

Index

Symbols

#Const statement, 259
#if compiler directive, 331
& concatenation operator, 308–309
***** (asterisk), 198–199
_ (underscore), 258–259
+ concatenation operator, 308–309
< operator, 309
<%= .. .%>, 482
<< operator, 308
<= operator, 309
<> operator, 309
= operator, 309, 314
> operator, 309
>= operator, 309
**** operator, 308

A

About dialog box, Windows Forms, 153–154
abstract classes, 567, 705
abstraction, and inheritance, 553–555
AcceptButton property, dialog boxes, 170
accessibility clause
 class declarations, 565–566
 event declarations, 591–592
 functions, 940
 subroutines, 333, 939
 variable declarations, 261–262, 271–272
accessibility keywords, constant declarations, 300–301
Add Connection dialog box, database connections, 404–405
Add Existing Item command, 22
Add method, strongly typed collections, 630
Add New Data Source command, 28
Add New Item command, 22
Add Query command, 28
Add Reference command, 23
Add Service Reference command, 24
Add submenu command, 18
Add Tab command, 38
Add Windows Form command, 164
AddHandler method, arrays, 277
AddHandler statement
 catching events, 593–595
 creating controls at runtime, 125–126
 declaring custom events, 596–599
 setting event handlers at runtime, 141–142
Add-In Manager command, 30
AddMessageFilter method, **Application** object, 831, 833
Administer Test Controllers command, 34
 administrator privileges. *See* **UAC (User Account Control) security**
ADO.NET, LINQ to, 487
Advanced Compiler Settings dialog box, 245–246
Advanced submenu, 20
Aero Glass style, 4
 aliases, namespace, 609–610, 655
Align command, **Format** menu, 28
Alignment property, 700, 729
Anchor property, controls, 129–130, 135
And operators, 311, 313
AndAlso operator, 311–312
 animations, XAML, 206–208
 anonymous types, 298–299, 460
 anti-aliasing, 670–672, 682–684
AppendText method, **File** class, 846–848
Application class
 configuration and resources, 830–835
 defined, 830
 events, 833–835, 1111
 methods, 831–833, 1110
 properties, 830–831, 1109–1110
application icons, Windows Forms, 157
application resources, 821–823
ApplicationExit events, 833–834
arithmetic operators, 307–308
ArrayList class, 627–629
arrays, 620–629
 Array class, 620–621
 Array.BinarySearch method, 626
 ArrayList class, 627–629
 Array.Reverse method, 624

arrays (continued)

- Array.Sort method, 625–626
- bounds_list clause specifying bounds for, 279–280
- declared ByVal and ByRef, 335–337
- defined, 619
- dimensions, 621–622
- IndexOf and LastIndexOf methods, 624
- initializing, 282–284
- lower bounds, 622
- not using WithEvents to declare, 277
- overview of, 620–621
- parameter, 339–341
- resizing, 623
- speed, 623–624
- using SortedList class vs., 639
- of variables vs. Array objects, 620–621

As clause, variable declarations, 280–281, 928

assemblies, strongly named, 514–516

Assembly Information dialog box, 240–241

Assert method, Debug, 248–249, 379

assignment operator, 314–316, 935

Attach to Process command, 29, 94

attribute_list

- class declaration with, 564
- event declaration with, 591
- subroutines, 328–332

attributes

- assigning to components, 510
- assigning to custom controls, 499–501
- overview of, 328–332
- in variable declarations, 270–271
- XAML line, 209–210

AttributeUsage attribute, 329

Audio object, My.Computer, 1117

Auto Hide All command, 35

Auto Hide command, 34

autoexec.nt file, 808

automatic imports, namespace, 607–609

Automatically Open Smart Tags setting, WFD, 50

AutoRedraw, 751–752

Autos window, Debug menu, 92–93

AutoToolboxPopulate setting, WFD, 51

Average function, LINQ, 470–471

axes, LINQ query XML, 485–487

B

BackColor property, Windows Forms, 158–159

Background property, XAML, 197–201

BackgroundColor property, HatchBrush class, 709

BackgroundImage property, Form object, 750

BackgroundWorker component, 104, 955

Begin a Group command, 47

BeginPrint event, PrintDocument, 762, 768–769

BeginTransaction method, 417–419

Bézier curves, 212–213, 663, 675–676

binary-file access, 856

BinaryReader class, 838, 841–843, 1133–1135

BinaryWriter class, 838, 841–843, 1133–1135

binding, data

- complex, 451–455
- simple, 446–447

BindingNavigator control, 104, 411, 955

BindingSource control

- complex data binding, 452–453
- data overview, 413–414
- overview of, 104
- simple data binding, 447
- summary of, 955

bit shift operators, 931–932

Bitmap class

- AutoRedraw using, 750–752
- loading bitmaps, 748
- methods, 1092
- overview of, 747–748
- pixel-by-pixel operations, 753–757
- saving bitmaps, 748–750

bitwise operator, 312–313, 934

BlinkStyle property, ErrorProvider, 966

block scope, 288–289

bookmarks, adding/removing, 73

Bookmarks submenu, 20

Border control, WPF, 177

Bottom property, Windows Forms controls, 133

bounds_list clause, variable declarations, 279–280

boxing, 580, 647

Break All command, Debug menu, 88

Breakpoint Hit Count dialog box, 96–97

breakpoints, 73, 295

Breakpoints window, Debug menu, 91–92, 95–97

Browsable attribute, 329, 501

Brush class, 705–719

- derived classes, 1089
- filling areas with, 662–663
- garbage collection issues, 723–725
- HatchBrush class, 709
- LinearGradientBrush class, 709–713
- overview of, 705–706
- PathGradientBrush class, 714–719
- SolidBrush class, 706
- TextureBrush class, 706–708

BufferedStream class, 841

bugs, 377–380

Build menu, 26–27, 494

BulletDecorator control, WPF, 177

Button class, 224–225

Button control

creating dialog boxes, 170–171

defined, 105

initiating action, 115

modifying commands, 45–47

summary of, 956

UAC privilege elevation, 540–541

wizards using, 172

WPF, 182

ByRef keyword

array declarations, 335–337

parameter declarations, 292–294

passing parameters, 578–579

subroutine parameter declarations, 335

ByVal keyword

array declarations, 335–337

parameter declarations, 292–294

passing parameters, 578–579

subroutine parameter declarations, 334–335

C

Call Stack window, Debug menu, 92–93

called program, UAC elevated privileges, 543

calling program, UAC elevated privileges, 542–543

Cancel buttons, 151, 153–154, 170–172

Canvas control, WPF, 177

case sensitivity

CollectionsUtil class, 639

compilation constant values, 246

naming conventions, 303–304

Path object in XAML, 210

Catch statements, error handling, 383–384

Category attribute, 329, 501

Center in Form command, 29

CenterColor property, PathGradientBrush class, 714, 718

CenterPictureInMargins subroutine, 773–775

certificate authorities, 516–517

Change Button Image command, 46

characters, data type, 264–265

characters, literal type, 265–266

CheckBox control

defined, 105

overview of, 956–957

as selection control, 112

WPF, 179

CheckedListBox control, 105, 112, 957–958

child controls, 158–159

Choose Data Source dialog box, database connections, 403–404

Choose decision statements, 946

Choose Items command, 38

Choose statement, 360–362, 935

Choose Toolbox Items command, 29

Clnt function, data type conversion, 268

Class statements, 242–243

classes, 563–573. See also exception classes

adding and modifying features, 555–557

boxing and unboxing, 580

catching events, 593–595

declaring constants, properties, and methods within, 589–590

declaring custom events, 595–599

declaring events, 591–592

defined, 563

deriving from other classes in inheritance, 550–551

drawing, 1084–1093

extension methods, 560–561

garbage collection, 584–589

generic, 1077–1078

graphics namespaces, 1081–1084

heap and stack performance, 577

information about, 1063–1064

instantiation details, 580–582

memory, 574–576

namespaces and, 613–614

object assignment, 577–578

in OOP languages, 547–549

parameter passing, 578–579

public interface of, 549

raising events, 593

shared methods, 600–602

shared variables, 599–600

structures vs., 574

System.Drawing namespace, 663–670

Visual Basic code editor outlining, 73–74

classes, declaring

accessibility clause, 565–566

attribute_list, 564

Implements interface, 570–573

inheritance clause, 567–568

Inherits parent_class, 569

Partial keyword, 564–565

Shadows keyword, 566–567

syntax, 563–564

Of type_list clause, 568–569

classic error handling, 391–395, 950

Clear method, Err object, 398

Click events, Button control, 956

client areas, of controls, 137

clipboard

drag-and-drop events. See drag-and-drop events
overview of, 519–520
using, 532–535

