 1158, 1241
AccessDataSourceControl control, 334–335
ACID (atomicity, consistency, isolation, durability), 1009
Action ⇨ Add (.NET Framework Configuration tool), 380
action delegate, 174
Action ⇨ Properties (Component Services Admin utility), 1007

ActionCancelEventArgs class, 183–184, 186
ActionEventHandler delegate, 182
Activate method (ServicedComponent class), 1003
Activator class, 472, 473–474, 476, 483

Active Directory

access control list, 839
ADSI, 838, 847–848, 850, 857, 859–860
attribute
GC, storing in, 842
referencing, 855
schema, 839, 843
value, returning, 854–855
authentication, 841, 849, 853
binding, 849–854
cache, 857
currency, 843
DC, 838, 841
domain, 840–841
editing, 845–848
GC, 841–842, 850
hierarchy, 838, 840, 843
Kerberos support, 539
LDAP support, 838, 850
object
collection, 855–856
creating, 857–859
information about, returning, 854
storage, 839, 855–856
user object, 864–868
port configuration, 850
replication, 838, 841, 842–843
schema, 839, 841, 843–845, 856, 864–866
searching, 849, 860–868
security, 839, 842
site, 841
standards supported, 838
updating, 858–859
user interface, 864

Active Directory Domains and Trusts MMC snap-in, 845

Active Directory Service Interfaces. See ADSI

Active Directory Sites and Services MMC snap-in, 845

Active Directory Users and Computers MMC snap-in, 845–846

Active Server Pages. See ASP

Active Template Library. See ATL

ActiveX control, 24, 982–985, 1193

ActiveX Control Importer utility. See aximp utility

ActiveX Data Objects. See ADO

Add ⇨ Add Existing Item (Visual Studio .NET), 524

- Add method**
- ControlBindingsCollection class, 753
 - Controls class, 950
 - DirectoryEntry class, 857
 - Hashtable class, 247
 - HelloCSharp class, 363
 - HelloMCPPE class, 357, 359
 - HelloVB class, 361
 - IMath interface, 976
 - ResultPropertyValueCollection class, 858–859
 - Rows class, 724, 768
 - SchemaNameCollection class, 856
- Add ⇄ New Element (Visual Studio .NET), 767**
- Add ⇄ Project Output (Visual Studio .NET), 558**
- AddCulturesToTree method, 516–517**
- AddData method, 923**
- AddEvent method, 919, 920**
- AddMessageFilter method (Application class), 575**
- AddOrderLine method (Order class), 1012, 1020**
- add.ovf statement, 362**
- AddProductRow method, 718–719, 768–770**
- AddRange method (PropertiesToLoad class), 867**
- AddRef method (IUnknown interface), 967**
- Add/Remove Programs dialog box, application display in, 1055**
- AddResource method (ResourceWriter class), 523**
- AddressFamily class, 1082**
- AddToMessage method, 275**
- AddToOutput method, 270**
- ADO (ActiveX Data Objects), 686**
- ADO.NET. See also database**
- class overview, 686–688
 - command, executing, 694–698
 - data binding, 892
 - disconnected nature of, 688
 - namespace, 686
 - tier division of application, 728–729
 - XML
 - converting ADO.NET data to XML, 812–820
 - converting XML to ADO.NET data, 820–822
- AdRotator control, 884**
- ADSI (Active Directory Service Interfaces), 838, 847–848, 850, 857, 859–860**
- ADSI Edit utility, 845, 846–847**
- Advise method, 972**
- ampersand (&)**
- bit-wise operator, 132, 137
 - logical operator, 132
 - unary operator, 200
 - Windows Form label prefix, 589
- ampersand, equals sign (&=) assignment operator, 132**
- ampersands (&&) logical operator, 132**
- AnalyzeType method (MainClass class), 270**
- A.netmodule module, 349**
- AppDomain class, 342, 343**
- AppDomainUnloadedException exception, 333**
- Append method (StringBuilder class), 221, 222**
- AppendFormat method (StringBuilder class), 222, 225, 226**
- AppendLiteralString method, 955**
- Application class**
- AddMessageFilter method, 575
 - ASPNET, Application object, 906–907
 - CommonAppDataPath property, 575
 - DoEvents method, 575
 - EnableVisualStyles method, 575, 605, 606
 - ExecutablePath property, 575
 - Exit method, 575, 600
 - ExitThread method, 575
 - FlatStyle property, 575
 - LocalUserAppDataPath property, 575
 - MessageLoop property, 575
 - Run method, 574, 1198
 - StartupPath property, 575
- Application_BeginRequest method, 539**
- ApplicationContext class, 575**
- ApplicationException class, 278, 296**
- Application_Start method, 919**
- AppSupport project, 547**
- AppSupport.dll file, 556, 559**
- AppSupportII.dll file, 561**
- argSample.xml file, 809**
- arithmetic**
- addition, 137, 145–146, 147–148, 151
 - division, 137
 - modulus, 132
 - multiplication, 137, 144, 149–150
 - pointer arithmetic, 203–204
 - subtraction, 137, 151
 - Visual Basic 6 arithmetic assignment operator, 1201–1202
- array**
- accessing element within, 58
 - bound checking, 281, 1295–1296
 - C++, 1293–1296
 - data type, 57
 - DataGrid control, displaying in, 738–739

array (continued)

array (continued)

- declaring without initializing, 58
 - delegate array, 173, 174–175
 - dictionary compared, 246
 - index, 58, 240
 - initializing, 57
 - jagged, 212, 1294–1295
 - Java, 1226, 1236
 - list, 240–241
 - memory stack-based, 212–215, 1300
 - multidimensional, 52, 57, 212, 1294–1295
 - one-dimensional, 57, 1293–1294
 - pointer, 202, 214
 - rectangular, 212
 - referencing, 57–58
 - reflection, returning array from, 268, 270, 274
 - size
 - returning, 58
 - specifying, 57, 240, 1296
 - sorting, 168, 174–177, 521
 - threading, array processing in, 452–453
 - Visual Basic 6, 1218–1219
 - XML serialization, 826, 832, 834
- Array**
- class, 214, 241
 - namespace, 239
- ArrayController object, 452–453**
- ArrayList class, 240**
- arrayOfInts array, 54**
- ASCII class, 1087**
- ASP (Active Server Pages), 21, 22, 873**
- ASP.NET**
- application, 906–907
 - ASP relation to, 22, 873
 - AutoEventWireup attribute, 878
 - code-behind directive, 22
 - COM object, using from, 985
 - database connection, 895
 - event handling, 878, 879, 882, 899–901, 906–907
 - function definition, 876
 - globalization, 539–540
 - language attribute, 875, 878
 - localization, 539–540
 - object model, 878
 - page, 22
 - postback, 875, 890
 - processing, server-side, 875, 878, 881
 - remote hosting in ASP.NET environment, 494–495
 - runat attribute, 875, 878, 881
 - scripting language, declaring, 875
 - security, role-based, 415
 - state management, 875, 882
 - template, 903–906
 - variable declaration, 876
 - viewstate field, 875
 - Visual Studio .NET, developing in, 22, 875–877
- Web Form**
- AdRotator control, 884
 - Button control, 881–882, 885
 - Calendar control, 885, 897–899
 - CheckBox control, 884
 - CheckBoxList control, 885
 - CompareValidator control, 887
 - CustomValidator control, 887
 - data binding, 892, 894–897, 901–906
 - DataGrid control, 886, 901–902
 - DataList control, 886, 904–906
 - DropDownList control, 884
 - event handling, 878, 879, 882
 - HTML server control, 880
 - HyperLink control, 885
 - Image control, 884
 - ImageButton control, 885, 937–938
 - introduced, 23
 - Label control, 881, 884, 891
 - layout, 877
 - LinkButton control, 885
 - ListBox control, 884
 - Literal control, 884
 - Panel control, 885
 - Placeholder control, 884
 - posting, 882–883, 890
 - processing, server-side, 875, 878, 881
 - RadioButton control, 885
 - RadioButtonList control, 885
 - RangeValidator control, 887
 - RegularExpressionValidator control, 887
 - Repeater control, 886, 903
 - RequiredFieldValidator control, 887
 - server control overview, 880–883
 - sizing, 92
 - state management, 882
 - Table control, 885
 - TableCell control, 885
 - TableRow control, 885
 - TextBox control, 884
 - user control, 880
 - validation control, 880, 886–892
 - ValidationSummary control, 887, 888
 - Visual Studio .NET, developing in, 334, 875–877

- Web server control, 883–886
- Xml control, 884
- Web Matrix Project, 335
- .aspx files, **874, 877–878**
- assembly. See also GAC (global assembly cache); specific assembly**
- aliasing, 352
- application domain environment, in, 341–344
- Authenticode signature, 369
- binding flag, 344
- certificate, signing with, 430–435
- CLS compliance, 364–366
- COM, 352, 979–980, 993
- configuration file
 - application configuration, 377, 379–382
 - assemblyBinding element, 382
 - assemblyIdentity element, 382
 - bindingRedirect element, 382
 - codeBase element, 387–388
 - dependentAssembly element, 382, 388
 - machine configuration, 377
 - probing element, 389–390
 - publisher policy, 377, 383–385
 - remoting configuration, 377
 - runtime configuration, 377, 382, 385–386
 - security configuration, 377
 - shared assembly, 377
 - startup configuration, 376
 - supportedVersion element, 386
 - versioning configuration, 377–379
- control, placing in, 939–940
- CopyLocal property, 373
- creating, 350–352, 371–372
- culture, 346, 352, 527
- deleting, 367
- dependency, listing, 380–381
- deploying, 18, 341, 376, 556
- described, 339–340
- directory structure, 387–390
- encryption, 370, 372, 374–375, 428
- exception thrown by missing, 343
- executing, 343
- executing assembly, returning, 525
- GAC, installing to/uninstalling from, 368, 556
- garbage collection, 377
- hash code, 349–350
- listing, 367, 368
- loading/unloading, 271, 342
- Main method, 342–343
- manifest, 17, 346
- metadata, 17–18, 271, 344
- module, comparing to, 349
- name
 - collision, 19
 - returning, 379
 - strong, 347, 369–371, 372, 428
- namespace, 346–347
- overwriting, 19
- primary interop assembly, 979–980
- private, 18, 310, 340–341, 347
- probing, 377, 389–390
- publisher policy, 377, 383–385
- referencing, 341, 343, 346, 373, 375–376
- reflection, 267, 271, 274
- remoting, 377, 457, 463–464
- resource file, adding, 524–525
- runtime
 - application configuration file runtime element, 382
 - configuration file setup, 377, 385–386
 - garbage collection, 377
 - version, specifying, 376, 385–386
- satellite assembly, 533, 539, 540
- security
 - configuration file security setup, 377
 - permission, 346, 388, 397–398, 401–403, 408–409
- shared
 - configuration file, 377
 - creating, 371–372
 - GAC role, 19, 341
 - installing, 373
 - key, public, 369, 370
 - name collision, 19
 - name, strong, 347, 369–371, 372
 - overwriting, 19
 - private compared, 340
 - referencing, 373
 - versioning, 352, 369, 377–379
- side-by-side, 341
- version
 - configuration file setup, 377–379
 - private assembly, 340–341
 - redirecting, 382, 388
 - runtime version, specifying, 376, 385–386
 - shared assembly, 352, 369, 377–379
 - viewing, 347–348
 - zero impact installation, 18
- Assembly Cache Viewer, 367–368**
- Assembly class
 - FullName property, 379
 - GetExecutingAssembly method, 525

Assembly class (continued)

Assembly class (continued)

- GetManifestResourceNames method, 525
 - GetTypes method, 271, 274
 - Load method, 271
 - LoadFrom method, 271, 561
 - Reflection namespace declaration, 537
- AssemblyA.exe **file**, **342, 343**
- AssemblyInfo.cs **file**
- ApplicationAccessControl attribute, 1004, 1006
 - ApplicationActivation attribute, 1004, 1006
 - ApplicationName attribute, 1004, 1006
 - ASPNET application, for, 876
 - AssemblyCompany attribute, 352
 - AssemblyConfiguration attribute, 352
 - AssemblyCopyright attribute, 352
 - AssemblyCulture attribute, 352
 - AssemblyDefaultAlias attribute, 352
 - AssemblyDelaySign attribute, 352, 375
 - AssemblyDescription attribute, 352
 - AssemblyInformationalVersion attribute, 352
 - AssemblyKeyFile attribute, 372, 375, 993, 1004, 1012
 - AssemblyKeyName attribute, 352
 - AssemblyProduct attribute, 352
 - AssemblyTitle attribute, 352
 - AssemblyTrademark attribute, 352
 - AssemblyVersion attribute, 378
 - CLSCompliant attribute, 365
 - Description attribute, 1004
 - EventTrackingEnabled attribute, 1004
 - introduced, 311
 - permission configuration, 408
 - Visual Studio .NET, generation by, 350–351
- asterisk (*)**
- arithmetic operator, 132
 - regular expression special character, 232
 - Visual Studio .NET autoversioning operator, 378
- asterisk, equals sign (=) assignment operator, 132**
- asterisk, slash (/) comment suffix, 31**
- AsyncProcessMessage **method (IMessageSink interface), 476**
- AsyncProcessMessage **method (SoapClientFormatterSink class), 471**
- at sign (@) string literal prefix, 46**
- ATL (Active Template Library), 973**
- atomicity, consistency, isolation, durability. See ACID**
- attendeeList **control, 894–897**
- attribute**
- assembly attribute, returning, 272
 - C++, 1304
 - compilation error caused by incorrect, 259, 261
 - custom, 258–265, 272
 - instantiating, 261
 - introduced, 17
 - parameter, specifying, 261–262
 - reflection, returning attribute information using, 272, 275
 - struct, using with, 1235–1236
- Attribute **class, 259, 272, 275**
- attributeSchema **object, 839**
- AttributeTargets **enumeration, 260**
- Audit **method, 412**
- AuditClass **method, 412, 413**
- AuthenticateUser **method, 925**
- AuthenticationToken **class, 925**
- Authenticator **class**
- ChangePassword method, 1141, 1143, 1146–1147
 - constructor, 1173–1174
 - GetMinPasswordLength method, 1150
 - IsPasswordCorrect method, 1141, 1146
 - minPasswordLength field, 1150–1151
- Authenticode signature, 369**
- AutoScrollPosition **class, 643**
- aximp (ActiveX Control Importer) utility, 982**

B

- backslash (\) escape sequence prefix, 44, 46, 232**
- backslashes (\\) escape sequence prefix, 46**
- BankAccounts.cs **file, 128**
- bar (|)**
 - bit-wise operator, 132
 - logical operator, 132
- bar, equals sign (|=) assignment operator, 132**
- bars (||) logical operator, 132**
- BaseValidator **class, 886**
- BasicConsoleApp.sin **file, 312**
- Basics **namespace, 32**
- BeginEdit **method (DataRow class), 709**
- BeginGetResponse **method (WebRequest class), 1073**
- BeginInit **method**
 - DataGrid class, 737
 - ISupportInitialize interface, 616–617
- BeginInvoke **method, 498**
- Beginning Visual C#, 30**
- Beginning XML (Wrox Press), 781**
- BeginTransaction **method, 692**
- binary file, reading/writing, 1041, 1042–1047**

Binary namespace, **826**
 BinaryFormatter class, **470**
 BinaryReader class, **1041, 1042**
 BinaryServerFormatterSinkProvider class, **480**
 BinaryWriter class, **1041–1042**
 Binding class, **755**
 BindingFlags class, **344**
 BindingManagerBase class, **582, 752, 778**
 BitBit (bitmap block transfer), **655**
 block statement, **690–692**
 BookOfTheDayForm class
 constructor, 534
 InitializeComponent method, 528, 530
 Localizable property, 530–531
 SetDateAndNumber method, 529
 Show method, 528
 TabIndex property, 531
 Text property, 531
 WelcomeMessage method, 528, 529, 537
 BookOfTheDayForm.de.resX file, **532**
 BookOfTheDayForm.fr.resX file, **532**
 BookProduct class, **831, 833**
 booksscript.xsl file, **811**
 books.xml file, **783–784**
 BookUtils class, **810**
 boxing/unboxing, **141–142, 160–161, 1236–1237, 1273**
 brackets, curly { } statement block constructor, **31**
 brackets, round ()
 cast operator, 132
 method suffix, 87
 brackets, square []
 array element index delimiters, 58, 132
 reflection attribute delimiters, 260
 regular expression alternative character delimiters, 233
 break statement, **55, 328**
 breakpoint, debugging using, **55, 328, 331–332**
 Brushes class, **648**
 BubbleSorter class, **174–177**
 BufferedReader object, **1249**
 BufferedStream class, **1042**
 BusEntity class, **184–185**
 Button control, **578, 585, 737, 881–882, 885**
 ButtonBase class, **585**
 Button_Click method, **180–181**
 buttonLoadProperties_Click method, **865**
 byte code marker, **1048**