Clipboard object, My.Computer, 1117–1118
clipping regions, 721

CLng function, data type conversion, 268

Clock object, My.Computer, 1118–1119

Close All Documents command, 35

Close command, 18

Close Project command, 18

code

adding to WFD controls, 57–59
adding to WPF Designer controls, 68–69
browser applications, 226
coloring and highlighting, 79–81
file structure, 242
optimized, 50
regions, 243
reuse, 552, 648
separated from WPF user interface, 188–189
snippets, 81–85
transformation, 685–686

Code command, 20

Code Editor, 58–59

Code Snippets Manager command, 29, 83–85

Collection class, 619–620

collection classes, 619–646

Array.List class, 627–629
arrays. See arrays
dictionaries. See dictionary classes
generics, 643–645
NameValueCollection class, 632–633
overview of, 619–620
queues, 641–642
stacks, 640–641
StringCollection class, 629
strongly typed collections, 629–631

CollectionBase class, strongly typed collections, 629–631

CollectionsUtil class, dictionary classes, 639–640

collision resolution, 635

Color property, SolidBrush class, 706

ColorDialog, 105, 958–959

colors, Visual Basic code editor, 79–81

ComboBox control

defined, 105
overview of, 959–961
as selection control, 112
WPF, 179

ComboBoxItem control, 179

command builders, 420

command objects, 412, 420–425

command verbs, adding to controls, 56–57

Command window, Debug menu, 97–99

commands

adding, 43–45
Build menu, 26–27
customizing, 4
Data attribute, Path object, 210–211
Data menu, 27
Debug menu, 27
Edit menu, 18–20
File menu, 16–18
Format menu, 27–28
Help menu, 35
keyboard shortcuts, 47–48
menu, 15–16
modifying, 45–47
Project menu, 21–26
removing, 45
Test menu, 33–34
Tools menu, 29–33
View menu, 20–21
Visual Basic code editor margin, 73

CommandType property, command objects, 424

comma-separated expressions, 357–358

Comment Selection command, 254

comments

adding to metafile, 758
overview of, 253–254
XML, 254–258

Commit method, transactions, 419

Community menu, 35

comparison operators, 309–311, 932–933

Compile property page, 24–26

complex data binding, 451–455

Component class, 511, 951

ComponentResourceManager, 827–829

components. See Windows Forms controls and components

components, for custom controls, 509–510, 514

composite controls, 498, 506–508

composite formatting, 1105–1106

compound properties, 128

CompoundArray property, Pen object, 701

concatenation operators, 308–311, 932

Conditional attribute, 331, 351

conditional compilation, 243–244, 245–246

CONFIG constant

catching bugs using, 379
defined, 247
working with, 250

.config extension, 819–820

configuration and resources, 805–835

Application object, 830–835
configuration files, 816–821
environment variables, 807–811

- My namespace, 805–807
- overview of, 805
- Registry, 811–816
- resource files. *See* resource files
- configuration files, 816–821**
- Configuration Manager, 248**
- configuration settings**
 - affecting appearance of Visual Studio IDE, 5
 - choosing or changing Visual Studio, 5–6
- Connect to Database command, 29**
- Connect to Device command, 29**
- Connect to Server command, 29**
- connection objects**
 - data overview, 413–414
 - defined, 412
 - overview of, 414–417
- ConnectionString property, 414**
- console applications, 848, 1136**
- Const keyword, 300**
- constants**
 - CONFIG, 250
 - DEBUG, 247–250
 - debugging level, 251–252
 - predefined, 246–247
 - setting in code, 245
 - setting with project's compilation settings, 245–246
 - TRACE, 250
- constants, declaring**
 - accessibility keywords when, 300–301
 - within classes, 589–590
 - syntax for, 300
 - as type, 301
- constrained types, 652–654**
- constraints**
 - building DataSet with, 427
 - defined, 439
 - ForeignKeyConstraint, 440–441
 - UniqueConstraint, 441–442
- constructors**
 - class and structure, 1064–1065
 - generic, 650
 - instantiating classes using, 581–582
 - instantiating structures using, 583
 - Pen object, 698
 - queue memory management using, 642–643
 - stack memory management using, 641
- container controls**
 - dragging controls in and out of, 53
 - list of, 953
 - overview of, 111
 - Windows Forms controls, 109–110
 - WPF controls, 176–179, 189
- container types. *See* classes; structures**
- context menus, commands in, 16**
- ContextMenu control, WPF, 182**
- ContextMenu property, Windows Forms, 158**
- ContextMenuStrip component, 105, 115–116, 961**
- continuation characters, lines, 258–259**
- Continue command, Debug menu, 88**
- Continue statement, 364, 366, 371**
- control arrays, 142–143**
- Control class**
 - all controls inheriting from, 951
 - picking, 511–512
 - useful events, 917–920
 - useful methods, 915–917
 - useful properties, 911–914
- control statements**
 - decision statements, 943–946
 - defined, 943
 - looping statements, 946–948
- controls**
 - components vs., 121–123
 - custom. *See* custom controls
 - database. *See* database controls and objects
 - defined, 121
 - WFD. *See* WFD (Windows Form Designer)
 - Windows Forms. *See* Windows Forms controls and components
 - WPF. *See* WPF (Windows Presentation Foundation) controls
 - WPF Designer. *See* WPF (Windows Presentation Foundation) Designer
- ControlTemplate, XAML, 203**
- conversion functions**
 - data type, 266–269, 929
 - LINQ, 473–474
- copy and paste, WFD, 53, 65**
- Copy Button Image command, 45**
- Copy method, arrays, 623**
- CopyTo method, arrays, 623**
- Count function, LINQ, 470–471**
- Create New Test List command, 34**
- CreateInstance method, Array class, 622**
- CreateParameter method, command objects, 425**
- CreateText method, File class, 846–848**
- cryptology, 893–897**
- CryptoStream class, 848, 893–897**
- Crystal Reports, 781–783. *See also* Standard Report Creation Wizard**
- CrystalReportViewer, 107, 1007**
- CrystalReportViewer control, 781–782, 792–794, 799–801**
- Ctrl+C, 53**
- Ctrl+V, 53**
- CType function, 268–269, 298**

CType statement, 929

CurrencyManager object, 448–451

Cursor property, Windows Forms, 154–156, 158

Cursors class, 155

custom commands, 43–48

custom controls, 493–517

advantages of, 493

building from scratch, 508–509

building methods, 493–494

components, 509–510

composite controls, 506–508

custom component security, 514–517

derived controls, 502–506

in executable projects, 512–514

invisible controls, 510–511

picking Control class, 511–512

custom controls, library

assigning attributes, 499–501

creating project, 494

design time and runtime management, 501

implementing properties, methods and events, 498

making test project, 496–497

setting Toolbox icon, 495

testing, 498

testing in UserControl Test Container, 495–496

custom exception classes, 389–391, 1098

custom format specifiers, 1100–1105

CustomEndCap property, Pen class, 702

Customize command, 30, 43–45

CustomLineCap, Pen class, 702–703

CustomStartCap property, Pen class, 702

D

data

list of controls, 954

Windows Forms controls displaying, 113–114

WPF controls displaying, 180–181

Data Adapter Configuration Wizard, 420–424

data adapters, 412, 413–414, 419–424

Data attribute, XAML, 210–213

data binding, complex, 451–455

data binding, simple, 446–447

data containers, 412

Data menu, 27

data source

changing for reports, 792–794

for complex data binding, 451–455

selecting database objects for, 406–410

selecting for database connection, 402–405

for simple data binding, 446–447

Data Source Configuration Wizard, 402–405, 407–409

Data Structures and Algorithms Using Visual Basic .NET (McMillan), 634

data type characters, 264–266, 928

data types

anonymous, 298–299

conversion functions, 929

CType statement, 929

declaring enumerated, 295–298

determining variable behavior, 261

for drag-and-drop events, 523–524

elementary Visual Basic, 262–263

generics, constrained, 652–654

generics, multiple, 650–652

generics, tying to, 647–648

literal type characters, 928

nullable, 299–300

parsing methods, 269

summary of, 930

types of, 927

widening conversions, 269

database controls and objects, 401–455

adding data controls to form, 406–410

automatically created objects, 410–412

command objects, 424–425

complex data binding, 451–455

connecting to data source, 402–405

connection objects, 414–417

CurrencyManager object, 448–451

data adapters, 419–424

data overview, 413–414

DataColumn object, 435–437

DataRelation object, 437–440

DataRow object, 433–435

DataRowView object, 445–446

DataSet object, 426–430

DataTable class, 430–433

DataGridView object, 442–445

ForeignKeyConstraint object, 440–442

other data objects, 412–413

simple data binding, 446–447

transaction objects, 417–419

Database Tools, Options dialog box, 33

DataColumn objects, 427, 435–437

DataConnector control, 117

DataContext class, 487–488

DataFormats class, 523–524

DataGridView control, 105, 113, 961–962

DataNavigator control, 117

DataObject class, 528–533

DataRelation object, 437–440

DataRow objects, 411, 433–435, 445

DataRowView object, 445–446

DataSet component, 105, 962

DataSet objects

- complex data binding with, 452
- data overview, 413–414
- defined, 411
- LINQ to, 488–491, 1075
- overview of, 426–430
- simple data binding with, 446–447
- DataSource property, complex data binding, 452**
- DataTable class, 430–433**
- DataTable objects**
 - building DataSet with, 427
 - customizing view, 442–445
 - defined, 411
 - for simple data binding, 446–447
- DataGridView objects, 442–447**
- Date operator, 318–320, 935–936**
- DateTimePicker control, 105, 112, 962–964**
- DEBUG constant, 247–250, 379**
- Debug menu**
 - Breakpoints window, 95–97
 - Command window, 97–99
 - defined, 27
 - Immediate window, 97–99
 - overview of, 87–90
 - Visual Basic code editor, 85
 - Windows submenu, 91–95
- Debug Tests command, 34**
- DEBUG_LEVEL constant, 251–252**
- DebuggerHidden attribute, 331**
- DebuggerStepThrough attribute, 331**
- debugging, 87–99**
 - after creating new project, 11
 - level constants, 251–252
 - overview of, 87
 - using Debug menu. See Debug menu
 - using Stop statement, 398
- Debugging setting, Options dialog box, 33**
- Debug.Print statement, 97–98**
- Debug.WriteLine statement, 921**
- decision statements, 353–362**
 - Choose, 360–362, 946
 - If and If, 359–360, 945–946
 - multiline If Then, 355, 944
 - Select Case, 355–358, 944–945
 - single-line If Then, 353–354, 943–944
- Default Style command, 46**
- DefaultEvent attribute, 329, 500**
- DefaultProperty attribute, 330, 500**
- DefaultValue attribute, 330, 501**
- deferred validation, 146–147**
- delegate variables, 302–303**
- delegates, 347–350, 596–599, 956**
- Delete, 45**
- Delete All Breakpoints command, Debug menu, 90**
- Delete Tab command, 38**
- DeleteSetting method, Registry values, 812–813**
- derived classes, from generics, 655–656**
- derived controls, 502–505**
- Descending keyword, LINQ syntax, 462**
- Description attribute, 330, 500**
- design time**
 - building data source at, 446
 - creating controls at, 123–124
 - creating data adapter control at, 421
 - creating event handlers at, 139–140
 - managing controls running at, 501
 - modifying control properties at, 127–132
- Designer command, 20**
- DesignMode property, controls, 501**
- DesignTimeVisible attribute, 500**
- Device Emulation Manager command, 30**
- Device Tools, Options dialog box, 33**
- dialog boxes, Windows Forms**
 - displaying, 117
 - using Windows Forms as, 170–171
 - wizards as type of, 172–173
- DialogResult property, 170–171, 224**
- dialogs, list of, 954**
- dictionary classes, 633–640**
 - CollectionsUtil, 639–640
 - HashTable, 635–636
 - HybridDictionary, 636
 - ListDictionary, 634–635
 - properties and methods, 633
 - SortedList, 639
 - StringDictionary, 639
 - strongly typed, 636–638
- DictionaryBase class, 636–638**
- digital ink controls, WPF, 184–185**
- digital signatures, 514–516**
- Dim keyword, 275**
- dimensions, array, 621–623**
- Direct3D, 901–908**
- DirectCast statement, 269, 930**
- directories, 13, 83–85**
- Directory class, 1143–1144**
- DirectoryEntry component, 105, 964**
- DirectoryInfo class, 1147–1149**
- DirectorySearcher component, 105, 964**
- DirectX, 901–902**
- display area, controls, 138**
- Dispose method, 587–589, 724–725**
- Distinct keyword, LINQ, 471, 1069**
- Do Loop statements, 369–371, 947–948**
- Dock property, controls, 136–137**
- docking icons, 35–36**
- DockPanel control, WPF, 177**
- Document property, PrintPreviewDialog component, 1003–1004**

- documents, WPF, 184, 218–221**
 - DocumentViewer control, WPF, 184, 221**
 - DoDragDrop method, 520–521, 526–528**
 - DomainUpDown control, 964–965**
 - DoubleAnimationUsingKeyFrames element, XAML, 207–208**
 - drag source control, 519–523**
 - drag-and-drop events, 520–532**
 - accepting dropped files, 525–526
 - changing format names, 528–529
 - data types available, 523–524
 - dragging multiple data formats, 529–531
 - dragging objects, 526–528
 - dragging within applications, 524–525
 - example of, 521–523
 - overview of, 520–521
 - DragEnter event, 520–521, 523–524**
 - dragging controls**
 - Windows Form Designer, 53
 - WPF Designer, 66
 - DragOver event, 520–521**
 - DrawBézier method, 675–676**
 - DrawBéziers method, 676**
 - DrawClosedCurve method, 677–678**
 - drawing, 661–695**
 - advanced transformations, 688–691
 - anti-aliasing, 682–684
 - basic transformations, 684–688
 - events, 693–695
 - filling methods, 679
 - graphics, 673
 - overview of, 661–663
 - saving and restoring graphics state, 692–693
 - simplifying printing and, 777–780
 - text, 728
 - drawing classes, 1084–1093**
 - Bitmap class, 1092
 - Brush class, 1089
 - Graphics object, 1084–1087
 - GraphicsPath object, 1089–1090
 - Image class, 1091–1092
 - Metafile class, 1093
 - Pen object, 1088
 - StringFormat object, 1090–1091
 - drawing methods, 673–682**
 - DrawBézier method, 675–676
 - DrawBéziers method, 676
 - DrawClosedCurve method, 677–678
 - DrawPath method, 678–679
 - types of, 673–675
 - drawing namespaces, 663–673**
 - System.Drawing namespace, 663–665, 1081–1082
 - System.Drawing.Drawing2D namespace, 665–668, 1082–1083
 - System.Drawing.Imaging namespace, 668–669, 1083–1084
 - System.Drawing.Printing namespace, 672–673, 1084
 - System.Drawing.Text namespace, 669–670
 - DrawPath method, Graphics object, 678–679, 719–720**
 - DrawString method, Graphics object, 728**
 - DriveInfo class, 1146–1147**
 - drop target**
 - accepting dropped files, 524–525
 - defined, 519
 - dragging within applications, 524–525
 - events received by, 520
 - example of, 521–523
 - dynamic properties, 816–819**
- ## E
- e.Cancel parameter, FormClosing event handler, 165–166**
 - Edit menu, 18–20**
 - editing**
 - button images, 45–46
 - test run configurations, 34
 - Visual Basic code, 194–197
 - XAML, 190–194
 - 8-bit UTF encoding, 840**
 - elements, namespace, 610**
 - elevated privileges. See UAC (User Account Control) security**
 - Ellipse control, WPF, 183**
 - Ellipse object, XAML, 209–210**
 - Else keyword, multilineIf Then statement, 354, 355**
 - embedded resources, 823–824**
 - Enabled property, Windows Forms, 158**
 - EnableRefactoring OnRename setting, WFD, 51**
 - encapsulation, in OOP languages, 549–550**
 - encryption providers, 894**
 - End Function statement, 343**
 - End Region statement, 243**
 - endpoints, WCF, 874, 877–878**
 - EndPrint event, PrintDocument, 763**
 - Entity Framework, LINQ to Entities, 488**
 - Enum statement, 358**
 - enumerated types**
 - declarations, 924–925
 - formatting, 1106–1107
 - overview of, 295–298
 - Select Case statements and, 358

enumerations

- System.Drawing.Drawing2D namespace, 666–668
- System.Drawing.Text namespace, 670–671

enumerators, looping statements, 367–369**Environ function, 808–809****Environment object, 809–811****Environment setting, Options dialog box, 33****environment variables**

- overview of, 807–808
- setting, 808
- using Environ function, 808–809
- using System.Environment object, 809–811

Err object, 397–398**error handling, 377–399**

- bugs vs. unplanned conditions, 377–381
- classic, 391–395, 950
- classic vs. structured, 396–397
- debugging, 398
- Err object and, 397–398
- global exception handling, 382
- structured. See structured error handling
- throwing exceptions, 949–950

Error List window, 11**error-handling mode, classic error handling, 395****ErrorProvider component**