C

C++

array, 1293–1296
 attribute, 1304
 class, 355–359, 1256, 1257, 1279–1287
 compilation, 1255, 1304
 constant, 1287–1289
 constructor, 1257, 1281–1283
 data type, 1266–1270, 1271–1273
 delegate, 1302–1303
 destructor, 1256, 1283–1284
 enumeration, 1256, 1296–1297
 escape sequence, 1270–1271
 event handling, 1303–1304
 exception handling, 1256, 1297–1298
 field, 1280–1281
 file structure, 1261–1262
 flow control, 1256, 1263–1266
 forward declaration, 1262–1263
 function, 1254, 1255, 1285–1287
 include statement, 1259
 inheritance, 1257, 1258, 1285
 interface, 1300–1302
 library, 1256, 1259
 main method, 1260, 1262
 memory management, 1255, 1273–1275, 1298–1300
 message box display, 1260–1261
 method, 1260, 1262, 1275–1276, 1302–1303
 namespace, 1259–1260
 object model, 1253, 1261
 operator, 1256, 1277–1279, 1289–1291
 preprocessor directive, 1256, 1304–1305
 property, 1276–1277
 statement, compound, 1255
 string handling, 1270
 struct, 1256, 1279, 1287
 syntax, 1256, 1262–1263
 target environment, 1256
 template, 1258
 terminology, 1254–1255
 try catch finally block, 1297–1298
 variable, 1254, 1266, 1272
 Visual Studio .NET, creating C++ code using, 356
 c XML tag, **68**
 cab files, **551, 557**
 CalculateDocumentSize method, **667–668**
 CalculateLineWidths method, **667**
 Calendar control, **518, 885, 897–899**

calendar, localizing, 510, 518

Callback **method, 498**

CallContext **class, 506, 507**

CallContextData **class, 506**

camel casing, **76**

CanBePooled **method (ServicedComponent class), 1003**

CancelEventArgs **class, 183**

Candidate **class, 955, 959, 960–961**

CandidateEventArgs **class, 960**

CanWeFixIt **method, 918**

CaptureCollection **class, 236**

caret (^)

logical operator, 132

regular expression special character, 232

caret, equals sign (^=) assignment operator, 132

case sensitivity

C#, 66, 73, 77, 1184

CLS, 12

Java, 1226

Visual Basic .NET, 12, 77

Visual Basic 6, 1184

caspol (Code Access Security Policy) tool, 391, 394–396

casting

C++, 1269–1270

checked operator, 140

classes, between, 155, 158–160

combining casts, 161–165

error handling, 140

explicit, 139–141

Java, 1236–1237

memory pointer, 202–203

overloading, 153

structs, between, 155

syntax, 139, 154

user-defined, 154–158

catch **block. See try catch finally block**

CComBSTR **class, 976**

CComDemo **class, 974, 976, 979**

CComDemoClass **class, 978**

CCW (COM Callable Wrapper), 986, 993

Ceiling **method (PointF struct), 633**

certificate, 429–435, 557. See also encryption

ChangeColorDepth **method, 442**

ChangePassword **method (Authenticator class), 1141, 1143, 1146–1147**

channel. See remoting, channel

ChannelDataStore **class, 468**

ChannelServices **class, 458, 470, 471**

CheckBox **control, 585, 884**

CheckBoxList **control, 885**

checked/unchecked operator, 134–135

CheckState **enumerator, 585**

chktrust.exe **utility, 432**

class

abstract, 115, 1165–1169, 1242, 1246

access modifier, 97, 1140–1141, 1158, 1241

ADO.NET, 686–688

base class, 19–20, 32, 286, 1166–1169, 1179

C++, 355–359, 1256, 1257, 1279–1287

channel class, 456, 467

CLS compliance, 354

CLSID, 969, 974, 989, 990

constant association with, 85, 1288–1289

constructor

C#, 86, 95–99, 116–121, 1172–1174

C++, 1257, 1281–1283

initializer, 98–99, 1282–1283

static, 84, 96–98, 1282

Visual Basic 6, 1199

database schema, mapping to, 716

defining

C#, 32, 182, 184–185, 362–364, 1140–1141

C++, 355–359, 1279–1280

Visual Basic .NET, in, 359–362

delegate, relation to, 168, 169, 178

destructor

C#, 193–194, 195–197, 1171–1172

C++, 1256, 1283–1284

Java, 1245

entity class, 1012–1015

event

creating handler class, 182, 184–185

member, as, 85

exception

base class library, thrown by, 286

overview of exception classes, 278–280

user-defined exception class, 290–297

hierarchy, 1159–1161

indexer, 86, 153, 1291

instantiating, 84, 170, 1196–1198

interoperability, 9

Java, 1240, 1241–1243, 1245–1249

marshal-by-reference class, 477

marshal-by-value class, 476

member, 85, 1139, 1161–1164, 1241–1245

module, 1138

name, fully qualified, 32

namespace

- defining within class, 58
- returning class namespace, 60
- .NET environment, 19–20
- object, relation to, 1139
- pointer, 202, 206–207, 209–212
- remoting channel
 - class representation, 456, 467
 - serialization, 470, 476, 480
- sealed, 115–116, 1169–1170
- SQL class overview, 687
- struct versus, 84, 101–102
- TCP class overview, 1083
- type, returning, 105
- unbound, 476
- variable, class-level, 37–38
- Visual Basic 6, 1178, 1187, 1192–1193, 1196–1198
- Web service proxy class, 917
- XML class overview, 782–783, 795–797
- XPath class overview, 803
- class id. See CLSID**
- Class1.cs **file, 308**
- classSchema **object, 839, 843**
- Clear **method** (ResultPropertyValueCollection **class**), **858**
- ClearAllFields **method, 1039**
- ClearTextFields **method, 518**
- Client **class, 500**
- Client.exe.config **file, 381, 382**
- ClientSponsor **class, 482**
- ClientValidationFunction **function, 887**
- Close **method**
 - Dispose method versus, 196
 - IResourceReader interface, 541
 - PCSSWebApp3 application, 895
 - RegistryKey class, 1059
 - XmlTextWriter class, 801
- CLR (Common Language Runtime)**
 - described, 4
 - interoperability, 5–7, 8–9
 - memory type safety, 7, 9–10
 - Object Remoting, 457
 - permission evaluation, 399, 407
 - platform independence, 4–5
 - VES, 393
- CLS (Common Language Specification), 12, 353–354, 364–366**
- CLSID (class id), 969, 974, 989, 990**
- Club **method, 937**
- CoCreateInstance **method, 969, 994**
- code**
 - block, defining for compilation, 72
 - distributing
 - certificate, using, 429–435, 557
 - strong name, using, 427–429
 - going to another line, 49, 54–55
 - group, 392, 393–398, 423–425
 - mobile, 391
 - security
 - code distribution, of, 427–435, 557
 - code-based, 14
 - evidence, 392
 - mobile code, 391
 - permission, 392, 399–403, 413, 414, 424–425
 - trust-based, 392
 - unmanaged code, 419
 - zone, 393
- Code Access Security Policy tool. See caspol tool**
- code **XML tag, 68**
- code-behind **directive, 22**
- CoInitialize **method, 994**
- ColdCallFileFormatException **exception, 290, 294**
- ColdCallFileReader **class, 292, 293, 294, 295**
- Collect **method** (GC **class**), **193**
- collection**
 - context, 460–462
 - described, 242–243
 - enumerator, 242
 - ICollection interface, 240, 242
 - IEnumerable interface requirement, 54, 242
 - iterating through, 54, 243, 1239–1240
 - Java, 1236
 - Vector struct, 243–245
- Collections **namespace, 105, 240**
- colon (:)** WriteLine **method format specifier prefix, 223**
- colons (::) event cancellation string, 184**
- Color
 - class, 605, 646
 - struct, 645
- ColorDialog **class, 1060**
- Colors **enumeration, 106**
- Columns **collection, 705**
- COM**
 - ActiveX control, 982–985
 - ASPNET, using COM object from, 985
 - assembly, 352, 979–980, 993
 - ATL project, creating for COM component, 973

COM (continued)

- attribute
 - custom, 975
 - interop, 989–992
- casting, 968
- CCW, 986, 993
- CLSID, 969, 974, 989, 990
- data type, 969, 972–973
- Enterprise Services environment, COM+ application in, 1003–1006
- error handling, 971
- event handling, 972, 980–982, 986, 996
- garbage collection, 13, 966–967
- GUID, 369, 976, 990
- interface, 124, 967–968, 989, 990, 991
- interoperability overview, 7, 965–966
- language of client, programming for, 353
- library application, 1004
- marshaling, 972–973
- metadata, 966
- method binding, 969, 989
- .NET client, using COM component from
 - ActiveX control, 982–985
 - component configuration, 973–977
 - connection point, 980–982
 - RCW, 977–980
 - registration, 974
 - threading, 980
- .NET component, using from COM client
 - CCW, 986, 993
 - client configuration, 993–995, 996–997
 - connection point, 995–996
 - interop attribute, 989–992
 - registration, 992–993
 - type library, 987–989
 - Windows Form control, 997–998
- object model, 8–9
- registry, component registration in, 969, 974, 992–993
- server, 973
- threading, 933, 969–971, 980

COM Callable Wrapper. See CCW

Combine **method (Path class), 1029**

ComboBox **control, 586**

COMClient.cpp **file, 993**

COMDemo.cpp **file, 976, 981**

COMDemo.h **file, 974, 976, 980**

comma (,)

- inheritance list separator, 111
- WriteLine method character number prefix, 223

Command **class**

- CommandType property, 693
- ExecuteNonQuery method, 694–695, 699
- ExecuteReader method, 694, 695–696, 701, 702
- ExecuteScalar method, 694, 696
- ExecuteXmlReader method, 696–698
- SQL clause, passing to, 693

CommandType **enumeration, 693–694**

comment

- C#, 31, 68–70
- Visual Basic 6, 1182–1183
- XML documentation, generating from, 68–70

Commit **method (Installer class), 1108**

CommitChanges **method (DirectoryEntry class), 857, 859, 860**

Common Language Runtime. See CLR

Common Language Specification. See CLS

Common **namespace, 686, 687**

Common Properties of Project Properties (Visual Studio .NET), 528

Common Type System. See CTS

Compare **method (String class), 218**

CompareOrdinal **method (String class), 218**

CompareValidator **control, 887**

comparison operation, 132, 137, 142–144, 151–153, 174–175

compilation. See also attribute

- Attribute class considerations, 259
- batch file, from, 31
- C++, 1255, 1304
- code block, defining for, 72
- conditional, 71, 329
- debugging information, inserting during, 328–329, 330
- error message, specifying, 72
- IL, 4
- Java, 1226
- JIT, 5, 94
- line numbering, 72–73
- Main method compilation considerations, 61–62
- .NET Framework targeted, 3
- out switch, 64
- output file, specifying, 64
- preprocessor directive, 70–73
- reference switch, 64
- target switch, 63
- Visual Basic .NET, 6
- Visual Basic 6, 1179
- Visual C++ .NET, 6–7
- Visual Studio .NET, 31, 302, 307–308, 326–331
- warning text, specifying, 72

CompilerServices **namespace, 351**

- Completed **method**, **980, 996**
- Component **class**, **579**
- Component Services Admin utility**, **1006–1008, 1009**
- Component.dll **file**, **344–345**
- COMWrapper **namespace**, **978**
- .config **files**, **484**
- Configure **method** (RemotingConfiguration **class**), **489, 490**
- Connect **method** (RemotingServices **class**), **473, 476**
- Connection **class**, **541, 1018**
- Console **class**
- OpenStandardError method, 1250
 - OpenStandardInput method, 1250
 - OpenStandardOutput method, 1250
 - Read method, 1250
 - ReadLine method, 33, 65, 283, 1250
 - SetError method, 587, 1250
 - SetIn method, 1250
 - SetOut method, 1250
 - Write method, 52, 65, 1250
 - WriteLine method, 65–66, 165, 223–224, 1148–1149, 1250–1251
- const **statement**, **1275**
- constant**
- advantage of using, 38–39
 - C++, 1287–1289
 - class, association with, 85, 1288–1289
 - declaring, 38, 85
 - inheritance, 1289
 - initializing, 38
 - variable, designating as, 38
- Constraint **class**, **687**
- Construct **method** (ServicedComponent **class**), **1003, 1016–1017**
- ContainerControl **class**, **583**
- Contains **method** (DirectoryEntry **class**), **858**
- context**, **460–462, 505–507, 1001**
- ContextAttribute **interface**, **461**
- ContextBoundObject **class**, **477**
- ContextDataRow **class**, **770, 772, 775**
- ContextMenu **class**, **595, 596**
- ContextUtil **class**, **1011**
- Continue
- ServiceController class method, 1121
 - statement, 55
- control**. *See also event, control event; specific control*
- color, 97, 883, 946, 947, 950–951
 - custom control
 - assembly, placing in, 939–940
 - bin directory, placing in, 939
 - color, 946, 947
 - composite, 949–951
 - data binding, 945
 - debugging, 941
 - event handling, 947–948, 959–962
 - Label control, deriving from, 945–946
 - library, 940–941
 - nesting, 940
 - state management, 946–949, 956
 - tag prefix, 940
 - user control compared, 939
 - Visual Studio .NET project, 940–943
 - data binding
 - ADO.NET, 892
 - ASP.NET Web Form control, 892, 894–897, 901–906
 - attendeeList control, 894–897
 - Calendar control, 897–899
 - context, 752–754, 777
 - currency, 755–757
 - custom control, 945
 - DataGrid control, 737–739, 750–751, 777–778, 901–902
 - described, 750
 - roomList control, 894–897
 - TextBox control, 750–751
 - Windows Form control, 582, 750–757
 - GAC, placing in, 931
 - introduced, 951–952
 - nesting, 940
 - user control
 - ASP.NET Web Form, 880
 - composite, 934
 - control custom compared, 939
 - event handling, 936, 938
 - files created with, 932–933
 - graphic, associating with, 933
 - postback, 936
 - referencing, 934
 - reusability, 932
 - state, specifying default, 934
 - tag name, 934
 - Visual Studio .NET user control project, 932
- Control **class**
- BackColor property, 580
 - BackgroundImage property, 580
 - BindingContext property, 582, 752–755
 - ForeColor property, 580
 - hierarchy, 582–584, 754
 - HWND property, 582
 - introduced, 579

Control class (continued)

Control **class (continued)**

- Invalidate method, 582
- IsHandleCreated properties, 582
- LParam property, 582
- MousePosition property, 673
- Msg property, 582
- PointToClient method, 580
- PointToScreen method, 580
- RectangleToScreen method, 580
- Result property, 582
- ScreenToRectangle method, 580
- WndProc method, 582
- WParam property, 582

ControlBindingsCollection **class**, 751, 753

ControlBuilder **class**, 955

Controls **class**, 950

Convert

- class, 283, 285
- System class method, 157–158

CopyTo **method**

- DirectoryInfo class, 1028
- File class, 1035
- FileInfo class, 1028
- StringCollection class, 1053

CoUninitialize **method**, 994

CountryItem **class**, 590

CountryList **class**, 590

Create **method**

- DirectoryInfo class, 1028
- FileInfo class, 1028
- Order class, 1012, 1020
- OrderLine class, 1014
- WebRequest class, 1071, 1077

CreateChildControls **method**, 950, 953

CreateCommand **method (Connection class)**, 541, 1018

CreateDataSet **method**, 748, 750

CreateDomain **method (AppDomain class)**, 343

CreateElement **method (XmlDocument class)**, 800

CreateEventSource **method (EventLog class)**, 1126

CreateFonts **method**, 665

CreateGraphics **method (Form1 class)**, 626–627

CreateInstance **method**

- Activator class, 472, 473–474, 483
- BindingFlags class, 344
- COMClient.cpp application, 994

CreateMessageSink **method (IChannelSender interface)**, 468, 470, 475

CreateSink **method**

- IClientChannelSinkProvider interface, 470
- IServerChannelSinkProvider class, 470

CreateSpecificCulture **method (CultureInfo class)**, 539–540

CreateSubKey **method (RegistryKey class)**, 1058, 1059, 1064

CreateText **method (FileInfo class)**, 1029

CrossContextChannel **class**, 462

CrossLanguage **namespace**, 346, 357, 359

cryptography, public key, 370, 374

.cs **files**, 30

csc **command**, 31

CSP (Crypto Service Provider), 372

CTS (Common Type System), 10–12, 40–41, 353, 354

culture

- assembly, 346, 352, 527
- calendar, 518
- date format, 512, 515–516
- default, specifying, 512–513
- introduced, 511
- invariant, 512
- language, translating automatically based on, 534–539
- listing all cultures, 516–520
- neutral, 512
- number format, 512, 513–515
- operating system language, dependent on, 512
- resource for, returning, 527
- resource reader based on, 540–544
- sorting order, 520–522
- specific, 511–512, 519
- text format, 510–511
- threading, 444, 512
- user interface, 512

CultureDemoForm **class**, 516–517

CultureInfo **class**

- Calendar property, 518
- CreateSpecificCulture method, 539–540
- GetCultures method, 517
- IFormatProvider interface implementation by, 514
- introduced, 511
- IsNeutralCulture property, 518, 519
- OptionalCalendars property, 518

Currency **struct**, 155–156, 161–163, 170–171

currency value, formatting, 66–67, 141–142, 160–161

CurrencyClass **class**, 209

CurrencyManager **class**, 753, 755–757

CurrencyStruct **struct**, 209–210

CurrentAccount **class**, 129

CurrentAccount.cs **file**, 130

CurrentPosition **method** (XPathNodeIterator **class**), **804**
 Customer **class**, **190–191**, **1153–1154**
 CustomerRow **class**, **774–775**
 CustomValidator **control**, **887**
 Cycle **method** (RainbowLabel **class**), **947**, **949**

D

data adapter, **722–723**, **724–726**, **760**

data binding

ADO.NET, 892
 ASP.NET Web Form control, 892, 894–897, 901–906
 attendeeList control, 894–897
 Calendar control, 897–899
 context, 752–754, 777
 currency, 755–757
 custom control, 945
 DataGrid control, 737–739, 750–751, 777–778, 901–902
 described, 750
 roomList control, 894–897
 TextBox control, 750–751
 Windows Form control, 582, 750–757

data encapsulation, **1151–1152**

Data **namespace**, **686–687**, **713**

data reader, **701–704**

data type

array, 57
 blittable/non-blittable, 972–973
 C++, 1266–1270, 1271–1273
 COM, 969, 972–973
 converting between primitive types, 137–142
 converting, explicit, 139–141, 154
 converting, implicit, 138–139, 154
 converting in Java, 1236
 converting using as operator, 136
 converting using boxing/unboxing, 141–142, 160–161, 1236–1237, 1273
 converting using casting
 C++, 1269–1270
 checked operator, 140
 classes, between, 155, 158–160
 combining casts, 161–165
 error handling, 140
 explicit, 139–141
 Java, 1236–1237
 memory pointer, 202–203
 overloading, 153
 structs, between, 155

syntax, 139, 154

user-defined, 154–158

converting using Convert class method, 283

converting using platform invoke, 20

converting using ternary operator, 134

CTS, 10–12, 40–41

database column data type support, 706

DataSet class type safety, 757

IL, 9–10

immutable, 219

interoperability, 10

Java, 1230–1236

memory storage, 39, 188–192

object compatibility, checking for, 135–136

predefined, 39–46

reference data type, 39–40, 190–192, 1147–1148, 1222, 1236

reflection, 266–271, 274

safety, 137, 170

size in bytes, returning, 136, 204–205, 210

string, 45–46, 219

strong data typing, 9

value data type, 39–40, 188–189, 1222, 1230–1236

Visual Basic 6, 1221–1222

Web service, available for, 916

database. See also ADO.NET; SQL (Structured Query Language)

column

 copying, 728–729

 data type support, 706

 DataGrid control column style, 747–750

 identity column, 730, 731–732, 762

 inserting, 705–707

connection

 ASP.NET application, 895

 closing, 689–692, 702

 Enterprise Services application, 1003, 1016–1019

 instance, specifying, 689

 OLE DB connection, 702

 opening, 689, 702

 SQL server, 686, 689, 758

 Visual Studio .NET, creating in, 758–759

 Web service application, 919, 920–921, 923

constraint, 713–715, 733

data source

 DataGrid control, 737, 738–746, 750–757, 777

 OLE DB source, 701, 702, 703

 updating, 763–764

mapping, 687

naming convention, 732–733

database (continued)

- permission, 399, 400
- procedure, stored
 - calling, 698–701
 - data adapter, using in, 722–723
 - naming, 733
- query result set, iterating through, 695
- reader, 701–704
- record
 - count, returning, 696
 - deleting, 699
 - inserting, 700–701
 - iterating through records, 701
 - selecting, 701
 - updating, 698–699, 724–726
- relationship
 - constraint, 713–715, 733
 - creating, 711–712
 - key, foreign, 714
 - key, generating using SQL Server, 730–732
 - key, primary, 700, 711, 713, 725
- resource reader application, 541
- row
 - child row, returning, 709
 - class containing row definition, 687
 - copying, 728–729
 - DataGrid control row color, 748
 - deleting, 726
 - editing data in, 709
 - inserting, 718–720, 724–726, 730–731, 774
 - parent row, returning, 712
 - pop-up menu, providing for, 770–778
 - state flag, 708–709, 724
 - type, returning, 774
 - updating, 726
 - value in, returning, 707–708
- schema
 - class, mapping to, 716
 - creating, 709–711, 764–770
 - definition file, 715–721
 - generating, 709–710, 716–721
 - introduced, 704–705
 - Visual Studio .NET environment, 762–763, 764–770
- scope, 732
- table
 - copying, 728–729
 - DataGrid control table style, 747–750
 - linking, 687, 704, 713, 732
 - XML schema, creating from, 821, 824
- tier division of application, 728–729
- transaction
 - aborting, 1011, 1019
 - ACID, 1009
 - automatic, 1001, 1009, 1017
 - beginning, 692
 - class overview, 687
 - committing, 692, 1011
 - consistent bit, 1010–1011
 - disabling, 1010
 - displaying active, 1020
 - distributed, 1001
 - done bit, 1010–1011
 - DTC, 1001
 - isolation level, 692–693
 - requiring, 1007, 1010, 1016
 - result, handling, 1010–1011
 - rollback, 692, 693, 1001
 - serializing, 693
 - success, determining, 1010–1011, 1019
- XML
 - converting ADO.NET data to XML, 812–820
 - converting XML to ADO.NET data, 820–822
 - DataSet, populating from, 723
 - outputting, 726–728
- DatabaseResourceManager **class**, 543, 544
- DatabaseResourceReader **class**, 541–542, 544
- DatabaseResourceSet **class**, 542–543
- DataBind **method**, 896, 902
- DataBindingsCollection **class**, 754
- Data-Centric .NET Programming with C# (Wrox Press)**, 836
- DataColumn **class**, 687, 705–707, 770
- DataColumnMapping **class**, 687
- DataGrid **control**
 - array, displaying in, 738–739
 - ASP.NET Web Form, 886, 901–902
 - BeginInit method, 737
 - button, 737
 - creating, 736
 - data binding, 737–739, 750–751, 777–778, 901–902
 - data source, 737, 738–746, 750–757, 777
 - DataMember property, 777–778
 - DataSet data, displaying in, 743–745, 748
 - DataTable data, displaying in, 739–740
 - DataGridView data, displaying in, 740–743
 - event handling, 737, 902
 - hierarchy, 746–750
 - HitTest method, 778
 - IList interface data, displaying in, 746
 - IListSource interface data, displaying in, 746