- defined, 105
- overview of, 965–966
- providing feedback to user, 114
- as selection control, 112

errors

- Visual Basic code editor displaying, 79–80
- XAML code, 64

event handlers

- adding code to controls in Code Editor, 58–59
- creating at design time, 139–140
- editing Visual Basic code, 194
- overview of, 139
- partial methods vs., 350–351
- relaxed delegates simplifying, 350
- setting at runtime, 141–142
- WithEvents, 141

event sequences, Control class, 921**EventLog component, 105, 966****events**

- Application class, 833–835, 1112
- assigning attributes to controls, 499–501
- catching, 593–595
- classes defining, 547–548
- connection object, 417
- control array, 142–143
- Control class, 917–920
- CurrencyManager object, 449
- custom control, 498
- DataTable class, 433

declaring, 591–592, 1065

declaring custom, 595–599

drawing, 693–695

FileSystemWatcher class, 865–866

firing or throwing, 139

Form class, 1057–1061

MaskedTextBox control, 984

MDI, 165–166

OpenFileDialog component, 994

PrintDocument class, 762–763

raising, 593

received by drag source control, 521

received by drop target, 520

RichTextBox control, 1011–1012

TreeView control, 1030–1031

validation, 143–147

example programs, in this book, 1157–1176**Exception class, 385–386****exception classes, 1095–1098****Exceptions command, Debug menu, 89–90****Exclude From Project command, 22****executable projects, custom controls in, 512–514****ExecuteNonQuery method, 419, 425****ExecuteReader method, 425****ExecuteScalar method, 425****Exit Function statement, 343–344****Exit statement**

Do Loop, 371

For Each loop, 366

For loop, 364

overview of, 371

While End loop, 371

Exit Sub statement, 342, 374**ExitCode property, UAC, 540****Expander control, WPF, 177****Export Template command, 18–19****Expression Design, 188–189, 193****Extensible Application Markup Language. See XAML (Extensible Application Markup Language)****Extensible Markup Language. See XML (Extensible Markup Language)****extension methods**

creating, 941

generics and, 643–644, 657–658

OOP languages and, 560–561

overview of, 344–346

extension methods, LINQ

method-based queries, 474–475

method-based queries with lambda functions, 476–477

overview of, 477–480, 1072–1076

types of, 472–473

extensions, generic, 1078–1079

external reports, 799

External Tools command, 30

F

Fail method, Debug, 249

feedback

Windows Forms controls, 114–115

WPF controls, 181

fields, report, 786

Fields page, Standard Report Creation Wizard, 785–786

FIFO (first-in, first-out) order, 641–643

File class, 1144–1146

File menu, 16–18

file numbers, 851

FileDrop data, 525–526

FileInfo object, 862–864, 1149–1150

files, 7

FileStream class, 838, 839–840

file-system, .NET framework classes. See .NET Framework classes, file system

file-system, Visual Basic methods, 849–856

binary-file access, 856

file methods, 849–852

file-system methods, 852

overview of, 1139–1141

permissions, 849

random-access file, 853–855

sequential-file access, 853

FileSystem class, 1141–1143

FileSystem object, My.Computer, 1119–1120

FileSystemInfo object, 864

FileSystemWatcher component, 105, 864–866, 966, 1151

Fill method, data adapter, 419

Fill property, 183

filling methods, 679

FillPath method, Graphics object, 719–720

Finally block, structured error handling, 384–385

Find Symbol command, 18

firing events, 139

first-in, first-out (FIFO) order, 641–643

FitPictureToMargins subroutine, 775–779

Fix function, data type conversion, 268

FixedDocument control, WPF, 221

FixedDocument objects, XPS, 221

FixedPage object, WPF, 221

Floating, Dockable, Tabbed Document command, 34

FlowDocument control, WPF, 218–221

FlowDocumentPageViewer control, WPF, 184

FlowDocumentReader control, WPF, 184

FlowDocumentScrollViewer control, WPF, 184

FlowLayoutPanel control, 105, 109–110, 967

FolderBrowserDialog, 105, 967–968

fonts

design units, 740

Font property, 128, 158–159

FontDialog component, 105, 968–970

FontFamily class, 739–743

metrics, 739–743

System.Drawing.Text namespace, 669–670

For Each loop, 365–367, 946–947

For loop, 362

For Next loop, 362–365, 946

ForeColor property, Windows Forms controls, 158–159

ForegroundColor property, HatchBrush class, 709

ForeignKeyConstraint object, 440–442

Form class

events, 1057–1061

implementing AutoRedraw using, 750

methods, 1053–1056

properties, 1047–1053

Window class vs., 189, 224

Form control, 116

Form Designer, 11, 157

Form1.vb command, 35

Format menu, 27–28, 53

format specifiers

composite, 1105–1106

date and time, 1099–1100

enumerated type, 1106–1107

numeric, 1103–1105

FormatFlags property, StringFormat object, 731–734

formats. See also text formatting

changing object names in drag-and-drop events, 528–529

dragging multiple data, 529–531

text. See text formatting

FormClosed events, 165–166

FormClosing event, 147, 165–166

Formula Workshop, reports, 796, 798

Frame control, WPF, 184, 227–229

FreeFile method, file-systems, 851

Friend keyword

class declarations with, 566

constant declarations with, 301

event declarations with, 592

read-only strongly typed collections, 631

subroutine accessibility clause, 333

variable declaration accessibility clause, 271

From clause, LINQ, 460–461, 1067
Full Screen command, 20–21
Function keyword, lambda functions, 346
functions, 343–351
 data type conversion, 268
 defined, 327
 defining code regions, 243
 extension methods, 344–346
 lambda (inline), 346–347, 941
 LINQ, 472–476, 1072–1076
 overview of, 343–344
 partial methods, 350–351
 passing delegate variables into, 303
 passing parameters to, 578
 property procedures, 344, 940
 relaxed delegates, 347–350
 syntax for, 939–940
 XEElement, 484–485

G

garbage collection

issues with, 723–725
 overview of, 584–585
 using `Dispose` method, 587–589
 using `Finalize` method, 585–587

GDI+ (Graphics Device Interface+), 662–663, 727

General Development settings, 5–6

generalization and inheritance, 553–554

generics, 647–658

advantages of, 647–648
 classes, 1077–1078
 collection classes defined by, 644–645
 constrained types, 652–654
 constructors, 650
 defining, 648–650
 derived classes, 655–656
 extension methods and, 657–658
 extensions, 1078–1079
 imports aliases, 655
 methods, 656, 1079
 multiple types, 650–652
 overview of, 643–644
 predefined, 656
 prohibited, 1079
 with `Of type_list` clause, 567

`GetAllSettings` method, Registry values, 812–813

`GetDataObject` method, clipboard, 532–533

`GetDataPresent` method

drag-and-drop events, 523–524
 dragging objects, 528

using clipboard, 532–533

`GetException` method, `Err` object, 398

`GetPixel` method, `Bitmap` class, 748, 753–754, 757

`GetSetting` method, Registry values, 812–813

`GetStringBuilder` method, 845

global exception handling, 382

Go To command, 19

`GoBack` method, `NavigationService`, 227

`GoForward` method, `NavigationService`, 227

GotDotNet Visual Basic Power Pack, 1045–1046

`GoTo` statement, 372–374, 948

graphics, 1081–1093

controls represented by, 121
 drawing classes, 673, 1084–1093
 namespaces, 1081–1084
 Windows Forms controls displaying, 116
 WPF controls displaying, 182–183

Graphics Device Interface+ (GDI+), 662–663, 727

Graphics object

advanced transformations, 688–691
 anti-aliasing, 682–684
 basic transformations, 684–688
 drawing events, 693–695
 drawing methods. See drawing methods
 drawing using, 661–662
 other properties/methods, 679–682
 properties and methods, 1085–1087
 saving and restoring graphics state, 692–693

`GraphicsPath` object, 719–723, 1089–1090

`GraphicsState` object, saving and restoring, 692–693

Grid, XAML, 203

Grid control, WPF, 177

Grid Size setting, WFD, 50

`GridSplitter` control, WPF, 177

`GridView` control, WPF, 177

Group By clause, LINQ, 467–471, 1069–1070

Group Expert command, customizing reports, 796–797

Group Join clause, LINQ, 466

group properties, WPF Designer control, 68

Group Sorting page, Standard Report Creation Wizard, 788–789

GroupBox control

containing and arranging with, 110
 defined, 106
 overview of, 970
 WPF, 178

Grouping page, Standard Report Creation Wizard, 786–787

gutters, printing text, 767

H

“has-a” relationships, inheritance, 554–555

hashing, 893–897

HashTable class, 635–636, 639

HatchBrush class, 706, 709

HatchStyle property, HatchBrush class, 709

heap performance, structures vs. classes, 577

Help menu, 35

HelpProvider component, 106, 114, 970–971

hidden files, 237–241

Hide command, 35

hiding parent features, derived controls, 505–506

hierarchies

inheritance, 552

namespace, 606, 615–617

WPF control, 189

XPS documents, 221

highlighting, 79–81

hives (subtrees), Registry, 811–812

Horizontal Spacing command, 28–29

hosting services, 882–883

How Do I command, Help menu, 35

HScrollBar control

defined, 106

initiating action, 116

overview of, 971–972

as selection control, 112

HTML Designer, Options dialog box, 33

HybridDictionary class, 636

I

icons

component, 121–122

Visual Basic code editor margin, 72–74

Windows Forms, 156–158

IDE (integrated development environment), 1–13

Command window, 98

complexity of, 3

configurations, 5–6

creating project, 9–11

customizing, 43–48

menus. See menus

projects and solutions, 6–7

saving project, 12–13

starting, 7–9

toolbars, 35–36

varying appearances of, 4–5

windows, 36–41

WPF in, 190–197

identifiers, rules for valid, 278

IEnumerable result, LINQ, 465–468, 483–487

If statement, 360, 935, 945–946

If Then Else statement

conditional compilation statements vs., 243–244

multiline If Then statement, 355

Select Case statement vs., 356

single-line If Then statement, 354

IIf statement, 359–360, 935, 945–946

Image class

Bitmap properties and methods inherited from, 747

overview of, 757–760

properties and methods, 1091–1092

Image control, WPF, 183

image processing, 745–760

Bitmap class, 747–748

Image class, 745–747

implementing AutoRedraw, 751–752

loading Bitmaps, 748

Metafile class, 757–760

pixel-by-pixel operations, 753–757

saving Bitmaps, 748–750

Image property, PictureBox, 750–751

ImageList component, 106, 117, 972

ImageList property, 129

Immediate window, Debug menu, 91, 97–99

Implements interface.subroutine, 341–342

Implements keyword, 570–573, 592

Import and Export Settings Wizard, 5–6

Import/Export Settings command, 30

Imports statement

accessing namespace inside file with, 253

code file structure, 242

defining alias for generics using, 655

importing namespaces, 606–609

resolving namespaces, 616

Indent methods, Debug object, 249

inferred data types, 281

Info object, My.Computer, 1120–1121

InfoMessage event, connection object, 417

information hiding, 549–550

inheritance, 550–557

adding and modifying class features, 555–557

in executable projects, 512–514

“has-a” and “is-a” relationships, 554–555

hierarchies, 552

interface, 557

multiple, 552

overview of, 550–551

refinement and abstraction, 552–554

using interface vs., 571–573

inheritance_mode, 332–333, 939, 940

Inherits Control statement, 508–509

Inherits statement, class declarations, 567–569

initialization

- expressions, 281–282, 301, 924
- initializing arrays, 282–283
- initializing object arrays, 283
- initializing XML variables, 284–285
- multiple variable declarations, 285–286

InitializeComponent method,

PageFunction, 230

ink controls, WPF, 184–185**InkCanvas control, WPF, 185****InkPresenter control, WPF, 185****inline functions. See lambda (inline) functions****InnerException value, Exception class, 385****Insert File As Text command, 19****instantiation, 549, 580–584****Int function, data type conversion, 268****Integer data type, 263****integrated development environment. See IDE (integrated development environment)****integrated validation, 143–145****IntelliSense**

- for classes, structures and modules, 611–612
- in Debug menu's Command window, 98
- defined, 20
- for generics, 647
- in Visual Basic code editor, 77–79
- weakness of in WPF Designer, 62
- in XAML, 197

interface inheritance, 552, 557, 573**interfaces**

- class declarations implementing, 570–571
- Implements interface.subroutine, 341–342

Internet Download Manager, 118**Interop Forms Toolkit, 1044****InterpolationMode property, anti-aliasing, 683–684****intuitive encapsulation, 550****invisible controls, 510–511****Is keyword, Select Case statement, 357****Is operator, 309****“is-a” relationships, inheritance, 554–555****IsDefault property, Button class, 224****IsFalse operator, operator overloading, 322****IsNot operator, 309****IsTrue operator, 322****Item method, strongly typed collections, 630, 631****iterators, looping statements, 369****J****Join keyword, LINQ, 466–467, 1069****joins, line, 259****K****key collisions, 635****key frames, animation in XAML, 207–208****Keyboard object, My.Computer, 1121–1122****keyboard shortcuts, 47–48****L****Label control**

- defined, 106
- displaying data, 113
- summary of, 973
- WPF, 181

labels, line, 259**lambda (inline) functions**

- method-based queries with, 476–477
- overview of, 346–347
- summary of, 941

Language Integrated Query. See LINQ (Language Integrated Query)**last-in, first-out (LIFO) order, 640–641****layout controls, 176–179****LayoutMode, Windows Form Designer, 50****lifetime, variable, 261–262****LIFO (last-in, first-out) order, 640–641****Like operator, 309–310****line continuation characters, 254, 258–259****Line control, WPF, 183****Line object, XAML, 208–209****LineAlignment property, DrawString method, 729****LinearGradientBrush class, 706, 709–713****LineLimit flags, 731–734****lines, joining, 259****Link page, Standard Report Creation Wizard, 784–785, 786****linked lists, 634–635****LinkLabel control, 106, 116, 973–974****LINQ (Language Integrated Query), 457–492**

- advanced syntax, 466–470
- aggregate functions, 470–471
- basic syntax, 1067–1071
- extending, 477–479
- extension methods, 474–475
- functions, 472–474, 1072–1076
- introduction to, 458–460
- lambda functions, 476–477
- limiting results, 471–472, 1069–1070
- LINQ into XML, 482–483, 1073
- LINQ out of XML, 484–487, 1073–1075
- LINQ to ADO.NET, 487
- LINQ to DataSet, 488–491, 1075
- LINQ to Entities, 488

LINQ (Language Integrated Query) (continued)

- LINQ to Objects, 480–481, 489–490
- LINQ to SQL, 487–488
- LINQ to XML, 481, 1073
- overview of, 457–458
- set operations, 471
- type inference when working with, 281
- using anonymous types, 298–299
- using query results, 1071
- using results, 465
- XML literals, 481–482

LINQ (Language Integrated Query), syntax

- basic syntax, 460–464
- From clause, 460–461
- Group By clause, 467–470
- Join keyword, 466–467
- Order By clause, 462
- Select clause, 462–464
- Where clause, 461–462

List View command, 38

ListBox control

- collection properties, 130–131
- defined, 106
- as selection control, 112
- useful properties, 975–976
- WPF selection control, 179

ListBoxItem control, WPF, 179

ListDictionary class, 634–636, 638

ListView control, 106, 113, 976–981

literal type characters, 265–266, 928

Load Metadata File command, 34

loading, Bitmaps, 748

Localizable attribute, 330

localization resources, 826–827

Locals window, Debug menu, 91

location, choosing for saved project, 12–13

Location property, 128

Lock Controls command, 29

LockBitmap method, Bitmap object, 754–757

logical operators, 311–312, 933–934

login forms, Windows Forms, 153–154

LongCount function, LINQ, 470–471

looping statements, 362–374

- Do Loop, 369–371
- For Each loop, 365–367
- enumerators, 367–369
- Exit and Continue, 371
- iterators, 369
- For Next loops, 362–364
- non-integer For Next loops, 364–365
- overview of, 362
- syntax for, 946–948
- While End loop, 371

lower bounds, arrays, 622

M

Macro Explorer, 31

macros, 30–32, 45

Main subroutine, 242

Make Same Size command, Format menu, 28

MakeKeyAndIV subroutine, 896

margin icons, 72–74

MarginBounds parameter, PrintDocument, 766

MaskedTextBox control, 106, 113, 982–984

master control, WFD, 52–53

mathematics, transformations, 684–685

MatrixOrder.Append method, 686

MatrixOrder.Prepend method, 686, 692–693

Max function, LINQ, 470–471

MDI (multiple-document interface) application

- defined, 149
- events, 165–166
- features, 162–165
- overview of, 161–162
- SDI vs., 167

MdiChildren collection, 163

Me object, 806–807

MeasureCharacterRanges method, Graphics object, 737–739

MeasureString method, Graphics object, 736–739

media, WPF controls, 182–183

MediaElement control, WPF, 183

memory

- managing with stacks, 641
- structures vs. classes, 574–577

MemoryStream class, 838, 840–841

Menu control, WPF, 182

MenuItem control, WPF, 182

menus, 15–35

- Build menu, 26–27
- commands in, 15–16
- Community menu, 35
- customizing appearance, 4
- customizing commands, 43–47
- Data menu, 27
- Debug menu, 27
- Edit menu, 18–20
- File menu, 16–17
- Format menu, 27–28
- Help menu, 35
- list of controls, 953
- Project menu, 21–26
- starting new project, 10
- Test menu, 33–34
- Tools menu, 29–33
- View menu, 20–21