- introduced, 705
- row color, 748
- SetDataBinding method, 737, 738, 777
- style, 747–750
- TableName property, 747
- TableStyles property, 747, 748–749
- DataGridColumnStyle **class, 747–750**
- DataGridTableStyle **class, 747–750**
- DataList **control, 886, 904–906**
- DataRelation **class, 687**
- DataRow **class**
 - BeginEdit method, 709
 - EndEdit method, 709
 - generating, 769–770, 771
 - GetChildRows method, 709
 - GetParentRow methods, 712
 - indexer, 707–708, 726
 - introduced, 687
 - Item property, 825
 - SqlDataAdapter class, accessing from, 707
 - version value, 707–708
- DataRowState **enumeration, 708, 729**
- DataRowView **class, 778**
- DataSet **class**
 - copying DataSet object, 728–729
 - DataGrid control, displaying DataSet data in, 743–745, 748
 - Fill method, 723, 816
 - GetChanges method, 729
 - hierarchy, 704
 - introduced, 686
 - Merge method, 729
 - PCWebApp3 application, 896
 - persistence, 723–725
 - populating DataSet object, 721–723
 - property, extended, 705
 - ReadXML method, 723, 820
 - relationship, establishing using, 711–712
 - serializing, 767
 - type safety, 757
 - Web service application, working with in, 919–923
 - WriteXml method, 727, 813, 818
 - XmlDataDocument class, passing DataSet as parameter to, 816
- DataTable **class**
 - Columns collection, 705
 - database table compared, 704
 - DataGrid control, displaying DataTable data in, 739–740
 - DiffGram data, reading into, 824–825
 - generating, 771
 - introduced, 686
 - linking DataTable objects, 687, 713
 - NewRow method, 720, 724, 768
 - ParentRelations property, 712
 - procedure, calling from, 700
 - Rows collection, 705
 - RowState property, 708–709
 - schema, 704–705, 709–711, 715–721
- DataTableMapping **class, 687**
- datatype **statement, 34**
- DataGridView **class, 740–743**
- date**
 - calendar
 - control, 885, 897–899
 - localizing, 510, 518
 - culture, 512, 515–516
 - day of week, calculating, 98
 - file creation date, reading into field, 100
 - formatting, 223, 512, 515–516
 - DateTime **class, 98, 100, 515, 516**
 - DateTimeFormatInfo **class, 514, 515**
 - DateTimePicker **control, 587**
 - DateTimePickerFormat **enumeration, 587**
 - DC (device context), 625–626**
 - DC (domain controller), 838, 841**
 - DDX (Dialog Data Exchange), 750**
 - Deactivate **method (ServicedComponent class), 1003**
 - deadlock, 450, 451–452**
 - DEBUG **preprocessor symbol, 330**
 - Debug ⇄ Windows ⇄ Breakpoints (Visual Studio .NET), 332**
 - debugging**
 - breakpoint, using, 55, 328, 331–332
 - compilation, inserting debugging information during, 328–329, 330
 - control, custom, 941
 - Enterprise Services application, 1007, 1021
 - GDI+, 637–638
 - variable value, watching, 332
 - Visual Studio .NET debugging features, 302, 328–329, 330, 331–334
- define **preprocessor directive, 70–71, 1305**
- delegate**
 - array, 173, 174–175
 - C++, 1302–1303
 - class, relation to, 168, 169, 178
 - defining, 169
 - described, 167–168, 1257

delegate (continued)

- event handling, in, 168, 179, 180–181, 182
- instantiating, 170
- loop operation involving, 173, 174
- multicast, 177–179
- syntax, 169
- thread, starting, 168
- type safety, 170

Delete method

- data adapter, of, 726
- Directory class, 1035
- DirectoryInfo class, 1028
- File class, 1035
- FileInfo class, 1028

DELETE SQL statement, 694, 699, 724

DeleteCommand class, 724

DeleteSubKey method (RegistryKey class), 1058, 1059

DeleteSubKeyTree method (RegistryKey class), 1059

DeleteValue method (RegistryKey class), 1059

Demand method (FileIOPermission class), 406

Demo.resources file, 523–524

DemoSolution.sln file, 313

Deny method, 411–412

deployment. See also installer

- Add/Remove Programs dialog box, application display in, 1055
- assembly, 18, 341, 376, 556
- bootstrapper option, 557
- cab files, 551, 557
- certificate, 429–435, 557
- COM+ application in Enterprise Services environment, 1005–1006
- compression, 557
- copying file set to target computer, via, 547
- file type association, establishing, 564
- folder on client machine, creating/specifying, 556, 562
- launch condition, specifying, 568–569
- NTD, 341, 561–562
- planning, 545
- registry, accessing, 551, 562–563, 568–569
- shortcut to application, creating, 556–557, 562
- system requirement, 546–547
- Visual Studio .NET project
 - copying, deploying application via, 546, 550
 - deployment project, 546, 551–552, 553, 556–559
 - installer project, 551
- Web application, 548–549, 551, 559–560

- Web site, from, 561
- xcopy deployment, 18, 376, 546, 548–549

depthChangeThread class, 443

Deserialize method

- IRemotingFormatter interface, 470
- XmlSerializer class, 825, 827

Design namespace, 625

destructor

- C#, 193–194, 195–197, 1171–1172
- C++, 1256, 1283–1284
- Java, 1245

device context. See DC

device driver, returning, 1117

Diagnostics namespace, 1126, 1130

Dialog Data Exchange. See DDX

DialogResult enumeration, 600–601

Diamond method, 937

dictionary

- adding item, 247, 248
- array compared, 246
- functionality provided by, 245
- hashing algorithm, 247, 248–249, 250, 252–253
- Hashtable class, representation in, 247, 469
- IDictionary interface, 475
- IMessage interface, representation in, 475
- key, 246, 247–248, 249
- load, 249–250
- placing item, 249
- removing item, 248
- retrieving data from, 247

DictionaryBase class, 247

DiffGram document, 813, 821, 822–825

Dimensions class, 101

directory. See folder

Directory class, 1026, 1027, 1035

DirectoryEntries class, 849, 855–856

DirectoryEntry class

- Add method, 857
- AuthenticationType property, 853
- Children property, 849, 855
- CommitChanges method, 857, 859, 860
- Contains method, 858
- Guid property, 854
- introduced, 849
- Invoke method, 859
- Name property, 854
- NativeGuid property, 852
- NativeObject property, 860
- Parent property, 855

- Password property, 853
 - Path property, 853, 861
 - Properties property, 855, 857
 - RefreshCache method, 857
 - SchemaClassName property, 854
 - Username property, 853
 - DirectoryInfo **class**, **1026, 1027, 1028, 1034**
 - DirectorySearcher **class**, **849, 860–863**
 - DirectoryServices **namespace**, **837, 848–849, 965**
 - DisableCommit **method** (ContextUtil **class**), **1011**
 - DisplayData **method** (TestHarness **class**), **255**
 - DisplayFile **method**, **1046, 1052, 1053**
 - DisplayFileInfo **method**, **1032**
 - DisplayFolderList **method**, **1034–1035**
 - DisplayNames **method**, **521**
 - DisplayNumbers **method**, **444–445, 446**
 - DisplayTypeInfo **method** (WhatsNewChecker **class**), **274**
 - Dispose **method**
 - Close method versus, 196
 - ColdCallFileReader class, 292, 293, 295
 - Form1 class, 654
 - IDisposable interface, 110, 124–125, 195–197, 542, 577
 - Image class, 654
 - Service1 class, 914
 - Distributed Transaction Coordinator. See DTC**
 - DN (distinguished name)**, **850–851**
 - Dns **class**, **1080–1082**
 - DNS (Domain Name System)**, **399, 1080–1082**
 - do while **statement**, **53, 1264**
 - DoActions **method** (BusEntity **class**), **185**
 - Document Type Definition. See DTD**
 - documentLines **field**, **663**
 - DoEvents **method** (Application **class**), **575**
 - dollar sign (\$) **regular expression special character**, **232**
 - DOM (Document Object Model)**, **795–797. See also XML (Extensible Markup Language)**
 - domain controller. See DC**
 - Domain Name System. See DNS**
 - DomainTest.exe **file**, **343**
 - DotnetComponent **class**, **986, 987, 989, 995–996**
 - DoubleOp **delegates**, **173**
 - Download Assembly Cache**, **391, 401**
 - DownloadData **method** (WebClient **class**), **1070, 1073**
 - DownloadFile **method** (WebClient **class**), **1068**
 - DPToLP **Windows API function**, **645**
 - DrawArc **method** (Graphics **class**), **650**
 - DrawBezier **method** (Graphics **class**), **650**
 - DrawClosedCurve **method** (Graphics **class**), **650**
 - DrawCurve **method** (Graphics **class**), **650**
 - DrawEllipse **method** (Graphics **class**), **627, 650**
 - DrawImage **method** (Graphics **class**), **652–653, 655**
 - DrawImageUnscaled **method** (Graphics **class**), **652, 655**
 - Drawing
 - assembly, 523
 - namespace, 625, 632
 - Drawing2D **namespace**, **625, 650**
 - DrawLine **method** (Graphics **class**), **650**
 - DrawLines **method** (Graphics **class**), **650**
 - DrawPie **method** (Graphics **class**), **650**
 - DrawPolygon **method** (Graphics **class**), **650**
 - DrawRectangle **method** (Graphics **class**), **627, 635, 650**
 - DrawString **method** (Graphics **class**), **655, 656–657, 669**
 - DropDownList **control**, **884**
 - .dsw **files**, **312**
 - DTC (Distributed Transaction Coordinator)**, **1001**
 - DTD (Document Type Definition)**, **791–792. See also XML (Extensible Markup Language)**
 - DWORD **memory block**, **201**
- ## E
- elif **preprocessor directive**, **71–72, 1305**
 - else
 - conditional statement, 47
 - preprocessor directive, 71–72, 1305
 - else if **statement**, **47–48**
 - Employee **class**, **176, 1203–1208, 1210–1211, 1213, 1216–1217**
 - EmployeeData **class**, **245–246, 247, 251, 253–254**
 - EmployeeID **class**, **245–246, 247, 251–252, 255**
 - EnableCommit **method** (ContextUtil **class**), **1011**
 - EnableVisualStyle **method** (Application **class**), **575, 605, 606**
 - encryption. See also security; SSL (Secure Sockets Layer)**
 - assembly, 370, 372, 374–375, 428
 - certificate, 429–435, 557
 - CSP, 372
 - key
 - private, 370, 375, 430
 - public, 370, 374
 - symmetric, 370
 - EndEdit **method** (DataRow **class**), **709**
 - EndGetResponse **method** (WebRequest **class**), **1073**

endif preprocessor directive

endif **preprocessor directive, 71–72, 1305**

EndInit **method (ISupportInitialize interface), 616–617**

EndInvoke **method, 498**

EndOfStreamException **class, 279**

endregion **preprocessor directive, 72, 1305**

Enterprise Services

business service layer, 1000

client, 1008–1009, 1020–1021

COM+ application in Enterprise Services environment, 1003–1006

component load balancing, 1000

component, queuing, 1002

concurrency, 1007

context, 1001

data service layer, 1000

database connection, 1003, 1016–1019

debugging, 1007, 1021

dump location, specifying, 1007

event handling, 1002, 1004

history, 999–1000

identity, 1007

LCE service, 1002

message queue, 1007, 1015

pooling, 1002, 1003, 1007

presentation service layer, 1000

recycling, 1007

security, 1002, 1007

SOAP access, 1007

transaction

aborting, 1011, 1019

ACID, 1009

automatic, 1001, 1009, 1017

committing, 1011

consistent bit, 1010–1011

disabling, 1010

displaying active, 1020

distributed, 1001

done bit, 1010–1011

DTC, 1001

requiring, 1007, 1010, 1016

result, handling, 1010–1011

rollback, 1001

success, determining, 1010–1011, 1019

EnterpriseServices **assembly, 1003**

EntryPoint

class, 446

method, 168

Enum **class, 56, 57**

enumeration. *See also specific enumeration*

C#, 55–57

C++, 1256, 1296–1297

Java, 1232–1234

string, converting to/from, 56–57

Equals **method (Object class), 104, 143, 151,**

152, 250

equals sign (=) assignment operator, 132

equals sign, colon (=:) assignment operator, 34, 132

equals signs (==) comparison operator, 48, 132, 143,

151, 152

error handling. See also exception

array out-of-bounds error, 281

cast operation, 140

casting error, 140

COM, 971

compilation error message, specifying, 72

interoperability, 17

.NET environment, 16–17

Visual Basic, 278

Windows Form, 587–588

XML validation error, 792–794

error **preprocessor directive, 72, 1305**

ErrorProvider **Windows Form component, 587–588**

escape sequence, 44, 46, 229, 232

Evaluate **method (XPathNavigator class), 806**

event

ASP.NET event handling, 878, 879, 882, 899–901, 906–907

C++ event handling, 1303–1304

canceling event processing, 182, 183, 184, 185

class

creating handler class, 182, 184–185

member, event as, 85

COM application event handling, 972, 980–982,

986, 996

control event

ActiveX control, 984

Button control, 585, 885

Calendar control, 885, 898

CheckBox control, 884

custom control, 947–948, 959–962

CustomValidator control, 887

DataGrid control, 737, 902

DropDownList control, 884

HyperLink control, 885

ImageButton control, 885

ListBox control, 586

ListView control, 591

- Menu control, 596
- RadioButton control, 592, 885
- RadioButtonList control, 885
- Splitter control, 594
- StatusBar control, 594
- TabControl control, 595
- TextBox control, 587, 620–621, 884
- user control, 936, 938
- delegate, using in event handling, 168, 179, 180–181, 182
- file event, 1033, 1036–1037
- GDI+
 - input event, 671–672
 - paint event, 629, 630, 637–638, 642, 664
- keyboard event, 580–581, 672
- LCE, 1002
- menu, pup-up, 777
- mouse event, 180–182, 185, 580–581, 672
- parameter, 181, 184
- print event, 676, 677–679
- raising, 182–184
- receiver, 180–182
- remoting, 499–505
- sender, 180
- terminology used in this book, 180
- toolbar event, 597–598
- value returned by handler, 181
- Visual Studio .NET, viewing in, 322
- Web service, 918, 927, 928
- Windows Form event, 180, 575, 580–582, 585, 599–600
- Windows Service
 - event log, 1103, 1104, 1123–1128
 - power event, 1105, 1135
- XML
 - serialization event, 828, 830, 832, 833, 834
 - validation event, 792, 793
- EventArgs **class**, **182–183**
- EventHandler **delegate**, **180–181, 776**
- EventLog **class**, **1126–1127**
- EventLogEntry **class**, **1126**
- EventLogEntryCollection **class**, **1126**
- EventLogInstaller **class**, **1126**
- EventLogTraceListener **class**, **1126, 1129**
- EventSink **class**, **503, 504**
- eventTable **object**, **898**
- example **XML tag**, **68**
- exception**. *See also error handling*
 - application exception, 278, 296, 333
 - assembly not found exception, 343
 - C++ exception handling, 1256, 1297–1298
 - class
 - base class library, exception thrown by, 286
 - overview of exception classes, 278–280
 - user-defined, 290–297
 - ColdCallFileFormatException, 290, 294
 - end of stream exception, 279
 - file not found exception, 290, 292
 - IDisposable interface, deriving exception handler
 - from, 292–293
 - overflow exception, 132, 134–135, 279
 - remoting exception, 481
 - security exception, 407, 408, 415, 421
 - stream, reading, 293
 - system exception, 278
 - threading exception, 443–444
 - trapping using try catch finally block
 - array out-of-bounds error, 281–282
 - described, 280
 - inserting multiple catch clauses, 282–286
 - nesting try block, 287–290
 - omitting catch clause, 281
 - omitting finally clause, 281
 - parameter, passing to catch clause, 284
 - syntax, 280–281
 - throw statement, inserting, 281, 282
 - throwing exception from catch block, 289
 - throwing exception from finally block, 289
 - user input error, checking for using, 282
 - Visual Studio .NET, working with in, 333–334
 - XPath exception, 803
- Exception **class**
 - HelpLink property, 286, 287
 - hierarchy derived from, 278–279
 - InnerException property, 287, 289
 - instance, throwing, 284
 - Message property, 286, 287
 - Source property, 286
 - StackTrace property, 287
 - TargetSite property, 287
- exception **XML tag**, **68**
- Exchange Server**, **837**
- exclamation mark, equals sign (!=) comparison operator**, **132, 151, 152**
- exclamation mark (!) logical operator**, **132**
- ExecuteAssembly **method (AppDomain class)**, **343**
- ExecuteCommand **method (ServiceController class)**, **1121**

ExecuteNonQuery method

ExecuteNonQuery **method**

Command class, 694–695, 699

InsertCommand class, 900

OrderData class, 1018

ExecuteReader **method** (Command **class**), **694**, **695–696**, **701**, **702**

ExecuteScalar **method** (Command **class**), **694**, **696**

ExecuteXmlReader **method** (Command **class**), **696–698**

Exit **method** (Application **class**), **575**, **600**

ExitThread **method** (Application **class**), **575**

expression, regular

capture, 235–236

character

 matching, 232–233, 234–235

 repeating, 232

escape sequence, 229, 232

group, 234–235, 236

input string, 231

instantiating, 230

match, 231

namespace, 229, 231

pattern, 231

phone number, performing on, 230

quantifier, 232

result, displaying, 233–234

URL, performing on, 235–236

whitespace, matching, 232

word boundary, matching, 230, 232

word, matching, 233

expression **statement**, **903**

Extensible Markup Language. See XML

F

fat-client application, **24**

field

C++, 1280–1281

described, 35

instance, 38

naming, 76

primitive, 95

public, 80

readonly, 99–101

scope clash, 37–38

static, 38, 96, 1143–1145, 1208–1209, 1281

struct field, 102

value, default, 34

viewstate field, 875

XML serialization, 826

file. See also stream

binary file, reading/writing, 1041, 1042–1047

browser interface to file system, creating, 1030–1035

C++ file structure, 1261–1262

copying, 1028, 1035–1039

creating, 1028

creation date, reading into field, 100

deleting, 1028, 1035–1039

event handling, 1033, 1036–1037

moving, 1028, 1035–1039

name, changing, 1028

system, 1027–1028

text file

 reading, 661–662, 1040, 1048–1050, 1051–1054

 writing, 813, 1040, 1047, 1050–1054

time data, changing, 1029

tree view control, creating custom, 611–617

type association, establishing during application

 deployment, 564

WebClient class, uploading/downloading using,

 1068–1070

File **class**, **1026**, **1027**, **1035**

File Transfer Protocol. See FTP

FileAccess **enumeration**, **1042–1043**

FileInfo **class**

 CopyTo method, 1028

 Create method, 1028

 CreateText method, 1029

 CreationTime property, 1028

 Delete method, 1028

 DirectoryName property, 1028, 1035

 Exists property, 1028, 1034

 Extension property, 1028

 FullName property, 1028

 GetDirectories method, 1029

 GetFiles method, 1029

 GetFileSystemInfos method, 1029

 instantiating, 1027–1028

 LastAccessTime property, 1028

 LastWriteTime property, 1028

 Length property, 1028

 MoveTo method, 1028

 Name property, 1028

 Open method, 1028, 1043

 OpenRead method, 1029, 1043

 OpenText method, 1029

 OpenWrite method, 1029, 1043

 Parent property, 1028

 stream creating from FileInfo instance, 1043

 TreeView custom control, using in, 612–613

FileMode **enumeration**, **1042–1043**

- FileNode class, 612–613**
FileInfo class, 613
FileNotFoundException exception, 290, 292
FileShare enumeration, 1042–1043
FileStream class
 binary file, reading/writing using, 1042–1047
 constructor, 1043
 instantiating, 1047
 introduced, 293
 Read method, 1044
 ReadByte method, 1044, 1084
 Write method, 1044
 WriteByte method, 1044
 XML, reading/writing using, 789, 795, 808
FileSystemInfo class, 1026
FileView ⇄ Typelib (Visual Studio .NET), 987
Fill method
 DataSet class, 723, 816
 PCSWebApp3 application, 896
FillEllipse method (Graphics class), 650
FillPie method (Graphics class), 650
FillRectangle method (Graphics class), 650
FillResult method, 867
Filter method (ContextDataRow class), 772
Finalize method (Object class), 104, 105, 193, 194
 finally **block**. **See** try catch finally **block**
Find methods, 234
FindAll method (DirectorySearcher class), 863
FindConnectionPoint method (IConnection-PointContainer interface), 972
FindOne method (DirectorySearcher class), 863
FindString method (ListControl class), 586
FindStringExact method (ListControl class), 586
FireCompleted method, 981
First.exe file, 31
firstStringMethod delegate, 170
FlatStyle enumeration, 605–606
float class, 1187
folder
 browser application, listing in, 1034–1035
 content, listing, 1029
 copying, 1035–1039
 creating, 1028
 deleting, 1028, 1035–1039
 moving, 1028, 1035–1039
 name, changing, 1028
 time data, displaying, 1031
 tree view custom control, creating, 611–617
FolderNode class, 613–617
FolderTree class, 611–612
font. See text, font
Font class, 658
FontFamily class, 658, 661
FontStyle enumeration, 661
for statement, 51–53, 1220–1221
foreach statement, 54, 243, 1239–1240, 1265–1266
Form class
 AutoScrollMinSize property, 640, 651, 668
 BackColor property, 645
 ControlBox property, 604
 DialogResult property, 601, 603
 FlatStyle property, 605–606
 FormBorderStyle property, 604–605
 hierarchy, 583, 599
 Icon property, 604
 Invalidate method, 666–667, 674
 IsMdiContainer property, 607
 MaximizeBox property, 604
 MdiChildren property, 608, 609
 MdiParent property, 607
 MinimizeBox property, 604
 ShowDialog method, 600, 603–604
 ShowInTaskbar property, 600, 605
 StartPosition property, 600
 TopMost property, 600, 637
 Visible property, 599, 601
form language, localizing, 527–533
Format method (String class), 218, 223–225, 228
FormatException class, 285, 286, 294
FormBorderStyle enumeration, 604–605
Form1 class
 chooseColorDialog field, 1062
 constructor, 320, 577–578
 CreateGraphics method, 626–627
 Dispose method, 654
 downloading, 626
 InitializeComponent method, 320–321, 578, 579, 626
 ResumeLayout method, 579
 Show method, 627
 SuspendLayout method, 579
Forms namespace, 574, 582
Forms.dll assembly, 269
FormStartPosition enumeration, 600, 1063
FromArgb method (Color class), 646
FromFile method (Image class), 652
FromKnownColor method (Color class), 605

FTP (File Transfer Protocol), 1082

function

- abstract, 115, 123, 127, 1165–1168
- C++, 1254, 1255, 1285–1287
- callback, 167
- calling, 114–115
- class member, as, 85
- constructor, relation to, 86
- method, relation to, 86
- output parameter, 91
- overriding, 112
- pointer, 167
- property, relation to, 86
- struct, defining for, 101
- virtual, 112, 115, 1285–1287

G

GAC (global assembly cache)

- control, placing in, 931
- image, native, 366–367
- installing/uninstalling assembly, 368, 556
- shared assembly, GAC role in, 19, 341
- viewing, 367–368

gacutil **utility, 368, 376**

garbage collection

- assembly runtime, 377
- code, running from, 13, 193
- COM, 13, 966–967
- heap, action on, 192–193
- introduced, 13–14
- LDGC, 481
- reference object, 35, 45, 105
- remoting, 481
- Visual Basic 6, 1197

GC **class, 193**

GC (global catalog), 841–842, 850

GDI+. See also specific member class

- area, measuring, 632–636
- BitBlt, 655
- brush, 648–649
- clipping region, 630–632, 669
- color, 627, 645–647
- coordinate, 632–636, 644–645, 670–671
- DC, 625–626
- debugging, 637–638
- drawing, 624–629, 638–645, 655–657
- event handling
 - input event, 671–672
 - paint event, 629, 630, 637–638, 642, 664

- GDI, relation to, 623
- gradient, 649
- hatch, 648–649
- image, working with, 652–655
- input, 671–675
- namespace, 625
- painting
 - calling painting routine directly, 666
 - DrawString method involvement, 669
 - event handling, 629, 630, 637–638, 642, 664
 - Invalidate method, using, 666–667, 674
 - shape, 629–630
- pen, 627–628, 649
- printing, 675–680
- region, 636
- scrolling behavior, 638–644
- size of item, calculating, 667–668
- text
 - case, converting, 661–666, 674
 - drawing, 655–657
 - font, 657–661
 - line index, 670–671
 - reading text file, 661–662

GDI (graphical device interface), 623, 653

Generate **method (ResourceWriter class), 523**

GenericCustomer **class, 117, 118–121**

Get

- accessor, 93, 94, 1151
- HTTP method, 915
- OleDbDataReader class Get methods, 703

GetAssemblyFullName **method (SharedDemo class), 379**

GetAString **delegate, 169, 170, 172**

GetAttribute **method (XmlTextReader class), 791**

GetChanges **method (DataSet class), 729**

GetChannel **method (ChannelServices class), 471**

GetChildControlType **method (ControlBuilder class), 955**

GetChildRows **method (DataRow class), 709**

GetConstructor **method (Type class), 268**

GetConstructors **method (Type class), 268**

GetCultures **method (CultureInfo class), 517**

GetCustomAttributes **method (Attribute class), 272, 275**

GetData **method**

- CallContext class, 506
- Web service application, 919, 920

GetDevices **method (ServiceController class), 1117**

- GetDirectories **method**
 - DirectoryInfo class, 1029
 - FileInfo class, 1029
- GetDirectoryEntry **method** (SearchResult class), **863**
- GetEnumerator **method**
 - collection iteration, in, 1240
 - IEnumerable interface, 242, 244, 245, 542
 - IResourceReader interface, 541, 542
- GetEvent **method** (Type class), **268**
- GetEvents **method** (Type class), **268**
- GetExecutingAssembly **method** (Assembly class), **525**
- GetField **method** (Type class), **268**
- GetFields **method** (Type class), **268**
- GetFiles **method**
 - DirectoryInfo class, 1029
 - FileInfo class, 1029
- GetFileSystemInfos **method**
 - DirectoryInfo class, 1029
 - FileInfo class, 1029
- GetFormat **method** (IFormatProvider interface), **514**
- getFreeDate **method**, **897, 898**
- GetHashCode **method** (Object class), **104, 151, 249, 250, 252–253**
- GetIDsOfNames **method** (IDispatch interface), **967–968**
- GetInterface **method** (Type class), **268**
- GetInterfaces **method** (Type class), **268**
- GetLength **method**, **1276**
- GetLifetimeService **method**, **481, 482–483**
- GetManifestResourceNames **method** (Assembly class), **525**
- GetMember **method** (Type class), **268**
- GetMembers **method** (Type class), **268, 270**
- GetMethod **method** (Type class), **267**
- GetMethods **method** (Type class), **267–268**
- GetMonthlyPayment **method**
 - Employee class, 1211, 1213
 - Manager class, 1212
- GetMonthlySalary **method** (Employee class), **1207**
- GetMyRemote **method**, **478**
- GetMySerialized **method**, **478**
- GetNextTextElement **method** (StringInfo class), **511**
- GetObject **method**
 - Activator class, 472, 473–474, 476
 - ResourceManager class, 526, 530
- GetParentRow **methods** (DataRow class), **712**
- GetProperties **method**
 - Active Directory user class application, 867
 - Type class, 268
- GetQuoteOfTheDay **method** (SharedDemo class), **371–372**
- GetRandomQuoteOfTheDay **method** (QuoteServer class), **1098**
- GetRealProxy **method** (RemotingServices class), **474**
- GetResponse **method** (WebRequest class), **1071–1072, 1073, 1077**
- GetRowType **method**, **774**
- GetSchemaProperties **method**, **866**
- GetServices **method** (ServiceController class), **1117**
- GetServiceTypeName **method** (ServiceController class), **1118, 1120**
- GetSocketOption **method** (Socket class), **1088**
- GetStream **method** (NetworkStream class), **1083**
- GetString **method** (ResourceManager class), **526**
- GetSubKeyNames **method** (RegistryKey class), **1059**
- GetTextElementEnumerator **method** (StringInfo class), **511**
- GetType **method** (Object class), **104, 105, 266, 271, 1161**
- GetTypes **method** (Assembly class), **271, 274**
- GetValue **method** (RegistryKey class), **1058, 1059, 1065**
- GetValueNames **method** (RegistryKey class), **1059**
- global assembly cache. See GAC**
- global catalog. See GC**
- global.asax file, 539, 876**
- Global.asax.cs file, 549, 919–920**
- Global.asax.resx file, 549**
- globalization, 509, 539–540. See also localization**
- Globalization namespace, 510**
- globally unique identifier. See GUID**
- GoldAccount class, 127**
- goto statement, 49, 54–55**
- graphical device interface. See GDI**
- Graphics class**
 - DrawArc method, 650
 - DrawBezier method, 650
 - DrawClosedCurve method, 650
 - DrawCurve method, 650
 - DrawEllipse method, 627, 650
 - DrawImage method, 652–653, 655
 - DrawImageUnscaled method, 652, 655
 - DrawLine method, 650

Graphics class (continued)

Graphics class (continued)

- DrawLines method, 650
- DrawPie method, 650
- DrawPolygon method, 650
- DrawRectangle method, 627, 635, 650
- DrawString method, 655, 656–657, 669
- FillEllipse method, 650
- FillPie method, 650
- FillRectangle method, 650
- introduced, 626
- MeasureString method, 667, 668, 674
- PageUnit property, 645
- Size property, 653
- TranslateTransform method, 643
- greater than sign (>) comparison operator, 132, 151**
- greater than sign, equals sign (>=) comparison operator, 132, 151**
- greater than signs (>) bit shifting operator, 132**
- greater than signs, equals sign (>=) assignment operator, 132**
- Greeting method, 463, 498, 506–507, 973–974, 976–977
- GreetingDelegate **delegate, 498**
- GregorianCalendar **class, 518**
- GregorianCalendarTypes **enumeration, 518**
- Group **class, 236**
- GroupCollection **class, 236**
- GUID (globally unique identifier), 369, 852, 976, 990**
- guidgen **utility, 976**

H

- HasAttributes **method (XmlTextReader class), 788**
- hash code, 247, 248–249, 250, 252–253, 349–350**
- Hashtable **class**
 - Add method, 247
 - Count property, 248
 - dictionary representation in, 247, 469
 - iterating through Hashtable collection, 1240
- HasValue **method (XmlTextReader class), 788**
- HatchBrush **class, 648**
- HatchStyle **enumeration, 648**
- Heart **method, 937**
- Hello
 - class, 363, 463
 - method
 - HelloMCP class, 357, 359
 - HelloVB class, 361, 362
- HelloClient **class, 498**

- HelloCSharp **class, 363**
- HelloMCP **class, 357, 358, 359**
- HelloMCP.dll **file, 360**
- HelloServer.cs **file, 468, 482**
- HelloVB **class, 361, 362**
- HelpProvider **Windows Form component, 588–589**
- HisBaseClass **class, 113–114**
- HitTest **method (DataGrid class), 778**
- HTTP (Hypertext Transfer Protocol)**
 - channel, 466–467, 468
 - GET method, 915
 - header, 1071–1072
 - POST method, 915
 - remoting using SOAP and HTTP, 456–457, 469, 470, 495–498
 - request handling, 915–916, 1070, 1071–1072, 1077
 - SOAP use of, 909, 910–911
- HttpApplication **class, 906–907**
- HttpChannel **class, 468**
- HttpRequest **class, 1071, 1077**
- HttpResponse **class, 1071, 1077**
- Hungarian notation, 75, 126**
- HWND (Windows handle), 582**
- HyperLink **control, 885**
- Hypertext Transfer Protocol. See HTTP**

I

- IADsServiceOperations **interface, 859**
- IADsUser **interface, 859, 860**
- IANA (Internet Assigned Number Authority), 1083**
- IAAsyncResult **interface, 1073**
- IBank **interface, 125**
- IBankAccount **interface, 126–128**
- IChannel **interface, 467, 468**
- IChannelReceiver **interface, 468, 470**
- IChannelSender **interface, 468, 470, 475**
- IClientChannelSinkProvider **interface, 470**
- ICollection **interface, 240, 242, 892**
- IComparer **interface, 175**
- ICompletedEvents **interface, 972, 980**
- IconLocation **method, 587**
- IConnectionPointContainer **interface, 972**
- IContextProperty **interface, 461**
- IContributeXXSink **interface, 461**
- Identity
 - interface, 417
 - SQL statement, 730, 731–732, 762
- IDictionary **interface, 475**

- IDispatch **interface, 967–968, 977**
- IDisposable **interface**
 - Dispose method, 110, 124–125, 195–197, 542, 577
 - exception handler, deriving from, 292–293
 - try catch finally block, inserting in, 195
- IEnumerable **interface**
 - collection IEnumerable interface requirement, 54, 242
 - control, binding to, 892
 - GetEnumerator method, 242, 244, 245, 542
 - object implementation, querying for, 125
 - Vector struct, 244
- IEnumerator **interface**
 - Current property, 242, 243
 - MoveNext method, 242, 1170, 1171
 - Reset method, 242, 1170, 1171
 - VectorEnumerator class implementation of, 245
- if
 - conditional statement, 47–49, 1188–1189, 1264
 - preprocessor directive, 71–72, 1305
- IFormatProvider **interface, 225, 513–514, 515**
- IFormattable **interface, 223, 225, 227, 244**
- IGreeting **interface, 1004**
- IL (intermediate language), 4, 8–10, 12**
- ilasm **utility, 525**
- ildasm **utility, 347–348**
- ILease **interface, 481, 482**
- IList **interface, 746**
- IListSource **interface, 746, 892**
- ILogicalThreadAffinative **interface, 506**
- Image
 - class
 - Dispose method, 654
 - Drawing assembly, referencing, 523
 - FromFile method, 652
 - GAC native image cache, 366–367
 - control, 884
- ImageButton **control, 885, 937–938**
- ImageList **Windows Form component, 585, 589, 590–591**
- Imaging **namespace, 625**
- IMath **interface, 976, 986, 988, 994**
- IMathEvents **interface, 995, 996**
- IMathPtr **interface, 994**
- IMessage **interface, 475**
- IMessageSink **interface, 468, 476**
- IMethodCallMessage **interface, 475**
- IMethodMessage **interface, 475**
- import **directive (Java), 1226, 1227**
- INamingContainer **interface, 949, 953**
- include
 - C++ include statement, 1259
 - XML tag, 68
- Increment **method**
 - PerformanceCounter class, 1133
 - ProgressBar control, 592
- indexer, 86, 153, 1291**
- IndexOf **method (String class), 218**
- IndexOfAny **method (String class), 218**
- IndexOutOfRangeException **class, 284–285, 286**
- inheritance**
 - C++, 1257, 1258, 1285
 - constant, 1289
 - cross-language, 354, 361
 - implementation inheritance, 109–110, 111–112, 1152, 1195
 - interface inheritance, 109–110, 123–130, 1152, 1170–1171, 1195
 - Java, 1245
 - multiple inheritance, 110, 1161, 1258
 - polymorphism, 1161–1164
 - sealing, 115–116, 1169–1170
 - struct, 103, 110–111
 - Visual Basic 6, 1178, 1194–1195, 1210–1212
 - Windows Form, 109–110
- Init **method, 616**
- InitializeComponent **method, 1103**
- InitClass **method (ProductDataTable class), 718, 719, 770**
- InitializeComponent **method**
 - BookOfTheDayForm class, 528, 530
 - Form1 class, 320–321, 578, 579, 626
 - PCSWebApp3 application, 895, 919
 - ServiceProcessInstaller class, 1110
 - SquareRootForm class, 1199–1200
 - straw poll application, 961
 - WebForm1 class, 880, 882
- InitializeLifetimeService **method (MarshalByRefObject class), 484**
- InitializeTree **method, 617**
- InitilizeComponent **class, 617**
- inlining, 94**
- Insert
 - method
 - IOrderUpdate interface, 1016
 - OrderData class, 1017
 - OrderLineData class, 1019
 - StringBuilder class, 222
 - Vectors array list, 240
 - SQL statement, 694, 724, 761, 1018

InsertCommand class

InsertCommand **class**, 724, 900

Install **method** (Installer **class**), 1108

InstalledFontCollection **class**, 659, 660

installer

Add/Remove Programs dialog box, application display in, 1055

bootstrapper option, 557

cab files, 551, 557

compression, 557

deployment project, 553

launch condition, specifying, 568–569

middle-tier application, 551

module, 551

package, building, 552, 557

registry entry, adding, 562–563

release build, 552

rollback option, 550, 566

shortcut to application, creating, 556–557, 562

uninstall option, 550, 555, 558, 566

user interface, 553–555, 564–566

Visual Studio .NET installer project, 551

Web application, 551, 559–560

Windows Installer, 551–552

wizard, creating, 551

Installer **class**, 1108

installutil **utility**, 1112–1113, 1127

interceptor, 461

interface. *See also specific interface*

C++, 1300–1302

COM, 124, 967–968, 989, 990, 991

custom, 967

defining, 125–128

derived, 128–130

dispatch interface, 967–968, 989

dual interface, 968, 988

IL support, 8

inheritance, 109–110, 123–130, 1152, 1170–1171, 1195

naming, 126

referencing, 128

remoting client and server code, separating using, 496

intermediate language. *See* IL

InternalGetResourceSet **method**
(ResourceManager **class**), 543

Internet Assigned Number Authority. *See* IANA

Internet Protocol address. *See* IP address

Interop.COMServer.dll **file**, 978

interoperability, 5–7, 8–9, 10–12, 17, 965–966. *See also* COM

InteropServices **namespace**, 351, 352

IntPtr **class**, 198

Invalidate **method**

Control class, 582

Form class, 666–667, 674

Inventory **class**, 831, 832–834, 835

Invoke **method**

DirectoryEntry class, 859

IDispatch interface, 967–968

RealProxy class, 474

Type class, 268

IO **namespace**, 278, 1025, 1031

IOException **class**, 278

IOrderUpdate **interface**, 1016

IP (Internet Protocol) address, 1079–1080

IPAddress **class**, 1080

IPEndPoint **class**, 1084, 1087

IPHostEntry **class**, 1080

IRemotingFormatter **interface**, 470

IResourceReader **interface**, 527, 541, 542

IResourceWriter **interface**, 527

IServerChannelSinkProvider **class**, 470

IsPasswordCorrect **method** (Authenticator **class**), 1141, 1146

ISponsor **interface**, 480, 482

IsStartElement **method** (XmlTextReader **class**), 788

IsStyleAvailable **method** (FontFamily **class**), 661

IsTransparentProxy **method** (RemotingServices **class**), 474, 478

ISupportErrorInfo **interface**, 971

ISupportInitialize **interface**, 616–617

ITransferBankAccount **interface**, 128–129

IUnknown **interface**, 967, 968, 977

IWelcome **interface**, 974, 975, 976, 986, 988

IXPathNavigable **interface**, 807, 808

J

Java

access modifier, 1241

array, 1226, 1236

case sensitivity, 1226

class, 1240, 1241–1243, 1245–1249

collection, 1236

compilation, 1226

data type, 1230–1236

destructor, 1245

entry point, 1226

enumeration, 1232–1234

flow control, 1238–1240
 import directive, 1226, 1227
 inheritance, 1245
 input/output, 1249–1251
 iteration, 1238–1240
 method, 1226, 1242, 1243–1244, 1247–1249
 operator, 1237–1238
 package, 1226–1229
 property, 1244–1245
 structure, 1234–1236
 variable, 1229–1230