MenuStrip control, 106, 115, 985–986

- MergableProperty** attribute, 330
- Message value, Exception class**, 385
- MessageQueue component**, 106, 987
- messages, WCF**, 874
- Metadata Exchange endpoint (MEX)**, 878
- Metafile class**, 757–760, 1093
- method-based queries, LINQ**, 474–477, 1076
- methods. See also extension methods**
 - accessing Registry with native VB, 812–813
 - Application class, 831–833, 1111
 - ArrayList class, 627–629
 - assigning attributes to controls, 499–501
 - BinaryReader and BinaryWriter class, 841–843
 - Bitmap class, 748
 - classes defining, 547–548
 - Clipboard object, 534
 - command objects, 425
 - connection object, 416
 - Control class, 915–917
 - CurrencyManager object, 448
 - DataRow class, 434–435
 - DataSet object, 429–430
 - DataTable class, 432
 - DataView object, 445
 - Date operator, 318–320
 - declaring within classes, 589–590
 - dictionary classes, 633
 - Directory class, 856–857
 - DirectoryInfo object, 861–862
 - drawing, 673–682
 - Environment object, 810–811
 - Err object, 397
 - executing code for controls through, 138–139
 - extension, 344–345
 - file, 849–852
 - File class, 846–848, 858–859
 - FileInfo object, 862–864
 - FileStream class, 839
 - file-system, 852
 - FileSystemWatcher class, 865
 - FontFamily class, 740
 - Form class, 1053–1056
 - generic, 656, 1079
 - GraphicsPath object, 722–723
 - Image class, 746–747
 - implementing for custom controls, 498
 - IntelliSense displaying info on parameters, 78
 - LinearGradientBrush class, 709–710
 - Metafile class, 758
 - My.Computer.FileSystem, 869–870
 - MySettings namespace, 820
 - NameValueCollection class, 632
 - overloading, 339
 - partial, 350–351, 941
 - Path class, 867–868
 - PathGradientBrush class, 714–715
 - Pen object, 698–699
 - random-access file, 854–855
 - RegistryKey objects, 814–816
 - sequential-file access, 853
 - shared, 600–602
 - stacks, 641
 - streams, 839, 1131–1137
 - strongly typed collections, 630
 - strongly typed dictionary, 637
 - subroutines and functions as, 327
 - text formatting, 730
 - TextReader class, 843
 - TextureBrush class, 706–707
 - TextWriter class, 844
 - Visual Basic code editor outlining, 73–74
- methods, Windows Forms controls/components**
 - ErrorProvider component, 965–966
 - ListBox control, 976
 - ListView control, 976–979
 - MonthCalendar control, 989
 - PageSetupDialog component, 994
 - PictureBox control, 997
 - RichTextBox control, 1011
 - TabLayoutPanel control, 1020
 - TextBox control, 1022–1023
 - Timer component, 1023
 - ToolTip component, 1027
 - TreeView control, 1030
 - WebBrowser control, 1033
- MEX (Metadata Exchange) endpoint**, 878
- Microsoft namespace**, 606, 886
- Microsoft Office Keyboard, Options dialog box**, 33
- Microsoft Power Packs**, 1043–1045
- Microsoft Word, printing with**, 762
- MicrosoftReportViewer control**, 107
- Min function, LINQ**, 470–471
- Mod operator**, 308
- module scope**, 290
- Module statements**, 242–243
- modules**
 - namespaces and, 253, 613–614
 - structure. See program and module structure
 - Visual Basic code editor outlining, 73–74
- Modules window, Debug menu**, 94
- MonthCalendar control**, 106, 112, 987–990
- Most Recently Used list (MRU)**, 168–170
- mouse**
 - cursors, 154–156
 - events, 921
- Mouse object, My.Computer**, 1122
- Move Down command**, 38

move events, controls, 922

Move Up command, 38

moving controls

Windows Form Designer, 53

WPF Designer, 66

MRU (Most Recently Used) lists, 168–170

MruList class, 168–170

multiline If Then statements, 355, 944

multiple inheritance, 552, 571

multiple types, 650–652

multiple-document interface. See MDI (multiple-document interface) application

MustInherit class, 567–568, 705

MustOverride keyword, 332

My namespace

Me object vs., 806–807

My.Application, 1113–1116

My.Computer. See My.Computer

My.Forms, 1128

My.Resources, 1128

My.User, 1129

overview of, 805–806

sections, 807

shortcut keys, 886

My.Computer, 1117–1128

Audio object, 1117

Clipboard object, 1117–1118

Clock object, 1118–1119

FileSystem object, 1119–1120, 1153–1155

FileSystem.SpecialDirectories property, 1155–1156

Info object, 1120–1121

Keyboard object, 1121–1122

Mouse object, 1122

Name object, 1122

Network object, 1122

overview of, 1117

Port object, 1123–1125

Registry object, 812–813, 1125–1126

Screen object, 1126–1128

MySettings namespace, 819–820

_MyType constant, 247

N

name clause, variable declarations, 278

Name object, My.Computer, 1122

namespace collision, 605

namespace pollution, 605

namespace scope, 291

Namespace statements, 242, 291, 612

namespaces, 605–617

aliases, 609–610

automatic imports, 607–609

categorizing code using, 252

classes, structures and modules, 613–614

creating, 611–612

drawing, 663–673

elements of, 610

graphics, 1081–1084

nested, 611

overview of, 252–253, 605

resolving, 614–617

root, 610

using Imports statement with, 606–609

Visual Basic code editor outlining, 73–74

Windows Forms vs. WPF, 187–188

namespaces, useful, 885–908

cryptography, 893–897

Direct3D, 901–908

high-level, 886–888

Microsoft, 886

overview of, 885

reflection, 897–901

regular expressions, 888–891

System, 887–888

XML, 891–893

NameValueCollection class, 632–633

naming conventions

building custom exceptions, 390

new control libraries, 494

objects in drag-and-drop events, 528–529

subroutine, 334

subroutines, 334

variable, 303–304

narrowing conversions, data types, 267–269

navigation, WPF, 184, 228

NavigationService.Navigate, 226–227

NearestNeighbor interpolation, anti-aliasing, 683

nested namespaces, 611

.NET Framework classes, file system, 856–868

Directory class, 856–857, 1143–1144

DirectoryInfo class, 861–862, 1147–1149

DriveInfo class, 860, 1146–1147

File class, 858–859, 1144–1146

FileInfo object, 862–864, 1149–1150

FileSystem class, 1141–1143

FileSystemInfo object, 864

FileSystemWatcher class, 864–866, 1151

Path class, 866–868, 1152–1153

.NET framework classes for file system

My.Computer.FileSystem, 868–870, 1153–1155

My.Computer.FileSystem.

SpecialDirectories, 870–871, 1155–1156

Network object, My.Computer, 1122

NetworkStream class, 848
New Breakpoint command, Debug menu, 90
New Horizontal Tab Group command, 35
New Items commands, 21
New keyword, 280, 580
New Project command, 16–17
New Project dialog, Visual Studio, 9–11
New Test command, 34
New Vertical Tab Group command, 35
New Web Site command, 16
New Window command, 34
Next Method command, 20
NoClip flags, 731–734
non-client areas, controls, 137
non-integer For Next loops, 364–365
NonSerializedAttribute, 331
Not operator, 311, 312
Nothing value, 309
notification icons, Windows Forms, 157–158
NotifyIcon control
 defined, 106
 displaying icons using, 157–158
 overview of, 990–991
 providing feedback to user, 114
NotInheritable keyword, class declarations, 567–568
NotOverridable keyword, subroutines, 332
nullable types, 299–300, 324
null-propagation rules, 300
NumericUpDown control, 106, 112, 991–992

O

object arrays, initializing, 283–284
Object Browser, 388–389
Object class, 549
objects
 assigning, 577–578
 boxing and unboxing, 580
 class instantiation details, 580–582
 classes as, 549
 data, 410–414
 FlowDocument, 219
 LINQ to Objects, 480–481
 XAML, 197–201, 205, 208–213
Obsolete attribute, 331
OdbcCommand class, 424–425
OdbcConnection object, 414–416
Of keyword, 648–649
Of type_list clause, class declarations, 568–569
OK buttons
 dialog boxes, 171
 login forms, 153
 wizards, 172
OleDbCommand class, 424–425
OleDbConnection object, 414–415
On Error statement, classic error handling
 On Error GoTo -1 statement, 394–395
 On Error GoTo 0 statement, 393–394
 On Error GoTo statement, 391–392
 On Error Resume Next statement, 392–393, 397
 error-handling mode, 395
 structured error handling vs., 396
online references
 _MyType constant, 247
 8-bit UTF encoding, 840
 Attribute class, 332
 attributes, 271, 564
 certificate authorities, 517
 Community menu, 35
 computer bugs, history of, 378
 connection objects, 414
 Crystal Reports, 781
 custom exception classes, 1098
 DirectX development kit, 901–902
 event sequences, 921
 EventLog component, 966
 exception handling, 388
 Expression Design, 188
 file formats, 750
 FlowDocument control, WPF, 220
 graphics API in Windows Vista, 727
 graphics terminology, 661
 hosting services, 882
 LINQ, 488, 491–492
 LINQ addendums, 458
 Microsoft Power Packs, 1043
 naming conventions, Microsoft Consulting Services, 304
 performance counters, 996
 Power Toys Pack Installer, 1046
 Process component, 1004
 regular expressions, 890
 SerialPort component, 1013
 signing assemblies, 517
 snippets, 81
 strongly named assemblies loopholes, 514
 System.Windows.Controls classes, 1042
 StreamReader and StreamWriter class, 844
 third-party controls, 118
 ToolboxBitmap attribute, 495
 ToolBoxBitmap attribute, 501
 transactions, 419
 Visual Studio free Express editions, 5

online references (continued)

- Visual Studio Tools for Office, 762
- Windows Communication Foundation, 884
- Windows Forms controls and components, 951
- Windows service applications, 1013
- Windows Vista underlying graphics model, 662
- WPF Designer, 61
- XAML, 197
- XPS documents, 221

OOP (object-oriented programming) concepts, 547–562

- classes, 547–549
- encapsulation, 549–550
- extension methods, 560–561
- inheritance, 550–557
- overloading, 559–560
- polymorphism, 558–559

Opacity property, forms, 150–153

Open File command, 16–17

Open Project command, 16

Open Web Site command, 16

OpenFileDialog, 106, 992–994

OpenText method, File class, 846–848

operands, 307

Operator statement, 318–320

operators, 307–325

- arithmetic, 307–308, 931–932
- assignment, 314–316, 935
- bitwise, 312–313, 934
- Choose, If and IIf statements, 935
- comparison, 309–311, 932–933
- concatenation, 308, 932
- Date and TimeSpan, 318–320, 935–936
- logical, 311–312, 933–934
- with nullable types, 324
- in order of precedence, 313–314, 934–935
- overloading, 321–324, 936–937
- StringBuilder class, 316–318

Optimized Code Generation, WFD, 50

Option Compare, Compile property page, 26

Option Explicit

- code file structure, 242
- Compile property page and, 25
- in data type characters, 264
- in variable declarations, 286–288, 926

Option Infer

- in code file structure, 242
- Compile property page and, 26
- in variable declarations, 281, 926–927

Option statements, 242

Option Strict

- assignment operators and, 315
- in code file structure, 242
- Compile property page and, 25–26

- in narrowing conversions, 267
- in variable declarations, 286–288, 926

optional parameters, 338–339

Options command, 30, 32–33

Options dialog box, Windows Form Designer, 33, 49–51

Or operator, 311, 313

OracleCommand class, 424–425

OracleConnection object, 414

Order By clause, LINQ, 462, 1068, 1076

Order command, 29

OrElse operator, 311–312

orphan lines/words, 762

outlining code, 20, 74–75

OutOfMemoryException, 384

Output command, Debug menu, 92

overloading

- methods, 339
- in OOP languages, 559–560
- operators, 321–324, 936–937

Overloads keyword, subroutines, 332

Overridable keyword, subroutines, 332

Overrides keyword, subroutines, 332

P

packages, XPS document, 221

Page objects, WPF

- browser applications, 226–227
- defined, 225
- hosting in Frame control, 227–229
- PageFunction applications, 229–231
- wizard applications, 231–235

Page Setup command, 18

PageBounds parameter, PrintDocument, 766

PageContent objects, FixedDocument control, 221

PageFunction objects, WPF, 229–231, 233–235

PageSetupDialog component, 106, 994–995

Paint event, 693–695

PaintEventArgs parameter, 694

Panel control

- containing and arranging with, 110–111
- defined, 106
- overview of, 995–996
- SplitContainer control containing, 1013–1015
- WPF controls, 178

ParagraphInfo structures, printing text, 768–769, 772

parameter arrays, 339–341

parameters

- declaring, 292–294

- optional, 338–339
- parenthesized, 314, 337
- of subroutines. See subroutine parameters
- parameters clause, 592**
- Parameters collection, 424–425**
- parent features, derived controls, 504–505**
- parenthesized parameters, 314, 337**
- ParentsizePropertyName attribute, 330**
- Parse method, data types, 269**
- Partial keyword**
 - in class declarations, 564–565
 - with code regions, 243
 - with Imports statement, 607
 - with partial methods, 350–351
- partial methods, 350–351, 941**
- parts, XPS document package, 221**
- PasswordBox control, WPF, 180**
- passwords**
 - database, 405
 - encryption, 894
- Paste Button Image command, 45**
- Path class, 866–868, 1152–1153**
- Path control, WPF, 183**
- Path object, XAML, 210–213**
- PathGradientBrush class, 706, 714–719**
- patterns, and comparison operators, 310–311**
- Pen object, 697–705**
 - Alignment property, 700
 - CompoundArray property, 701
 - constructors, 698
 - CustomLineCap object, 702–703
 - defined, 697
 - how lines are drawn, 662
 - properties and methods, 698–699, 1088
 - transformations, 703–705
- Performance Tools, 30, 33**
- PerformanceCounter component, 106, 996**
- permissions, 849, 880**
- Person objects, 625**
- PictureBox control, 107, 116, 996–998**
- pixel-by-pixel operations, 753–757**
- PLATFORM constant, 247**
- Pointer tool, 104, 955**
- PointToScreen method, 720–721**
- Polygon control, WPF, 183**
- Polygon object, XAML, 209**
- Polyline control, WPF, 183**
- Polyline object, XAML, 209**
- polymorphism, 558–559**
- pop, adding items to stack with, 640**
- PopUp control, WPF, 181**
- Port object, My.Computer, 1123–1125**
- position properties, of controls, 137–138**
- PositionChanged event, CurrencyManager, 449–450**
- power packs, 1043–1046**
- Power Toys Pack Installer, 1046**
- precedence, operator, 313–314, 934–935**
- predefined constants, 246–247**
- Prepare method, command objects, 425**
- Preserve keyword, resizing arrays, 623**
- Preview Data command, 28**
- Previous Method command, 20**
- Print commands, 18**
- PrintDialog control**
 - basic printing with, 764–766
 - defined, 107
 - overview of, 998–1000
 - WPF, 182
- PrintDocument control**
 - basic printing with, 762–766
 - defined, 107, 116–117
 - overview of, 1000–1001
 - printing text, 767–772
 - System.Drawing.Printing namespace, 672–673
- Printer Compatibility Library, 1044**
- PrintForm component, 1044–1045**
- printing, 761–780**
 - basic, 762–766
 - centering printouts, 773–775
 - fitting pictures to page, 775–777
 - how not to do, 761–762
 - list of controls for, 954
 - simplifying drawing and, 777–780
 - System.Drawing.Printing namespace and, 672–673
 - text, 766–772
- PrintPage event, PrintDocument, 763, 766, 769–772**
- PrintPreviewControl control, 107, 116, 1001–1002**
- PrintPreviewDialog control, 107, 116, 1003–1004**
- Private keyword**
 - class declarations with, 566
 - constant declarations with, 301
 - event declarations with, 592
 - subroutine's accessibility clause with, 333
 - variable declarations with, 261, 271, 275
 - variables with module scope, 290–291
- privileges. See UAC (User Account Control) security procedure scope, 290**
- procedures, 327**
- Process component, 107, 1004**
- Processes window, Debug menu, 94**
- Professional Crystal Reports for Visual Studio .NET (Wiley), 781**

program and module structure, 237–260

- code files, 242
- code regions, 243
- comments, 253–254
- conditional compilations, 243–244
- CONFIG constant, 250
- DEBUG constant, 247–250
- debugging level constants, 251–252
- hidden files, 237–241
- line continuation characters, 258–259
- line joining, 259
- line labels, 259
- namespaces, 252–253
- predefined constants, 246–247
- setting constants in code, 245
- setting constants with project's compilation settings, 245–246
- TRACE constant, 250
- XML comments, 254–258

program control statements

- decision statements. See decision statements
- defined, 353
- GoTo statement, 372–374
- looping statements. See looping statements

ProgressBar control

- defined, 107, 115
- summary of, 1005
- WPF, 181

prohibited generics, 1079

Project menu, 21–26

projects

- creating, 9–11
- defined, 237
- overview of, 6–7
- saving, 12–13
- setting constants with compilation settings, 245–246

Projects and Solutions, Options dialog box, 33

properties

- Application class, 830–831, 1109–1110
- ArrayList class, 627–629
- assigning attributes to controls, 499–501
- binding to data source, 446–447
- classes defining, 547–548
- connection objects, 415–416
- Control class, 911–914
- CurrencyManager object, 448–450
- data adapters, 420
- DataColumn class, 435–436
- DataRelation class, 438–439
- DataRow class, 433
- DataRowView, 445
- DataSet object, 428
- DataTable class, 430–431