JavaEcho **class**, 1249

JIT (Just-In-Time) compilation, 5, 94

Join **method**, 444

JUMP (Java User Migration Path), 315

jumping to another line in program, 49, 54–55

K

Kerberos, 839

keyfile **utility**, 978

KnownColor **enumeration**, 605, 646

L

Label **control**

ASP.NET Web Form, 881, 884, 891
 custom control, deriving from, 945–946
 Windows Form, 589, 618

LandLineSpyFoundException **class**, 291, 295–296

language, translating automatically based on culture, 534–539

LastIndexOf **method** (String **class**), 218

LastIndexOfAny **method** (String **class**), 218

LastModifiedAttribute **class**, 263, 275

LCE (loosely coupled event), 1002

LDAP (Lightweight Directory Access Protocol), 838, 850

LDGC (Leasing Distributed Garbage Collector), 481

ldstr **statement**, 362

lease, 481–484

less than sign (<) comparison operator, 132

less than sign, equals sign (<=) comparison operator, 132, 151

less than sign, percent sign (<%) ASP.NET code block prefix, 876

less than signs (<<) bit shifting operator, 132

less than signs, equals sign (<<=) assignment operator, 132

Lifetime **namespace**, 482

LifetimeServices **class**, 482, 483–484

Lightweight Directory Access Protocol. See LDAP

line **preprocessor directive**, 72–73, 1305

LinearGradientBrush **class**, 649

LineIndexToWorldCoordinates **method**, 670

LinkButton **control**, 885

Linker ⇄ Input (Visual Studio .NET), 356

list **XML tag**, 68

ListBox **control**, 586, 884, 1068

listBox1_SelectedIndexChanged **method**, 814

ListControl **class**, 586

Listen **method** (Socket **class**), 1088

Listener **method** (QuoteServer **class**), 1098, 1133

ListItem **object**, 884, 889–890

ListItemCollection **class**, 884, 890

Literal **control**, 884

Load **method**

Assembly class, 271

XmlDocument class, 814

XslTransform class, 811

LoadBook **method**, 805

LoadFile **method**

GDI+ application, 665, 666, 668, 676

TextBox control, 593

LoadFrom **method** (Assembly **class**), 271, 561

LoadLibrary **method**, 340

LoadList **method**, 790

LoadTree **method**, 616

.local **files**, 340

localization

ASP.NET, 539–540

calendar, 510, 518

culture

assembly, 346, 352, 527

calendar, 518

date format, 512, 515–516

default, specifying, 512–513

introduced, 511

invariant, 512

language, translating automatically based on, 534–539

listing all cultures, 516–520

neutral, 512

number format, 512, 513–515

operating system language, dependent on, 512

resource for, returning, 527

resource reader based on, 540–544

sorting order, 520–522

specific, 511–512, 519

text format, 510–511

localization, culture (continued)

- threading, 444, 512
- user interface, 512
- described, 509
- form language, 527–533
- number formatting, 513–515
- resource file
 - accessing, 525–526
 - assembly, adding to, 524–525
 - binary, 536–537
 - creating, 522–524, 531, 532
 - language translation using, 528–533, 536–538
 - reader, creating custom, 540–544
 - XML, 537–538
- satellite assembly, 533, 539, 540
- sorting, 520–522
- Unicode considerations, 510–511

Localization namespace, **528**

lock, mutual exclusion, **450**

logging

- event log access permission, 399
- Web application, 560
- Windows Service event log, 1103, 1104, 1123–1128

Login method, **926, 928**

Login.cs file, **689**

LongWorking method, **501, 505**

LookupWhatsNew assembly, **262, 272–273**

looping

- do while loop, 53, 1264
- for loop, 51–53, 1220–1221
- foreach loop, 54, 243, 1239–1240, 1265–1266
- message loop, 574–575, 1192, 1198
- method delegate, loop operation involving, 173, 174
- while loop, 53, 1264

loosely coupled event. *See* LCE

LPToDP Windows API function, **645**

M

Main

method

- argument, passing to, 62–63
- assembly, 342–343
- C++, 1260, 1262
- calling, 33, 61, 167
- compilation considerations, 61–62
- entry point, as, 32–33, 439, 1142
- MainClass class, 270
- MainEntryPoint class, 1142
- multiple, using, 61–62

- permission, denying in, 412
- Startup class, 607
- threading, starting in, 441, 445
- ThreadPlayaround class, 445
- Visual Studio .NET, created by, 304, 308
- WhatsNewChecker class, 274
- Windows Form, 574, 605
- Windows Service, 1104, 1105
- Visual Basic 6 function, 1195–1196

MainClass class, **270**

MainEntryPoint class, **1196**

MainEntryPoint class, **1141–1142, 1145**

MainExample.cs file, **62**

MainMenu class, **595, 596, 608**

makecert utility, **430**

managed code, **4–7**. *See also* CLR (Common Language Runtime)

Manager class, **1210–1211, 1212, 1214–1215, 1216–1217**

ManufactureProductDataTable method, **711**

map. *See* dictionary

Marshal class, **978, 979**

MarshalByRefObject class, **461, 462–463, 484, 494, 1026**

MatchCollection object, **231**

Matches method (Regex class), **231, 236**

MathLibrary.cs file, **64–65**

MathOperations class, **172, 173, 178**

MathTest class, **87–89**

MathTest.cs file, **88**

Math.xml file, **69**

Matrix statement, **144**

MDI (Multiple Document Interface), **599, 607–609**

MeasureString method (Graphics class), **667, 668, 674**

MemberInfo object, **269, 270**

MemberwiseClone method (Object class), **104, 105**

memory

- address
 - hexadecimal format, displaying in, 208, 212
 - virtual, 188
- allocating, 190–191, 212
- application domain environment, in, 14–16
- buffer, 1042
- C++ memory management, 1255, 1273–1275, 1298–1300
- CLR memory type safety, 7, 9–10
- data type storage, 39, 188–192
- DWORD block, 201

- garbage collection
 - assembly runtime, 377
 - code, running from, 13, 193
 - COM, 13, 966–967
 - heap, action on, 192–193
 - introduced, 13–14
 - LDGC, 481
 - reference object, 35, 45, 105
 - remoting, 481
 - Visual Basic 6, 1197
 - heap, managed, 39, 190–191, 192–193, 1299–1300
 - location, accessing, 167, 188–189
 - method, memory implementation of static, 1145–1147
 - pointer
 - arithmetic operation on, 203–204
 - array, declaring pointer to, 202
 - array syntax, applying to pointer, 214
 - C++, 1255, 1298–1300
 - class, 202, 206–207, 209–212
 - class member, to, 206–207, 209–212
 - CLR memory type safety considerations, 9
 - compatibility, backward, 198
 - declaring, 200–202
 - function pointer, 167
 - performance optimization, using for, 198, 212–215
 - safety, 198–200
 - stack pointer, 188–189
 - struct, 205–206, 209–212
 - syntax, 200
 - void, 203
 - process memory access, 14–15
 - processor, 32-bit, 201
 - referent, 198
 - stack
 - allocating stack memory, 212
 - array, stack-based, 212–215, 1300
 - data stored in, 188
 - heap versus, 191
 - introduced, 39
 - overflow, 132, 134–135, 198, 202, 279
 - pointer, 188–189
 - variable scope considerations, 188–189
 - stream, 813, 1040, 1042
 - string memory allocation, 45–46, 219–221, 222
 - struct memory allocation, 103
 - threading, 449
 - type-safe, 16
 - virtual, 188
- MemoryStream class, 813, 1040**
- menu**
 - control, 595–596, 608
 - database row pop-up menu, 770–778
 - Windows Form, 595–596, 608
 - Menu class, 595**
 - menuFileExit_Click method, 664**
 - menuFileOpen_Click method, 664**
 - MenuItem class, 595, 596**
 - Merge method (DataSet class), 729**
 - message loop, 574–575, 1192, 1198**
 - MessageBox**
 - class, 269, 1260–1261
 - Windows API function, 1260
 - method**
 - abstract, 122, 1166, 1242
 - C++, 1260, 1262, 1275–1276, 1302–1303
 - calling, 1143
 - COM method binding, 969, 989
 - declaring, 33, 86–87, 95
 - delegate
 - array, 173, 174–175
 - C++, 1302–1303
 - class, relation to, 168, 169, 178
 - defining, 169
 - described, 167–168, 1257
 - event handling, in, 168, 179, 180–181, 182
 - instantiating, 170
 - loop operation involving, 173, 174
 - multicast, 177–179
 - syntax, 169
 - thread, starting, 168
 - type safety, 170
 - exiting, 33
 - hiding, 113–114, 1164–1165, 1248–1249
 - inheritance, sealing to, 115–116
 - instance method, 86
 - invoking, 87–89
 - Java, 1226, 1242, 1243–1244, 1247–1249
 - name, CLS compliance, 366
 - overloading
 - C#, 91–92, 95, 1148–1150
 - C++, 1276
 - Visual Basic 6, 1215–1216
 - overriding, 45, 1157, 1162, 1213, 1247–1249
 - parameter, passing to, 89–91, 1243–1244
 - private, 86
 - property versus, 80
 - public, 86
 - signature, 113, 123, 1164
 - static method, 86, 1145–1147

method (continued)

method (continued)

unsafe, marking as, 199, 1298–1299
variable, passing to, 89–90
virtual, 112–113, 1162–1164
Visual Basic 6 subroutine versus, 1184–1185

Microsoft Transaction Server. See MTS

minus sign (-) arithmetic operator, 132

minus sign, equals sign (=) assignment operator, 132

minus sign, greater than sign (->) indirection operator, 132

minus signs (—) increment operator, 132

MMC (Microsoft Management Console)

Active Directory Domains and Trusts snap-in, 845
Active Directory Sites and Services snap-in, 845
Active Directory Users and Computers snap-in, 845–846
.NET Framework Configuration snap-in, 380–381, 385, 492

mobile code, 391

module

class module, 1138
creating, 349–350
installer module, 551

Monitor class, 450

Mono project, 5

MoveNext method

collection iteration, in, 1240
IEnumerator interface, 242, 1170, 1171
XPathNodeIterator class, 804, 805

MoveTo method

Directory class, 1035
DirectoryInfo class, 1028
File class, 1035
FileInfo class, 1028
XPathNavigator class, 803

MoveToAttribute method (XPathNavigator class), 803

MoveToContent method (XmlTextReader class), 789

MoveToFirst method (XPathNavigator class), 803

MoveToFirstAttribute method

XmlTextReader class, 791
XPathNavigator class, 803

MoveToFirstChild method (XPathNavigator class), 804

MoveToId method (XPathNavigator class), 804

MoveToLast method (XPathNavigator class), 804

MoveToNext method (XPathNavigator class), 804

MoveToNextAttribute method

XmlTextReader class, 791
XPathNavigator class, 803

MoveToParent method (XPathNavigator class), 804

MoveToPrevious method (XPathNavigator class), 804

MoveToRoot method (XPathNavigator class), 804

mscorlib.dll file, 993
mscorlib assembly, 455
MSI files, 375
msvcrt.lib file, 356
MSXML parser, 783–786

MTA (multi-threaded apartment), 970–971

MTS (Microsoft Transaction Server), 999

MulticastDelegate class, 178

Multiple Document Interface. See MDI

multitasking, pre-emptive, 441

mutex, 450

mutual exclusion lock, 450

MxDataGrid control, 334
MyClass destructor, 194
MyFirstClass class, 32
MyForm class, 574
mykeypub.snk file, 375
mykey.snk file, 372
mypolicy.config file, 383

N

namespace. See also specific namespace

Active Directory domain namespace, 841
ADO.NET, 686
aliasing, 60–61
assembly namespace, 346–347
C++, 1259–1260
class
defining within namespace, 58
returning class namespace, 60
contiguous, 841
convention, 21, 60, 77, 370
declaring, 32
expression, regular, 229, 231
GDI+ class set, 625
header file, confusion with, 60
Java package compared, 1226–1229
nesting, 58–59
.NET Framework, 21
object, defining within, 60
referencing, 59–60
reflection data type namespace, returning, 266, 268–271
Visual Basic 6, 1192–1194
Windows Form, 574

NamespaceExample **class**, 59, 60
 naming convention, 21, 60, 74–79, 732–733, 1230
 NativeMethods **class**, 1104
 Navigate **method**, 984
 .NET Admin tool, 380, 388
 .NET Framework Configuration MMC snap-in, 380–381, 385, 492
 .NET namespace, 456
 .NET runtime. *See* CLR (Common Language Runtime)
 .NET Services Installation Tool, 1005
 NetComponent **class**, 986
 net.exe **utility**, 1114–1115
 NetworkStream **class**, 1040, 1083
 Nevermore60Customer **class**, 117–121, 190, 1156–1159, 1161–1165, 1168–1169
 New ⇨ Project (Visual Studio .NET), 312
 new **statement**, 35, 84, 102, 123, 1274–1275, 1279
 newInv **object**, 833
 NewOrder **method** (OrderControl **class**), 1015, 1020
 NewProductRow **method**, 768
 NewRow **method** (DataTable **class**), 720, 724, 768
 NewRowFromBuilder **method**, 720, 774
 ngen **utility**, 366–367
 Norm **method**, 228
 NTD (No Touch Deployment), 341, 561–562
 number, formatting for culture, 512, 513–515
 number sign (#) preprocessor directive prefix, 70
 NumberFormatInfo **class**, 514
 NumberTable.cs **file**, 52–53

O

object

Active Directory
 collection, 855–856
 creating, 857–859
 information about, returning, 854
 storage, 839, 855–856
 user object, 864–868
 application domain-bound object, 477
 ASP.NET object model, 878
 C++ object model, 1253, 1261
 class, relation to, 1139
 client-activated
 client configuration, 488–489
 creating, 473–474
 described, 464
 leasing mechanism, 482–483
 lifetime management, 481
 proxy, returning, 472

 remoting configuration file, setup in, 485, 486, 488–490
 server configuration, 472, 488
 COM object model, 8–9
 context, 460–462
 copying, 105
 data binding, object used in, 751
 data type compatibility, checking for, 135–136
 described, 1138–1139
 destructor
 C#, 193–194, 195–197, 1171–1172
 C++, 1256, 1283–1284
 Java, 1245
 equality, comparing objects for, 105, 142–144
 identity, distributed, 463
 IEnumerable interface implementation, querying
 for, 125
 indexer, 86, 153, 1291
 instantiating, 442, 444, 1141–1143
 namespace, defining within, 60
 proxy object, 474–475
 reference object, 35, 45, 105
 remote, 458, 462–463, 464, 472–480
 serialized, 480
 single-call, 464
 stateless, 472
 string representation, returning, 105–107
 thread object
 instantiating, 442, 444
 referencing, 444
 value literal, treating as object, 141
 Visual Studio .NET, browsing using, 324–325
 well-known, 464, 471, 472, 473, 487–488

Object class
 derivation from, 44–45, 103, 104, 1161
 Equals method, 104, 143, 151, 152, 250
 Finalize method, 104, 105, 193, 194
 GetHashCode method, 104, 151, 249, 250, 252–253
 GetType method, 104, 105, 266, 271, 1161
 MemberwiseClone method, 104, 105
 ReferenceEquals method, 104, 142–143, 144
 referencing, 112
 ToString method, 104, 105–107, 170, 225–226, 227–228

ObjectCollection object, 586
ObjectDisposedException object, 293
ObjRef class, 471
Odbc namespace, 686
ODBCCommand class, 687
ODBCCommandBuilder class, 687

ODBCConnection class

- ODBCConnection **class**, **687**
- ODBCDataAdapter **class**, **687**
- ODBCDataReader **class**, **687**
- ODBCParameter **class**, **687**
- ODBCTransaction **class**, **687**
- OleDb **namespace**, **686, 965**
- OleDbCommand **class**, **687, 701, 702, 900**
- OleDbCommandBuilder **class**, **687**
- OleDbConnection **class**, **687, 894**
- OleDbDataAdapter **class**, **687, 895, 896**
- OleDbDataReader **class**, **687, 701, 702, 703**
- OleDbParameter **class**, **687**
- OleDbTransaction **class**, **687**
- OnAction **method** (ActionCancelEventArgs **class**), **184, 186**
- OnClick **method**, **672**
- OnClickChooseColor **method**, **1062**
- OnContinue **method**, **1105, 1106**
- OnCustomCommand **method**, **1105**
- OnDoubleClick **method**, **672, 673**
- OnEntryWritten **method**, **1129**
- OnInit **method** (WebForm1 **class**), **880**
- OnKeyDown **method**, **672**
- OnKeyPress **method**, **672**
- OnKeyUp **method**, **672**
- OnMouseDown **method**, **672, 673**
- OnMouseHover **method**, **672**
- OnMouseMove **method**, **672**
- OnMouseUp **method**, **672**
- OnPaint **method**, **629–630, 631–632, 637–638, 641–642, 668–670**
- OnPause **method**, **1105, 1106**
- OnPowerEvent **method**, **1105, 1135**
- OnSelectCulture **method**, **518**
- OnSelectedIndexChanged **method** (ServiceController **class**), **1120**
- OnShutdown **method**, **1106**
- OnStart **method**, **1104, 1106**
- OnStatusChange **method**, **984**
- OnStop **method**, **1105, 1106**
- OnVoted **method**, **960**
- Open **method**
 - ColdCallFileReader class, 292, 293
 - FileInfo class, 1028, 1043
 - PCSWebApp3 application, 895
- OpenFileDialog **method**, **662, 1044**
- OpenRead **method**
 - FileInfo class, 1029, 1043
 - WebClient class, 1068
- OpenStandardError **method** (Console **class**), **1250**
- OpenStandardInput **method** (Console **class**), **1250**
- OpenStandardOutput **method** (Console **class**), **1250**
- OpenSubKey **method** (RegistryKey **class**), **1058, 1059, 1064, 1065**
- OpenText **method** (FileInfo **class**), **1029**
- OpenWrite **method**
 - FileInfo class, 1029, 1043
 - WebClient class), 1069
- operator**
 - addition, 137, 145–146, 147–148, 151
 - address, 132
 - as, 136
 - assignment
 - C#, 132, 137, 153
 - C++, 1278
 - Visual Basic 6, 1222, 1223
 - bitwise, 132, 137, 153, 200
 - C++, 1256, 1277–1279, 1289–1291
 - comparison, 132, 137, 142–144, 151–153
 - conditional, 132, 134
 - decrement, 132
 - division, 137
 - increment, 132
 - indexing, 132, 153
 - indirection, 132
 - instanceof, 1238
 - is, 135–136
 - Java compared, 1237–1238
 - logical, 132
 - member access, 132
 - modulus, 132
 - multiplication, 137, 144, 149–150
 - object creation, 132
 - overflow exception control, 132, 134–135
 - overloading, 86, 144–153, 218, 1256, 1289–1291
 - postfix, 133–134
 - precedence, 137
 - prefix, 133–134
 - shortcut list, 133
 - sizeof, 136, 204–205, 210, 1238
 - string concatenation, 132
 - subtraction, 137, 151
 - ternary, 134, 137, 1223, 1278
 - type information, 132
 - typeof, 57, 136, 1238
 - unary, 137, 153, 200
 - Visual Basic 6, 1222–1224
- Oracle **namespace**, **686, 687**
- Order **class**, **1012, 1015, 1020**
- OrderControl **class**, **1011, 1015, 1020**

OrderData **class**, 1016–1018
 OrderLine **class**, 1014
 OrderLineData **class**, 1019
overflow checking, 132, 134–135
 OverflowException **class**, 279

P

PadLeft **method** (String **class**), 218
 PadRight **method** (String **class**), 218
 Page **class**, 879
 PageCoordinatesToLineIndex **method**, 674
 Page_Load **method**
 PCSWebApp3 application, 895, 896, 897, 901
 Web service application, 923, 927
 WebForm1 **class**, 878, 879
 PaintEventArgs **class**, 629, 631
 Panel **control**, 591–592, 885
 param **XML tag**, 68
 ParameterTest.cs **file**, 89–90
 paramref **XML tag**, 68
 ParentForm **class**, 607
 Parse **method**
 Enum **class**, 57
 float **class**, 1187
 IPAddress **class**, 1080
 ParseCombiningCharacters **method** (StringInfo **class**), 511
Pascal casing, 76
 Path **class**, 1026, 1029–1030
 PathGradientBrush **class**, 649
 Pause **method** (ServiceController **class**), 1121
 PayAsYouGoCustomer **class**, 1166–1167
 PCSCustomWebControls.dll **file**, 940, 942
 PCSCustomWebControlsTestApp **namespace**, 943
 PCSUserC1 **class**, 935, 937
 PCSUserC1.ascx.cs **file**, 935
 PCSWebApp3.mdb **file**, 893
 pd_PrintPage **method**, 678
 Pens **class**, 627, 628, 649
 people.txt **files**, 297
percent sign (%) arithmetic operator, 132
percent sign, equals sign (%) assignment operator, 132
percent sign, greater than sign (%) ASP.NET code block suffix, 876
 perfmon.exe **utility**, 1133–1134
 PerformanceCounter **class**, 1130, 1133
 PerformanceCounterCategory **class**, 1130, 1131
 PerformanceCounterInstaller **class**, 1130

period (.)
 dot notation, 40
 member access operator, 132
 method calling operator, 1143
 namespace separator, 58–59
 regular expression special character, 232
permission. See also security
 assembly, 346, 388, 397–398, 401–403, 408–409
 AssemblyInfo.cs file configuration, 408
 asserting, 412–414
 CLR evaluation, 399, 407
 code access, 392, 399–403, 413, 414, 424–425
 database access, 399, 400
 demanding, 406–407
 denying, 407, 411–412
 directory service, 399
 DNS, 399
 drive access, 399, 406, 408
 environment variable access, 399
 event log access, 399
 Everything permission, 401, 410
 execution, 400
 file dialog, 399
 file I/O, 399, 406, 413
 full, 400, 419
 identity, 400, 402, 403
 implicit, 410–411
 Internet, 401
 intranet, 401, 407, 421–422
 isolated storage, 399
 merging permissions, 402
 message queue, 399
 minimum, 407
 Nothing permission, 400
 optional, 407
 policy, 423
 printing access, 399
 publisher identity, 400
 reflection access, 399
 registry access, 399
 requesting, 407–410
 security permission, 399, 413
 service controller, 400
 set, 425–427
 site identity, 400
 socket permission, 400
 SQL client, 400
 strong name identity, 400
 try catch block, inserting in, 400, 422
 union, 402

permission (continued)

permission (continued)

- URL identity, 400
- user interface access, 400, 408
- Web access, 400
- Web application, 560
- zone identity, 400

permission **XML tag**, 68

Permissions **namespace**, 406

PermissionSet **class**, 426–427

PhoneCustomer **class**, 85

piccyBounds **array**, 653

PictureBox **control**, 592

PlaceHolder **control**, 884

platform invoke mechanism, 20, 358

plus sign (+)

- arithmetic operator, 132
- regular expression special character, 232, 234–235
- string concatenation operator, 132

plus sign, equals sign (=) assignment operator, 132

plus signs (++) increment operator, 132

Point **struct**, 632–633, 638

pointer. See memory, pointer

PointerPlayaround.cs **file**, 209

PointerPlayaround2.cs **file**, 211

PointF **struct**, 632–633

PointToClient **method** (Control **class**), 580

PointToScreen **method** (Control **class**), 580

polymorphism, 1161–1164

PopupMenu **method**, 770, 772

POST **HTTP method**, 915

pre-emptive multitasking, 441

preprocessor directive, 70–73, 1256, 1304–1305

Principal **namespace**, 416

Print **method** (PrintDocument **class**), 676, 677

PrintDialog **class**, 675

PrintDocument **class**, 675, 676, 677

printf

- function, 357
- method, 358

Printing **namespace**, 625, 679

PrintPreviewDialog **class**, 675, 676, 677, 680

procedure, stored, 698–701, 722–723, 733, 760

ProcessAndDisplayNumber **method**
(MathOperations **class**), 173–174, 178–179

ProcessNextPerson **method**
(ColdCallFileReader **class**), 294

ProCSharp **namespace**, 59

Product **class**, 831–832, 833

ProductDataTable **class**, 717–718, 768, 769–770

ProductRow **class**, 717, 720, 769–770

Products

- class, 827–828, 829–830
- method, 718

products.xsd **file**, 715–716

Product.xsd **file**, 716–717

Professional ASP.NET 1.1 (Wiley), 880

Professional SQL Server 2000 XML (Wrox Press), 697

Professional XML (Wrox Press), 781

ProgressBar **control**, 592

Project ⇨ **Add Reference (Visual Basic .NET)**, 360

ProjectInstaller **class**, 567, 1108, 1109, 1127

Properties editor window (Windows Service),
1102–1103

Properties **property** (IMessage **interface**), 475

PropertiesToLoad **class**, 867

property

- accessor, 92–93, 1151
- C++, 1276–1277
- channel property, setting, 469
- context property, 461–462
- data encapsulation, 1151–1152
- defining, 93
- deployment project property display, 553–555
- function, relation to, 86
- inlining, 94
- Java, 1244–1245
- method versus, 80
- read-only, 93–94
- static, 96
- struct, 101
- virtual, 112
- Visual Studio .NET, editing using, 322–323
- write-only, 93–94

PropertyGrid **class**, 323

PropertyManager **class**, 753, 755–757

protocol, pluggable, 1077

publisher policy configuration file, 377, 383–385

Q

QueryInterface **method** (IUnknown **interface**), 968

question mark, colon (:): ternary operator, 134

question mark (?) regular expression special character, 232

quotation marks, double (“ ”) string literal delimiters, 46

QuoteServer **class**, 1097–1099, 1133

QuoteService **class**, 1103–1104, 1105

R

- race condition**, 450, 452–453
- RadioButton **control**, 592, 885
- RadioButtonList **control**, 885
- RainbowLabel **class**, 945–946, 947
- RainbowLabel2 **class**, 949–951
- RangeValidator **control**, 887
- RCW (runtime callable wrapper)**, 977–980
- RDN (relative distinguished name)**, 851
- Read **method**
 - BinaryReader class, 1042
 - Console class, 1250
 - FileStream class, 1044
 - XmlTextReader class, 788
- ReadByte **method** (FileStream class), 1044, 1084
- ReadChars **method** (XmlTextReader class), 788
- ReadElementString **method** (XmlTextReader class), 789–790
- ReadFileIntoStringCollection **method**, 1053
- ReadLine **method**
 - Console class, 33, 65, 283, 1250
 - StreamReader class, 1048
- ReadQuotes **method** (QuoteServer class), 1097, 1098
- ReadSettings **method**, 1062, 1063, 1064–1065
- ReadStartElement **method** (XmlTextReader class), 788
- ReadString **method** (XmlTextReader class), 788
- ReadXML **method** (DataSet class), 723, 820
- ReadXmlSchema **method**, 820–821, 824
- RealProxy **class**, 474, 475
- Receive **method** (UdpClient class), 1086–1087
- RecordCall **method**, 1153, 1155, 1157–1158, 1167–1168
- RecordPayment **method**, 1153
- Rectangle
 - class, 87
 - struct, 631, 635, 636
- RectangleF **struct**, 632, 635–636, 678
- RectangleToScreen **method** (Control class), 580
- Red-Green-Blue color**. See RGB color
- ReferralChasingOption **enumeration**, 849
- reference
 - compilation switch, 64
 - object, 35, 45, 105
- Reference.cs **file**, 917
- ReferenceEquals **method** (Object class), 104, 142–143, 144
- Reference.map **file**, 917
- referent, 1147
- reflection
 - array, returning, 268, 270, 274
 - assembly
 - metadata for, accessing, 271
 - referencing, 267
 - attribute information, returning, 272, 275
 - data type, 266–271, 274
 - described, 19, 257
 - permission, 399
 - remoting, role in, 458
 - Type class role in, 266–271
 - typeof operator, working with, 136
- Reflection **namespace**, 271, 351–352, 537
- RefreshCache **method** (DirectoryEntry class), 857
- RefreshQuotes **method** (QuoteServer class), 1099
- RefreshServiceList **method** (ServiceController class), 1117
- regasm **utility**, 993
- regedit **utility**, 1055
- Regex
 - class, 231, 236
 - method, 230
- RegexOptions **enumeration**, 231
- Region
 - class, 636
 - preprocessor directive, 72, 1305
- RegionDescription **field**, 700
- RegionInfo **object**, 520
- RegionInsert **procedure**, 700, 725, 730
- RegionSelect **procedure**, 723
- Register **method** (ILease interface), 482
- RegisterActivatedServiceType **method**, 482
- RegisterChannel **method** (ChannelServices class), 471
- RegisterServiceCtrlHandler **method**, 1104
- RegisterServiceCtrlHandlerEx **method**, 1135
- RegisterWellKnownClientType **method** (RemotingConfiguration class), 473
- RegisterWellKnownServiceType **method** (RemotingConfiguration class), 464, 471, 472
- registry**
 - access permission, 399
 - Add/Remove Programs dialog box registry requirement, 1055
 - COM component registration, 969, 974, 992–993
 - deployment, accessing during, 551, 562–563, 568–569
 - editing, 1055
 - hierarchy, 1055

registry (continued)

- hive, 1055–1057, 1064
- key, 1056, 1058–1059, 1064
- searching, 569
- window location, storing in, 1059–1065
- Windows Service registry entry, 1095, 1103, 1107–1108
- Registry **class**, **1055, 1057**
- RegistryKey **class**, **1055, 1057–1059, 1064, 1065**
- regsvcs **utility**, **1005**
- regular expression**. **See expression, regular**
- RegularExpressions **namespace**, **229, 231**
- RegularExpressionValidator **control**, **887**
- RejectChanges **method**, **825**
- relative distinguished name**. **See RDN**
- Release **method** (IUnknown **interface**), **967**
- ReleaseComObject **method** (Marshal **class**), **979**
- remarks **XML tag**, **68**
- RemoteHello **assembly**, **463**
- RemoteObject **class**, **499, 500, 501**
- remoting**
 - activator, 458, 465
 - ASP.NET environment, hosting in, 494–495
 - assembly configuration, 377, 457, 463–464
 - asynchronous, 498–499
 - attribute, directional, 480–481
 - call context, 505–507
 - channel
 - class representation, 456, 467
 - client channel loading, delaying, 490–491
 - contexts, in communication between, 462
 - creating, 467
 - described, 466
 - formatter, 458, 460, 470–471
 - HTTP, 466–467, 468
 - information about, returning, 467, 468, 471
 - instantiating, 467
 - machine.config file, predefined in, 486–487
 - pluggability, 470
 - priority, 467, 469
 - property, setting, 469
 - proxy messaging to, 465, 475
 - registering, 464, 471–472
 - serialization, 470, 476, 480
 - sink association with, 458
 - TCP, 456–457, 464–465, 466–467, 468, 469
 - CLR Object Remoting, 457
 - configuration files
 - client configuration, 487–489, 490, 503–504
 - ClientActivated_Client.config, 488–489
 - clientactivated.config, 485
 - ClientActivated_Server.config, 488
 - formatter configuration, 491–492
 - leasing configuration, 491
 - machine.config, 486–487
 - server configuration, 487, 488, 489–490, 502–503
 - Wellknown_Client.config, 487–488
 - wellknown.config, 485
 - WellknownHttp_Client.config, 488
 - Wellknown_Server.config, 487
 - context, 460–462
 - described, 456
 - encoding, 457
 - event handling, 499–505
 - exception handling, 481
 - formatter, 458, 460, 470–471, 491–492
 - garbage collection, 481
 - interceptor, 461
 - interface, separating client and server code using, 496
 - leasing, 481–484, 491
 - lifetime management, 481–484, 491
 - listening, 460, 466–467, 468, 470
 - message, 458
 - object, client-activated
 - client configuration, 488–489
 - creating, 473–474
 - described, 464
 - leasing mechanism, 482–483
 - lifetime management, 481
 - proxy, returning, 472
 - remoting configuration file, setup in, 485, 486, 488–490
 - server configuration, 472, 488
 - object, remote, 458, 462–463, 464, 472–480
 - pluggability, 457, 470, 475
 - proxy, 457, 458–459, 465, 474–475, 478
 - reflection mechanism role in, 458
 - security, 480
 - SingleCall mode, 464, 465, 473
 - sink
 - channel association with, 458
 - creating, 468, 470, 475, 476
 - described, 458
 - envoy sink, 459, 476
 - event sink, 503
 - object sink, 460, 476
 - provider, 469, 470
 - server context sink, 460, 476
 - SOAP and HTTP, using, 456–457, 469, 470, 495–498
 - URL, specifying, 472–473
 - WSDL document, generating, 497

- Remoting
 - assembly, 464, 486
 - namespace, 455
 - RemotingConfiguration **class**
 - client reading of, 503
 - Configure method, 489, 490
 - introduced, 458
 - RegisterWellKnownClientType method, 473
 - RegisterWellKnownServiceType method, 464, 471, 472
 - RemotingException **class**, 481
 - RemotingServices **class**, 473, 474, 476, 478
 - Remove **method**
 - ResultPropertyValueCollection class, 858
 - StringBuilder class, 222
 - RemoveAt **method**
 - ResultPropertyValueCollection class, 858
 - Vectors array list, 241
 - Render **method**, 945, 946, 951, 954, 957
 - RenderControl **method**, 951, 957
 - Renew **method** (ILease **interface**), 481
 - Repeater **control**, 886, 903
 - Replace **method**
 - String class, 218, 219–220, 221
 - StringBuilder class, 222
 - RequiredFieldValidator **control**, 887
 - Reset **method** (IEnumerator **interface**), 242, 1170, 1171
 - ResGen **utility**, 523, 537
 - resource file**
 - accessing, 525–526
 - assembly, adding to, 524–525
 - binary, 536–537
 - creating, 522–524, 531, 532
 - language translation using, 528–533, 536–538
 - reader, creating custom, 540–544
 - XML, 537–538
 - resource reader**, 540–544
 - ResourceGobbler **class**, 195
 - ResourceManager **class**
 - GetObject method, 526, 530
 - GetString method, 526
 - InternalGetResourceSet method, 543
 - introduced, 525
 - Resources namespace declaration, 537
 - ResourceSet returning using, 527
 - Resources **namespace**, 527, 537
 - ResourceSet **object**, 527, 542
 - ResourceWriter **class**, 523
 - ResultPropertyCollection **class**, 849, 854, 863, 867
 - ResultPropertyValueCollection **class**, 849, 858
 - Resume **method**
 - depthChangeThread class, 443
 - QuoteServer class, 1099
 - ResumeLayout **method** (Form1 **class**), 579
 - .resX **files**, 522
 - ResXResourceReader **class**, 527
 - ResXResourceSet **class**, 527
 - ResXResourceWriter **class**, 527
 - retrieveButton_Click **method**, 737
 - return **statement**, 33, 55
 - returns **XML tag**, 68
 - RevertAssert **method**, 413
 - RevertDeny **method**, 413
 - RGB (Red-Green-Blue) color**, 645–646
 - RhsIsGreater **class**, 176
 - RichTextBox **control**, 593, 1047
 - Rollback **method** (Installer **class**), 1108
 - roomList **control**, 894–897
 - Round **method** (PointF **struct**), 633
 - Rows **class**, 705, 724, 768
 - Rule **enumeration**, 715
 - Run **method**
 - Application class, 574, 1198
 - ServiceBase class, 1104, 1105
 - TestHarness class, 255
 - runtime callable wrapper**. *See* RCW
- ## S
- SampleClientApp **project**, 547, 548
 - SampleClientApp.exe **file**, 556
 - SampleWebApp **project**, 547, 548–549
 - SampleWebForm.aspx **file**, 549
 - SampleWebForm.aspx.resx **file**, 549
 - satellite assembly**, 533, 539, 540
 - Save **method** (XmlDocument **class**), 801
 - SaveFile **method**, 1052
 - SaverAccount **class**, 127
 - SaveSettings **method**, 1063
 - SAX (Simple API for XML)**, 781, 787
 - sc.exe **utility**, 1115–1116
 - schema**
 - Active Directory schema, 839, 841, 843–845, 856, 864–866
 - database schema
 - class, mapping to, 716
 - creating, 709–711, 764–770

schema, database schema (continued)

- definition file, 715–721
- generating, 709–710, 716–721
- introduced, 704–705
- Visual Studio .NET environment, 762–763, 764–770
- XDR schema, 791–792, 794, 821, 826
- XML schema
 - database table, creating from, 821, 824
 - generating from XDR schema, 826
 - ignoring, 813
 - reading, 820–821, 824
 - validating against, 787, 791–793, 794
 - writing, 813–814, 823–824
 - W3C standard, 782
- `SchemaNameCollection` **class**, 856
- SCM (Service Control Manager)**, 1093–1095, 1098, 1104–1105, 1106
- `ScreenToRectangle` **method** (`Control` **class**), 580
- scripting language**, 7
- `ScrollableControl` **class**, 582
- `ScrollingDataBinding` **method**, 755–756
- SDI (Single Document Interface)**, 599
- `SearchAll` **method**, 867
- `SearchResult` **class**, 849, 863
- `SearchResultCollection` **object**, 849, 863, 867
- `SearchScope` **enumeration**, 849, 861
- Secure Sockets Layer**. *See* **SSL**
- security**
 - activating/deactivating, 423
 - Active Directory, 839, 842
 - assembly
 - configuration file security setup, 377
 - permission, 346, 388, 397–398, 401–403, 408–409
 - code
 - basing security in, 14
 - distribution, of, 427–435, 557
 - evidence, 392
 - group, 392, 393–398, 423–425
 - mobile code, 391
 - permission, 392, 399–403, 413, 414, 424–425
 - trust-based, 392
 - unmanaged, 419
 - zone, 393
 - configuration file, 419–423
 - declarative, 414–415, 418–419
 - Download Assembly Cache, 391, 401
 - Enterprise Services, 1002, 1007
 - evidence, 392
 - exception, 407, 408, 415, 421
 - fine-grained, 839
 - intranet, 401, 407, 421–422, 435
 - .NET environment, 14, 405–406
 - object, serialized, 480
 - performance versus, 405
 - policy
 - code, considerations involving unmanaged, 419
 - configuration file, managing using, 419–423
 - enterprise-level, 403–405, 419
 - machine-level, 403–405, 419
 - permission management, 423
 - resetting, 423
 - user-level, 403–405, 419
 - process security, 15
 - remoting, 480
 - role-based
 - ASP.NET cookie-authenticated user, 415
 - declarative, 418–419
 - described, 14
 - Passport account, 415
 - principal, 415–417
 - Windows account, 415, 416–417
 - SSL, 436
 - turning on and off, 423
 - zone, 393, 400, 435–436
- `SecurityAction` **enumeration**, 408–409
- `SecurityException` **exception**, 407, 408
- `secutil.exe` **utility**, 432
- seealso* **XML tag**, 68
- `Select` **methods** (`XPathNavigator` **class**), 804, 805
- SELECT SQL statement**, 696–697, 702, 722
- `SelectCommand` **class**, 724
- `SelectDescendants` **method**, 805
- `SelectedIndexChanged` **method**, 799
- `SelectNodes` **method** (`XmlDocument` **class**), 799
- `SelectSingleNode` **method** (`XmlDocument` **class**), 799, 814
- semicolon (;)**
 - database connection string separator, 689
 - statement suffix, 31–32
- `Send` **method**
 - `Socket` class, 1098
 - `UdpClient` class, 1086–1087
- serialization**
 - remoting channel, 470, 476, 480
- XML
 - array, 826, 832, 834
 - deserializing, 826, 829
 - event handling, 828, 830, 832, 833, 834
 - field, 826
 - namespace, 825

- object graph, 826
- source code access, without, 833–836
- stream-based object, 827, 828
- Serialization **namespace**, 825, 1025
- Serialize **method**
 - IRemotingFormatter interface, 470
 - XmlSerializer class, 825, 827, 835
- serialprod1.xml **file**, 830–831
- Server **class**, 499
- Service Control Manager**. *See* **SCM**
- ServiceBase **class**, 1095, 1104–1106, 1126
- ServiceCommandCallback **method**, 1105
- ServiceController **class**, 1095, 1116–1122
- ServiceControllerStatus **enumeration**, 1118
- ServiceComponent **class**, 1003, 1004, 1016–1017
- ServiceInstaller **class**, 1095, 1108, 1110–1112, 1126
- ServiceInstallerDialog **class**, 1112
- ServiceMainCallback **method** (ServiceBase **class**), 1104, 1106
- Service1 **class**, 914
- Service1.asmx **file**, 913–914
- Service1.asmx.cs **file**, 913, 920, 925
- Service1.disco **file**, 917
- ServiceProcess **namespace**, 1095, 1104, 1108, 1135
- ServiceProcessInstaller **class**, 1095, 1108, 1110–1112
- Services administration tool**, 1092
- set **accessor**, 93–94, 1151
- SetAbort **method**
 - ContextUtil class, 1011
 - OrderLineData class, 1019
- SetComplete **method**
 - ContextUtil class, 1011
 - OrderLineData class, 1019
- SetData **method** (CallContext **class**), 506, 507
- SetDataBinding **method** (DataGrid **class**), 737, 738, 777
- SetDateAndNumber **method** (BookOfTheDayForm **class**), 529
- SetError **method** (Console **class**), 587, 1250
- SetHelpKeyword **method**, 588
- SetIcon **method**, 588
- SetIn **method** (Console **class**), 1250
- SetLength **method**, 1276
- SetLogonInformation **method**, 865
- SetNamingContext **method**, 865, 866
- SetOrderId **method** (Order **class**), 1012
- SetOut **method** (Console **class**), 1250
- SetPassword **method** (IADsUser **interface**), 859
- SetPrincipalPolicy **method**, 417
- setreg.exe **utility**, 432
- SetServiceStatus **method** (ServiceController **class**), 1118, 1119
- SetSocketOption **method** (Socket **class**), 1088
- SetUserProperties **method**, 865
- SetValue **method** (RegistryKey **class**), 1058, 1059, 1064, 1065
- SharedDemo **class**, 371–372, 379
- SharedDemo.dll **file**, 373, 375, 384
- Sharing **namespace**, 371, 373
- shfusion **utility**, 367–368
- Show **method**
 - BookOfTheDayForm class, 528
 - Form1 class, 627
 - MainClass class, 270
 - MessageBox class, 1261
 - Windows Form, 600
- ShowActivatedServiceTypes **method**, 489
- ShowChannelProperties **method**, 468
- ShowDialog **method**
 - Form class, 600, 603–604
 - PrintPreviewDialog class, 676
- ShowMessage **method**, 419
- ShowPanels **method** (StatusBar **control**), 594
- ShowRegionInformation **method**, 519, 520
- ShowSamples **method**, 519
- ShowWellKnownServiceTypes **method**, 489
- side-by-side feature**, 340, 341
- signcode.exe **utility**, 430
- Simple API for XML**. *See* **SAX**
- Simple Object Access Protocol**. *See* **SOAP**
- Simple TCP/IP Services**, 1097
- SimpleDelegate **namespace**, 172–174, 177
- Single Document Interface**. *See* **SDI**
- SingleCall **mode**, 464, 465, 473
- single-threaded apartment**. *See* **STA**
- Size **struct**, 632, 634–635, 638
- SizeF **struct**, 632, 634–635
- slash (/) **arithmetic operator**, 132
- slash, asterisk (/*) **comment prefix**, 31
- slash, equals sign (/=) **assignment operator**, 132
- slashes (//) **comment prefix**, 31
- slashes (///) **comment prefix**, 68
- Sleep **method** (Thread **class**), 501
- .sln **files**, 312
- sn (**strong name**) **utility**, 372, 1003
- soap **namespace**, 910

SOAP (Simple Object Access Protocol)

SOAP (Simple Object Access Protocol). *See also* **Web service**

- body, 910
- Enterprise Services application access, 1007
- header, 910, 924–929
- HTTP, use of, 909, 910–911
- length parameter, 910, 911
- remoting using SOAP and HTTP, 456–457, 469, 470, 495–498
- request handling, 910–911, 912, 915–916
- Web service, role in, 909
- W3C standard, 782
- SoapClientFormatterSink **class**, **471**
- SoapFormatter **class**, **470**
- SoapServerFormatterSinkProvider **class**, **469, 470**
- SoapSuds **utility**, **496–498**
- socket**
 - server implementation using Windows Service, 1096–1100
 - SSL, 436
 - streaming, 1088
- Socket **class**, **1083, 1088, 1098**
- Sockets **namespace**, **1067, 1083**
- SolidBrush **class**, **649**
- Sort **method** (BubbleSorter **class**), **174–175**
- SortDirection **enumeration**, **849, 863**
- sorting**
 - array, 168, 174–177, 521
 - localizing, 520–522
- SortOption **class**, **849**
- Spade **method**, **937**
- Specialized **namespace**, **1053**
- Split **method** (String **class**), **218**
- Splitter **control**, **594**
- sprintf **C function**, **1040**
- SQL (Structured Query Language)**
 - class overview, 687
 - Command class, passing SQL clause to, 693
 - command, executing, 694–698
 - database connection, 686, 689, 703–704
 - DELETE statement, 694, 699, 724
 - IDENTITY statement, 730, 731–732, 762
 - INSERT statement, 694, 724, 761, 1018
 - SELECT statement, 696–697, 702, 722
- Server
 - connection, 686, 689, 758
 - Enterprise manager, 731
 - key, generating using, 730–732

- scope, 732
- transaction isolation level, 692
- UPDATE statement, 694, 698–699, 761
- SqlClient **namespace**, **541, 686**
- SqlCommand **class**, **687, 699, 722–723, 725–726, 1018**
- SqlCommandBuilder **class**, **687**
- SqlConnection **class**, **541, 894**
- SqlDataAdapter **class**, **687, 707, 722, 723, 813**
- SqlDataReader **class**, **687**
- SqlDataSourceControl **control**, **334–335**
- SqlParameter **class**, **687**
- SqlTransaction **class**, **687**
- SqlTypes **namespace**, **686**
- Sqrt **method**, **1189**
- SquareRootForm **class**, **1193, 1194–1195, 1196–1197, 1198–1202**
- SSL (Secure Sockets Layer)**, **436**. *See also* **encryption**
- STA (single-threaded apartment)**, **574, 970**
- StackOverflowException **class**, **279**
- Start **method**
 - QuoteServer class, 1098, 1099
 - ServiceController class, 1121
 - TcpListener class, 1083
 - Thread class, 168, 442, 447
- StartListening **method** (IChannelReceiver **interface**), **468**
- StartServiceCtrlDispatcher **method**, **1104**
- Startup **class**, **607**
- StatusBar **control**, **594**
- StatusEvent **delegate**, **500–501**
- StatusEventArgs **class**, **500**
- StatusHandler **method**, **500, 503**
- stdio.h **file**, **357**
- Stop **method**
 - QuoteServer class, 1099
 - ServiceController class, 1121
- StopListening **method** (IChannelReceiver **interface**), **468**
- StoredProc.sql **file**, **722**
- StoredProcs.sql **file**, **698, 730**
- StrawPoll **class**, **954–955**
- StrawPollControlBuilder **class**, **955**
- strawPoll1_Voted **method**, **960**
- stream**
 - binary, 1041, 1042–1047
 - buffering, 1042
 - described, 1040
 - exception file, 293
 - memory stream, 813, 1040, 1042

- reading, 786–787, 813, 1040–1042, 1047–1054, 1068–1070
 - socket, streaming, 1088
 - writing, 786–787, 813, 1040–1044, 1047–1054, 1069
 - XML
 - memory stream, 813
 - pull model, 787, 807
 - push model, 787
 - reading, 786–787, 813
 - serializing stream-based object, 827, 828
 - writing, 786–787, 813
 - XSLT operation in, 808
 - StreamReader **class**, **293, 663, 813, 1040, 1048–1050**
 - StreamWriter **class**, **813, 1040, 1048, 1050–1051, 1052**
 - string**. *See text, string*
 - String **class**
 - Compare method, 218
 - CompareOrdinal method, 218
 - Format method, 218, 223–225, 228
 - IndexOf method, 218
 - IndexOfAny method, 218
 - LastIndexOf method, 218
 - LastIndexOfAny method, 218
 - PadLeft method, 218
 - PadRight method, 218
 - Replace method, 218, 219–220, 221
 - Split method, 218
 - Substring method, 218
 - ToLower method, 218
 - ToUpper method, 219, 674
 - Trim method, 219
 - StringBuilder **class**, **219–220, 221, 222, 225, 226**
 - StringCollection **class**, **1053**
 - StringDictionary **class**, **247**
 - StringInfo **class**, **511**
 - StreamReader **class**, **1040**
 - StringWriter **class**, **1040**
 - strong name utility**. *See sn utility*
 - struct**. *See also specific struct*
 - attribute, using with, 1235–1236
 - C++, 1256, 1279, 1287
 - class versus, 84, 101–102
 - constructor, 101, 103–104
 - declaring, 84
 - field, 102
 - function, defining for, 101
 - inheritance, 103, 110–111
 - initializing, 103
 - instantiating, 84
 - Java structure, 1234–1236
 - memory allocation, 103
 - Object class, derivation from, 103
 - pointer, 205–206, 209–212
 - property, 101
 - removing when out of scope, 103
 - value type, as, 102–103
 - ValueType class, derivation from, 110–111
 - Structured Query Language**. *See SQL*
 - Sub **method (IMath interface)**, **976**
 - submitButton_Click **method**, **899, 902, 923**
 - Subscriber **struct**, **58**
 - Substring **method (String class)**, **218**
 - suitList_SelectedIndexChanged **method**, **936**
 - summary **XML tag**, **68–69**
 - SupportsWhatsNewAttribute **class**, **263**
 - SuppressFinalize **method (GC class)**, **193**
 - Suspend **method**
 - depthChangeThread class, 443
 - QuoteServer class, 1099
 - SuspendLayout **method (Form1 class)**, **579**
 - switch **statement**, **49–51, 1239, 1264–1265**
 - SynchronizationAttribute **class**, **462**
 - SyncProcessMessage **method (IMessageSink interface)**, **476**
 - SyncProcessMessage **method (SoapClient FormatterSink class)**, **471**
 - System
 - class, 157–158
 - namespace, 32
 - SystemException **class**, **278**
- ## T
- TabControl **control**, **594–595**
 - Table **control**, **885**
 - TableCell **control**, **885**
 - TableCellCollection **class**, **885**
 - TableDirect **command**, **722**
 - TableRow **control**, **885**
 - TabPage **control**, **594–595**
 - TakeAWhile **method**, **499**
 - target **compilation switch**, **63**
 - TCP (transmission control protocol)**
 - channel, 456–457, 464–465, 466–467, 468, 469
 - class overview, 1083
 - described, 1083
 - UDP versus, 1086
 - TcpChannel **class**, **468**

TcpClient class

TcpClient **class, 1083, 1084, 1100–1101**

TcpClientChannel **class, 465, 467**

TcpListener **class, 1083, 1085, 1098, 1107**

TcpReceive **class, 1083–1086**

TcpSend **class, 1083–1086**

TcpServerChannel **class, 464, 469**

template

ASP.NET, 903–906

C# support, 1258

TestHarness **class, 254, 255**

text. See also expression, regular

case, converting, 218–219, 661–666, 674

culture, formatting for, 510–511

escape sequence, 44, 46, 229, 232, 1270–1271

file, reading, 661–662, 1040

font

family, 657, 659–661

GDI+ application, in, 657–661

listing available fonts, 659–661

sans serif, 658

serif, 658

size, 658

style, 657

True Type, 658

GDI+ application, in

case, converting, 661–666, 674

drawing, 655–657

font, 657–661

line index, 670–671

reading text file, 661–662

sorting, localizing, 520–522

string

appending to, 220, 221, 222

C++ string handling, 1270

comparing strings, 218

concatenating strings, 132, 218

data type, 45–46, 219

dictionary string key, 247

enumeration, converting to/from, 56–57

formatting, 218, 223–229

initializing, 45–46

inserting substring, 222

IP address, converting to/from, 1080

length, maximum, 45–46, 219–221, 222

length, returning, 220

literal, 46

location within, returning, 218

memory allocation, 45–46, 219–221

object string representation, returning, 105–107

padding, 218, 224

removing substring, 222

replacing substring, 218, 219–220, 221

returning substring, 218

splitting, 218

trimming, 219

whitespace, removing, 218, 219

XML text node, 788

Text

class, 221

namespace, 625

TextBox **control**

ASP.NET Web Form, 884

data binding, 750–751

Windows Form, 578–579, 593, 610, 620–621,
1045–1047

TextBoxBase **class, 593**

TextLineInformation **class, 663–664**

thick-client application, 24

this **statement, 95, 99, 1278–1279**

Thread **class**

CurrentCulture property, 512, 513, 516, 521, 534

CurrentThread property, 444

CurrentUICulture property, 512, 530, 534

Name property, 442

Sleep method, 501

Start method, 168, 442, 447

ThreadAbortException **class, 443**

threading

aborting thread, 443–444

array processing, 452–453

COM, 933, 969–971, 980

culture, 444, 512

deadlock, 450, 451–452

described, 439–440

exception handling, 443–444

instantiating thread object, 442, 444

logical thread, 506

memory, 449

MTA, 970–971

multitasking, pre-emptive, 441

mutex, 450

name of thread

assigning, 442

returning, 444

performance considerations, 447

priority, 448–449

race condition, 450, 452–453

referencing thread object, 444

sleeping, 501

STA, 574, 970

- starting thread, 168, 441–444, 445–447
- suspending/resuming thread, 443
- synchronization, 449–453, 462
- time slice, 441
- variable access, 449–450
- Visual Basic 6, 1196
- Web application, 1085–1086
- Windows account, 416
- Windows Form, 574, 575
- Windows Service, 1107
- worker thread, 442, 445
- Threading
 - class, 442
 - namespace, 416, 441, 1085
- ThreadPlayaround **class**, 445, 446
- ThreadPriority **enumeration**, 448
- ThreadStart **delegate**, 442
- throw **statement**, 281, 282
- tilde (~)**
 - destructor prefix, 194
 - logical operator, 132
- time**
 - current, returning, 98
 - file time data, changing, 1029
 - folder time data, displaying, 1031
 - threading time slice, 441
- TimeOfDay **enumeration**, 56–57
- tlbexp **utility**, 987
- tlbimp **utility**, 978
- tmain **method**, 994
- ToInt32 **method** (**Convert class**), 283, 285
- ToLongDateString **method** (**DateTime class**), 515, 516
- ToLongTimeString **method** (**DateTime class**), 515
- ToLower **method** (**String class**), 218
- ToolBar **class**, 597
- ToolBarAppearance **enumeration**, 597
- ToolBarButton Collection Editor, 597
- ToolBarButtonClickEventArgs **object**, 598
- Tools ⇨ **OLE/COM Object Viewer (Visual Studio .NET)**, 987
- ToShortDateString **method** (**DateTime class**), 515
- ToShortTimeString **method** (**DateTime class**), 515
- ToString **method**
 - AddressFamily class, 1082
 - Currency struct, 165
 - date formatting using, 515
 - Employee class, 1207–1208
 - IFormatProvider interface argument, 225, 513–514, 515
 - Manager class, 1212
 - Object class, 104, 105–107, 170, 225–226, 227–228
 - StringBuilder class, 222
- ToUpper **method** (**String class**), 219, 674
- Trace
 - class, 1129
 - preprocessor symbol, 330
- TransferTo **method**, 129
- Transform **method** (**XslTransform class**), 808, 810
- TranslateTransform **method** (**Graphics class**), 643
- transmission control protocol**. **See TCP**
- TreeNode **class**, 517, 518, 612–613
- TreeView **class**, 517, 518, 611–617
- triggerButton_Click **method**, 882
- Trim **method** (**String class**), 219
- Truncate **method** (**PointF struct**), 633
- try catch finally **block**
 - array out-of-bounds error, trapping using, 281–282
 - C++, 1297–1298
 - catch clause
 - described, 280
 - inserting multiple, 282–286
 - omitting, 281
 - parameter, passing to, 284
 - throwing exception from catch block, 289
 - database connection, closing using, 690
 - destructor, inserting, 194
 - finally clause
 - described, 280
 - omitting, 281
 - throwing exception from finally block, 289
 - IDisposable interface, inserting, 195
 - permission, inserting, 400, 422
 - syntax, 280–281
 - throw statement, inserting, 281, 282
 - try clause
 - described, 280
 - exiting try block, 284
 - nesting, 287–290
 - user input error, checking for using, 282
- TwoLongsOp **delegate**, 169
- Type **class**
 - BaseType property, 267
 - FullName property, 266
 - GetConstructor method, 268
 - GetConstructors method, 268
 - GetEvent method, 268
 - GetEvents method, 268
 - GetField method, 268

Type class (continued)

Type class (continued)

GetFields method, 268
GetInterface method, 268
GetInterfaces method, 268
GetMember method, 268
GetMembers method, 268, 270
GetMethod method, 267
GetMethods method, 267–268
GetProperty method, 268
GetProperty method, 268
instance of class derived from, returning, 105, 266
instantiating, 266
Invoke method, 268
IsAbstract property, 267
IsArray property, 267
IsClass property, 267
IsEnum property, 267
IsInterface property, 267
IsPointer property, 267
IsPrimitive property, 267
IsPublic property, 267
IsSealed property, 267
IsValueType property, 267
Name property, 266
Namespace property, 266
reflection, role in, 266–271
UnderlyingSystemType property, 267
TypeView **assembly**, 268–271

U

UDP **class**, 1086–1087
UDP (user datagram protocol), 1086–1087
UdpClient **class**, 1083, 1086
undef **preprocessor directive**, 70–71, 1305
underscore (_) field name prefix, 76
UnexpectedException **class**, 296
Unicode, 510–511, 1048
Uniform Resource Identifier. See URI
Uniform Resource Locator. See URL
Uninstall **method (Installer class)**, 1108
Unmarshal **method (RemotingServices class)**, 476
UnsafeNativeMethods **class**, 582
Update **method**
 data adapter routine, from, 724
 IOrderUpdate interface, 1016
update sequence number. See USN
UPDATE **SQL statement**, 694, 698–699, 761
UpdateCommand **class**, 724
UpdateRowSource **enumeration**, 699, 700

UploadData **method (WebClient class)**, 1070
UploadFile **method (WebClient class)**, 1070
UPN (user principal name), 853
Uri **class**, 1077–1078
URI (Uniform Resource Identifier), 235–236, 1068, 1077–1078
UriBuilder **class**, 1077–1078
URL (Uniform Resource Locator), 472–473, 1068
user control
 ASP.NET Web Form, 880
 composite, 934
 control custom compared, 939
 event handling, 936, 938
 files created with, 932–933
 graphic, associating with, 933
 postback, 936
 referencing, 934
 reusability, 932
 state, specifying default, 934
 tag name, 934
 Visual Studio .NET user control project, 932
user datagram protocol. See UDP
user principal name. See UPN
UserControl **class**, 933
using **directive**, 32, 59–60, 195–196, 1194, 1226
USN (update sequence number), 842
util.netmodule **module**, 344

V

ValidationSummary **control**, 887, 888
value **XML tag**, 68
ValueType **class**, 110–111
variable
 array, referencing, 57–58
 ASP.NET, declaration in, 876
 C++, 1254, 1266, 1272
 class-level, 37–38
 compilation error caused by variable violation, 34, 37
 constant, designating as, 38
 counter, 51
 declaring, 34
 environment variable permission, 399
 initializing, 34–35
 Java, 1229–1230
 local, 35–36
 method, passing to, 89–90
 naming convention, 74–75, 1230
 scope, 35–38, 188–189
 threading, variable access in, 449–450

- unsafe, marking local variable as, 199–200
- user-defined, 1141
- value
 - assigning, 34
 - comparing values for equality, 48
 - watching, 332
- Visual Basic 6, 1181–1182, 1185, 1197
- .vbq files, 312
- VBToCSharp_CSharpSource.pdf file, 1192
- vcvars32.bat file, 31
- Vector
 - class, 275
 - struct, 146–150, 152, 223, 227–229, 243–245
- VectorClass assembly, 262, 264–265, 274, 275
- VectorEnumerator class, 244–245, 275
- Vectors array list, 240–241
- Vectors3.cs file, 152–153
- Vectors2.cs file, 151
- VES (Virtual Execution System), 393
- View ⇨ Detail (Component Services Admin utility), 1009
- viewstate field, 875
- Visual Basic .NET
 - case sensitivity, 12, 77
 - class, creating, 359–362
 - compilation, 6
 - error handling, 278
 - Visual Basic 6, converting application from, 6
 - Visual Studio .NET, reading Visual Basic project in, 314–315
- Visual Basic 6
 - ActiveX control, 1193
 - array, 1218–1219
 - case sensitivity, 1184
 - class, 1178, 1187, 1192–1193, 1196–1198
 - commenting code, 1182–1183
 - compilation, 1179
 - constructor, 1199
 - data type, 1221–1222
 - for loop, 1220–1221
 - garbage collection, 1197
 - If statement, 1188–1189
 - inheritance, 1178, 1194–1195, 1210–1212
 - Main function, 1195–1196
 - member, static, 1208–1210
 - method versus subroutine, 1184–1185
 - namespace, 1192–1194
 - .NET base class library, 1179
 - Object Browser, 325
 - operator, 1222–1224
 - program entry point, 1195–1196
 - program execution, 1178, 1190–1192
 - syntax, 1182–1184
 - threading, 1196
 - variable, 1181–1182, 1185, 1197
 - Visual Basic .NET, converting application to, 6
- Visual C++ .NET, 6–7, 314
- Visual J# .NET, 7
- Visual Studio .NET
 - Add Project Output Group dialog box, 558
 - ASP.NET, developing in, 22, 875–877
 - AssemblyInfo.cs file generation, 350–351
 - C++ code, creating using, 356
 - Class View, 323–324
 - COM Object Viewer, 987
 - compilation, 31, 302, 307–308, 326–331
 - Console Application option, 308–310
 - Copy Project dialog box, 550
 - Custom Actions Editor, 566–568
 - database schema, 762–763, 764–770
 - debugging features, 302, 328–329, 330, 331–334
 - Design view, 302, 318–321
 - editor, folding, 315–318
 - event, viewing in, 322
 - Exceptions dialog box, 333–334
 - File System Editor, 562
 - File Types Editor, 564
 - FlowLayout mode, 877
 - GridLayout mode, 877
 - IntelliSense feature, 301–302, 318, 611
 - JUMP, 315
 - Launch Conditions Editor, 568–569
 - Linker, 356
 - Main method created by, 304, 308
 - MSDN documentation, 302
 - My Profile option, 306
 - .NET Framework installed with, 303
 - New Project dialog box, 306–307, 312–313
 - Object Browser, 324–325
 - perfmon profiling tool, 303
 - Performance Counter Builder, 1130–1131
 - pin button feature, 326
 - project
 - ASP.NET Web application project, 876
 - Cab Project, 551
 - compilation option, specifying when creating, 307–308
 - console project, 308–310
 - creating, 306–307, 312
 - custom control project, 940–943
 - database data, adding, 759–762
 - Debug configuration, 329

Visual Studio .NET, project (continued)

Visual Studio .NET, project (continued)

- deploying application via copying, 546, 550
- deployment project, 546, 551–552, 553, 556–559
- described, 311
- editing configuration, 329–331
- files created with, 310, 312
- installer project, 551
- J++ project, reading, 315
- Merge Module Project, 551
- Release configuration, 329
- Setup Wizard Project, 551
- solution, adding project to, 312–313
- solution versus, 311–312
- startup, specifying project to run at, 313–314
- type, selecting, 306–308
- user control project, 932
- Visual Basic project, reading, 314–315
- Visual C++ project, reading, 314
- Visual Studio 6 project, reading, 314–315
- Web service project, 913, 917
- Web Setup Project, 551
- Properties window, 322–323
- RCW, creating using, 978
- Registry Editor, 562–563
- Server Explorer, 325–326, 758, 1116
- Solution Explorer, 311, 524
- Solution Property Pages dialog box, 552
- Start Page, 305–306
- Toolbox, 319
- User Interface Editor, 564–566
- version
 - 2002/2003 versions compared, 303
 - upgrading to 2003, 304
- Visual Studio 6 project, reading, 314–315
- View Designer, 578
- View menu, 318
- Watch window, 332
- Web Form development using, 334, 875–877
- Web Matrix feature, 334–335
- WinCV utility, 335–336
- window, pinning, 326
- VoidOperation **delegate**, 169

W

WaitForStatus **method** (ServiceController **class**), 1122

warning **preprocessor directive**, 72, 1305

Web Form

- AdRotator control, 884
- Button control, 881–882, 885

- Calendar control, 885, 897–899
- CheckBox control, 884
- CheckBoxList control, 885
- CompareValidator control, 887
- CustomValidator control, 887
- data binding, 892, 894–897, 901–906
- DataGrid control, 886, 901–902
- DataList control, 886, 904–906
- DropDownList control, 884
- event handling, 878, 879, 882
- HTML server control, 880
- HyperLink control, 885
- Image control, 884
- ImageButton control, 885, 937–938
- introduced, 23
- Label control, 881, 884, 891
- layout, 877
- LinkButton control, 885
- ListBox control, 884
- Literal control, 884
- Panel control, 885
- Placeholder control, 884
- posting, 882–883, 890
- processing, server-side, 875, 878, 881
- RadioButton control, 885
- RadioButtonList control, 885
- RangeValidator control, 887
- RegularExpressionValidator control, 887
- Repeater control, 886, 903
- RequiredFieldValidator control, 887
- server control overview, 880–883
- sizing, 92
- state management, 882
- Table control, 885
- TableCell control, 885
- TableRow control, 885
- TextBox control, 884
- user control, 880
- validation control, 880, 886–892
- ValidationSummary control, 887, 888
- Visual Studio .NET, developing in, 334, 875–877
- Web server control, 883–886
- Xml control, 884

Web Matrix Project, 335

Web service. See also SOAP (Simple Object Access Protocol)

- authentication, 925–927
- consuming, 913, 916–918
- data type availability, 916
- database connection, 919, 920–921, 923

- DataSet, working with, 919–923
- event handling, 918, 927, 928
- exposing, 913–916
- proxy class, 917
- referencing, 917–918
- SOAP role in, 909
- type definition, 911–912
- Visual Studio .NET Web service project, 913, 917
- WSDL, 497, 911–912, 916
- Web Service Description Language. See WSDL**
- WebClient **class**, **1068–1070, 1073**
- Web.config **file**, **876**
- WebControl **class**, **883, 939**
- WebCustomControl1 **class**, **941, 944–945**
- WebForm1 **class**, **878, 879, 880, 882**
- WebForm1.aspx **file**
 - ASP.NET application, 881, 882–883, 888, 901, 904
 - control configuration, 934, 936, 941–942
 - introduced, 876
- WebForm1.aspx.cs **file**, **878–880, 918, 927, 934, 943**
- WebRequest **class**, **1070–1073, 1075–1077**
- WebResponse **class**, **1070–1073, 1075–1077**
- Welcome **method**, **1004–1005**
- Welcome.de.resx **file**, **537**
- WelcomeMessage **method** (BookOfTheDayForm **class**), **528, 529, 537**
- welcome.txt **file**, **536**
- wellknownhttp.config **file**, **485**
- WellKnownObjectMode **enumeration**, **471–472, 473**
- WhatsNewAttributes **assembly**, **262, 263–264**
- WhatsNewChecker **class**, **274**
- while **statement**, **53, 1264**
- Wiley (Professional ASP.NET 1.1), 880**
- window location, storing in registry, 1059–1065**
- Windows Form**
 - application based on, creating, 574–579
 - application support provided by, 24
 - border, 604–605
 - closing, 599, 600, 601
 - color, 605
 - component, adding/removing, 577–578
 - control
 - ActiveX control, 24, 982–985
 - attribute list, 610–611
 - background, loading in, 617
 - CheckBox, 585
 - COM client, accessing from, 997–998
 - ComboBox, 586
 - creating custom, 610
 - data binding, 582, 750–757
 - DateTimePicker, 587
 - image, 589, 590–591, 617
 - initializing, 615–617
 - Intellisense support, 611
 - Label, 589, 618
 - ListBox, 586
 - Panel, 591–592
 - PictureBox, 592
 - placing, 580
 - ProgressBar, 592
 - RadioButton, 592
 - RichTextBox, 593, 1047
 - scrolling behavior, 582, 591
 - sizing, 580, 589, 592
 - Splitter, 594
 - StatusBar, 594
 - TabControl, 594–595
 - TabPage, 594–595
 - TextBox, 578–579, 593, 610, 620–621, 1045–1047
 - TreeView-based, 611–617
 - user control, 618–622
 - destroying, 599, 601
 - displaying/hiding, 600, 601
 - error handling, 587–588
 - ErrorProvider component, 587–588
 - event handling, 180, 575, 580–582, 585, 599–600
 - HelpProvider component, 588–589
 - ImageList component, 585, 589, 590–591
 - inheritance, 109–110
 - instantiation, 599–604
 - layout, 579
 - loading, 599
 - location, specifying default, 600
 - Main method, 574, 605
 - MDI application, 599, 607–609
 - menu, 595–596, 608
 - message filtering, 575
 - namespace, 574
 - path, 575
 - SDI application, 599
 - sizing, 600
 - style, visual, 575, 605–606
 - system menu access, 604
 - taskbar display, 600, 605
 - threading, 574, 575
 - toolbar, 596–598
 - Windows Form, 574
- Windows handle. See HWND**
- Windows Installer, 551–552**

Windows registry. See **registry**

Windows Resource Localization Editor, 533

Windows Service

- client, 1096, 1098, 1100–1102, 1113
- command, sending to, 1121
- configuration program, 1093, 1095
- connection, 1098
- control program, 1095, 1116–1122
- creating, 1096, 1102
- dependent service, returning, 1112, 1118
- described, 24, 1091–1092
- Designer surface, 1102
- device driver, returning, 1117
- event log, 1103, 1104, 1123–1128
- handler function, 1094
- initializing, 1105
- installing, 1095, 1107–1113
- interactive, 1123
- listening, 1098, 1129–1130
- logging, 1103
- machine running, returning, 1118
- Main
 - function, 1094, 1105
 - method, 1104, 1105
- MMC Services snap-in, 1114
- monitoring, 1113–1114, 1130–1135
- name, 1103, 1112, 1118
- NT kernel requirement, 1092
- pausing/resuming, 1099, 1106, 1118, 1121
- port setup, 1100
- power event, 1105, 1135
- project, 1102–1107
- Properties editor window, 1102–1103
- registry entry, 1095, 1103, 1107–1108
- request handling, 1103
- returning all services, 1092, 1117
- SCM, 1093–1095, 1098, 1104–1105, 1106
- service program, 1093
- shutting down, 1106, 1118
- socket server implementation using, 1096–1100
- starting, 1099, 1104, 1106, 1112, 1121
- status
 - setting, 1118, 1119
 - waiting for, 1122
- stopping, 1099, 1106, 1121
- threading, 1107
- tracing, 1128–1129
- type, returning, 1118
- user account, 1111
- Visual Studio .NET Server Explorer, 325–326, 758, 1116

WindowsIdentity **class,** 416, 417

WinMain **method,** 1261, 1262

winres.exe **utility,** 533

WinServices **namespace,** 1103

Win32 **namespace,** 1025

WithoutSchema.xml **file,** 727

WithSchema.xml **file,** 727–728

WndProc **method** (Control **class**), 582

workerStart **method** (ThreadPlayaround **class**), 446

World Wide Web Consortium. See **W3C**

WorldYCoordinateToLineIndex **method,** 669, 670

Write **method**

- BinaryWriter class, 1041–1042

- Console class, 52, 65, 1250

- FileStream class, 1044

- StreamWriter class, 1051

WriteAttributeInfo **method,** 275

WriteAttributeString **method** (XmlTextWriter **class**), 795

WriteByte **method** (FileStream **class**), 1044

WriteCData **method** (XmlTextWriter **class**), 795

WriteChars **method** (XmlTextWriter **class**), 795

WriteComment **method** (XmlTextWriter **class**), 795

WriteContentTo **method** (XmlDocument **class**), 801

WriteElementString **method** (XmlTextWriter **class**), 795

WriteEndElement **method** (XmlTextWriter **class**), 795

WriteEntry **method** (EventLog **class**), 1126–1127

WriteLine **method**

- Console class, 65–66, 165, 223–224, 1148–1149, 1250–1251

- StreamWriter class, 1048

WriteMatches **method,** 233

WriteStartDocument **method** (XmlTextWriter **class**), 795

WriteStartElement **method** (XmlTextWriter **class**), 795

WriteTo **method** (XmlDocument **class**), 801

WriteXml **method** (DataSet **class**), 727, 813, 818

WriteXmlSchema **method,** 814

Wrox Press

- Beginning XML,* 781

- Data-Centric .NET Programming with C#,* 836

- Professional SQL Server 2000 XML,* 697

- Professional XML,* 781

WSDL (Web Service Description Language), 497, 911–912, 916

WSDL.exe file, 917

W3C (World Wide Web Consortium), 782

X

xcopy utility, 18, 376, 546, 548–549

XDR (XM-Data Reduced) schema, 791–792, 794, 821, 826

Xml control, 884

XML (Extensible Markup Language). See also specific tag

ADO.NET

converting ADO.NET data to XML, 812–820

converting XML to ADO.NET data, 820–822

attribute

iterating through attributes, 791

value, returning, 790–791

CData section, 796

class overview, 782–783, 795–797

comment, generating XML documentation from, 68–70

DataSet class, populating from XML file, 723

DOM, 795–797

DTD, 791–792

element

creating, 800

navigating, 803–804

reading data from, 788–790

selecting, 804

entity, 796, 797

event handling

serialization event, 828, 830, 832, 833, 834

validation event, 792, 793

MSXML parser, 783–786

.NET Framework, role in, 781

node

child, 795

content, returning, 785, 788

count, returning, 804

declaration node, 797

entity reference node, 797

inserting, 799–802, 804

iterating through nodes, 785, 788–789

listing nodes, 796

parent, 795

position, returning, 804

returning, 796

saving, 801

selecting, 799, 804, 805

text node, 788

reading

Command class, using, 696–698

DataSet class, using, 723, 820

DiffGram document, using, 822–825

element data, 788–790

schema, 820–821, 824

stream, 786–787, 813

XmlTextReader class, using, 782, 787–791, 793

resource file, XML, 537–538

SAX, 781, 787

schema

database table, creating from, 821, 824

generating from XDR schema, 826

ignoring, 813

reading, 820–821, 824

validating against, 787, 791–793, 794

writing, 813–814, 823–824

W3C standard, 782

SELECT SQL statement FOR XML clause, 696–697

serialization

array, 826, 832, 834

deserializing, 826, 829

event handling, 828, 830, 832, 833, 834

field, 826

namespace, 825

object graph, 826

source code access, without, 833–836

stream-based object, 827, 828

stream

memory stream, 813

pull model, 787, 807

push model, 787

reading, 786–787, 813

serializing stream-based object, 827, 828

writing, 786–787, 813

XSLT operation in, 808

validating

error handling, 792–794

event handling, 792, 793

schema, against, 787, 791–793, 794

XmlValidatingReader class, using, 791–794

whitespace, 796

writing

DataSet class, using, 727, 813, 818

DiffGram document, using, 822–825

indentation, 785, 795

schema, 813–814, 823–824

stream, to, 786–787, 813

XmlDocument class, using, 801

XML (Extensible Markup Language), writing (continued)

XML (Extensible Markup Language), writing (continued)

- XmlTextWriter class, using, 782, 787, 794–795, 801
- XmlWriter class, using, 782, 786–787, 810
- W3C standards, 782
- Xml namespace, 697, 782–783
- XML Path Language. See XPath**
- XmlArray class, 834
- XmlArrayAttribute class, 832
- XmlArrayItem constructor, 832
- XmlArrayItems class, 834
- XmlAttribute class, 796
- XmlAttributeOverrides class, 833, 834, 835
- XmlAttributes class, 833, 834, 835
- XmlCDATASection class, 796
- XmlCharacterData class, 796
- XmlComment class, 796
- XmlDataDocument class, 783, 807, 815–816
- XmlDeclaration class, 797, 802
- XmlDocument class, 783, 796, 797–802, 814–815
- XmlDocumentFragment class, 796
- XmlDocumentType class, 797
- XmlElement class, 797
- XmlElementAttribute class, 827, 834, 835
- XmlElement collection, 834
- XmlEntity class, 796
- XmlEntityReferenceNode class, 797
- XmlLinkedNode class, 796
- XmlNode class, 783, 795
- XmlNodeList class, 796
- XmlNotation class, 796
- XmlProcessingInstruction class, 797
- XmlReader class, 782, 786–787, 797, 799
- XmlResolver class, 783
- XmlRootAttribute class, 827–828
- XmlSerializer class, 825, 827, 829, 834–835
- XmlSignificantWhitespace class, 796
- XmlTextReader class, 782, 787–791, 793
- XmlTextWriter class, 782, 787, 794–795, 801
- XmlUrlResolver class, 783
- XmlValidatingReader class, 791–794
- XmlWhitespace class, 796
- XmlWriteMode enumeration, 813
- XmlWriter class, 782, 786–787, 810
- XPath (XML Path Language), 785, 803–807**
- XPathDocument class, 803, 807, 808, 810
- XPathException class, 803
- XPathExpression class, 803
- XPathNavigator class, 803–804, 805–806
- XPathNodeIterator class, 803, 804, 805
- XSD files, 715
- XSD.EXE utility, 716–721, 826
- Xsl namespace, 807
- XSLT (XSL Transforms), 782, 802, 807–812**
- XsltArgumentList object, 808, 809–811
- XsltTransform class, 808, 810, 811
- .xsl files, 765