- DataView object, 444–445
- Debug object, 249
- declaring within classes, 589–590
- defining project, 240
- dictionary classes, 633
- DirectoryInfo object, 861–862
- DriveInfo object, 860
- dynamic, 816–819
- editing Visual Basic code, 195–196
- Environment object, 809–810
- Err object, 397
- FileInfo object, 862–864
- FileStream class, 840
- FileSystemWatcher class, 864–865
- ForeignKeyConstraint, 440
- Form class, 1047–1053
- Graphics object, 679–682
- GraphicsPath object, 722–723
- Image class, 746–747
- LinearGradientBrush class, 709–710
- modifying component, 509–510
- My.Computer.FileSystem, 868
- My.Computer.Registry, 813
- MySettings namespace, 820
- NameValueCollection class, 632
- Path class, 867
- PathGradientBrush class, 714–715
- Pen object, 698–701
- queues, 642
- RegistryKey objects, 814–816
- special directory, 871
- stacks, 641
- Stream class, 838
- streams, 1131–1137
- strongly typed collections, 630
- text formatting, 730
- TextureBrush class, 706–707
- TextWriter class, 844
- UniqueConstraint, 441
- Windows Forms, 158–159

properties, Windows Forms controls/components

- CheckBox control, 956–957
- CheckBoxList control, 957–958
- ColorDialog component, 958–959
- ComboBox control, 959–961
- custom controls, 498
- DataGridView control, 961–962
- DateTimePicker control, 962–964
- at design time, 127–132
- DomainUpDown control, 964–965
- ErrorProvider component, 965–966
- FlowLayoutPanel control, 967
- FolderBrowserDialog component, 967–968
- FontDialog component, 968–969

GroupBox control, 970
 HelpProvider component, 970–971
 HScrollBar control, 971–972
 ImageList control, 972
 Label control, 973
 LinkLabel control, 974
 ListBox control, 975–976
 ListView control, 981–982
 MaskedTextBox control, 982–984
 MenuStrip control, 985–986
 MonthCalendar control, 987–988
 NotifyIcon component, 991
 NumericUpDown control, 991–992
 OpenFileDialog component, 992–993
 overview, 126–127
 PageSetUpDialog component, 995
 Panel control, 995–996
 PictureBox control, 997
 position and size, 137–138
 PrintDialog component, 998–1000
 PrintDocument component, 1000–1001
 PrintPreviewControl control, 1002
 Process component, 1004
 ProgressBar control, 1005
 PropertyGrid control, 1005
 RadioButton control, 1006–1007
 RichTextBox control, 1008–1011
 at runtime, 132–133
 SplitContainer control, 1014
 TabControl control, 1016–1018
 TabLayoutPanel control, 1019–1021
 TextBox control, 1021–1022
 Timer component, 1023
 ToolStrip control, 1024
 ToolTip component, 1026
 TrackBar control, 1027–1028
 TreeView control, 1029–1032
 useful, 133–137
 WebBrowser control, 1033
Properties window, 34
 changing control location in, 28
 changing properties at design time, 11
 complex data binding in, 452–453
 displaying component properties, 508–509
 displaying control events in, 58
 as dockable, 34
 implementing custom controls in, 498–501, 507
 overview of, 38–41
 setting properties in, 54–56
 setting Toolbox icon in, 495
 simple data binding in, 446
 using command verbs in, 56–57
 viewing attribute properties in, 329
 in Windows Form Designer, 61–62, 67–68

in Windows Forms controls, 123, 127–130, 139
 Windows Forms icons, 157
 WPF controls, editing XAML, 190–193
Property Pages command, 21
property procedures
 functions, 344
 overview of, 294–295
 syntax for, 940
PropertyGrid control, 107, 113–114, 1005–1006
Protected Friend keyword
 class declaration with, 566
 constant declaration with, 290–291
 event declaration with, 592
 subroutine's accessibility clause, 333
 variable declaration accessibility clause, 271
 variables with module scope, 290–291
Protected keyword
 class declaration with, 565
 constant declaration with, 290–291
 event declaration with, 592
 subroutine's accessibility clause, 333
 variable declaration accessibility clause, 271
 variables with module scope, 290–291
public interface, classes, 549
Public keyword
 class declaration with, 565
 constant declaration with, 301
 event declaration with, 591
 subroutine's accessibility clause, 333
 variable declaration accessibility clause, 271
 variable declaration with, 261
public-key encryption, 893–897
push, adding items to stack, 640

Q

Query Builder, 421, 423–424
query expressions, 458
QueryPageSettings event, 763, 769
queues, collection classes, 641–642
Quick Find command, 18
Quick Replace command, 18–19
QuickWatch command, 85, 88–89
QuoteClient, 881–882
QuoteService, 878–881

R

RadioButton control
 defined, 107
 overview of, 1006–1007

RadioButton control (continued)

within Panel control, 110
as selection control, 112
WPF, 179

Raise method, Err object, 398

RaiseEvent keyword, 593, 596–599

random-access file, 853–855

ReadOnly attribute, 330

ReadOnly keyword, 274–275, 300

read-only properties, 133, 940

read-only variables, 295

ReadOnlyCollectionBase class, 631

Rearrange Commands button, 47

Recent Files and Recent Projects submenu, 18

RecommendedAsConfigurable attribute, 330

Rectangle control, WPF, 183

Rectangle object, XAML, 209

ReDim statement, arrays, 280, 623

reference types, classes as, 574

refinement, and inheritance, 552–555

reflection, 897–901

RefreshProperties attribute, 331

Region statement, 243

regions

code, 243
defining with GraphicsPath objects, 720

Registry

accessing with My.Computer.Registry, 814–816
accessing with native Visual Basic methods,
812–813
overview of, 811–812

Registry object, My.Computer, 1125–1126

RegistryKey objects, 814–816

regular expressions, 888–891

relaxed delegates, 347–350, 594–595, 956

RemoveHandler statement

declaring custom events, 596–599
removing event handler with, 594
setting event handlers at runtime, 141–142

Rename Tab command, 38

RenderTransform element, XAML, 205, 206–207

RepeatButton control, WPF, 182

ReportDocument object, 799–801

reports, 781–801

building. See Standard Report Creation Wizard
CrystalReportDocument control, 799–801
CrystalReportViewer control, 794–795
customizing, 795–799
external, 799
list of components, 954
MicrosoftReportViewer control, 107
report objects, 781–782

Reset Button Image command, 45

Reset methods, form properties, 159

Reset Toolbox command, 38

Reset Window Layout command, 35

resize events, controls, 921

resizing

arrays, 623
auto-redraw and, 752
controls. See sizing controls

resource files, 821–829

application resources, 821–822
ComponentResourceManager, 827–829
embedded resources, 823–824
localization resources, 826–827
overview of, 821
satellite resources, 824–826
using application resources, 822–823

resources

defining in XAML, 199–201
overview of, 238–239

Resources tab, setting project, 240

Resources.resx file, 237

Restore method, graphics state, 692–693

restricted properties, 129

restricted scope, 291–292

results, LINQ query

limiting, 471–472, 1069–1070
using, 465, 1071

Resume Next statement, classic error handling, 392–393

Resume statement, classic error handling, 392

Return statement, functions, 940

Rich Text Format (RTF) files, 1007

RichTextBox control

defined, 107
entering data, 113
overview of, 1007–1012
WPF, 180

root namespace, 610–612

RotateTransform, XAML, 205, 206–207

RotateTransform method, 685–688, 692–693

routed events, 207

routines, 327

.rpt extension, 781

RTF (Rich Text Format) files, 1007

Run Tests command, 34

runtime

creating controls at, 124–126
invoking methods at, 139
making controls invisible at, 510–511
managing controls running at, 501
modifying control properties at, 132–133
setting control's Anchor property at, 135
setting event handlers at, 141–142
Visual Basic code editor at, 71–72, 85–86
Visual Basic code editor tooltips at, 75–76

S

- satellite resources, 824–826**
- Save All command, 18**
- Save Form1.vb As command, 18**
- Save Form1.vb command, 18**
- Save method, graphics state, 692–693**
- Save Project dialog, 12–13**
- SaveFileDialog, 107, 1012**
- SaveSetting method, Registry values, 812–813**
- SaveSettings command, 820**
- saving**
 - Bitmaps, 748–750
 - configuration information, 820
 - projects, 12–13
- ScaleTransform, XAML, 205**
- ScaleTransform, XAML, 776**
- ScaleTransform method, 685–688**
- Schema Editor, 407**
- scope**
 - block, 288–289
 - determining variable behavior, 261
 - module, 290
 - namespace, 291
 - procedure, 290
 - restricting, 291–292
 - and visibility, 262
- Screen object, My . Computer, 1126–1128**
- ScrollBar control, WPF, 179**
- ScrollViewer control, WPF, 178**
- SDI (single-document interface), 161, 164, 167**
- search-and-replace features, 243**
- secret-key encryption, 893–897**
- sections, 240–241, 807**
- security**
 - custom component, 514–517
 - UAC. See UAC (User Account Control) security
- Select Active Test Run Configuration command, 34**
- Select Case statement**
 - block scope, 288–289
 - Choose statement vs., 360–362
 - comma-separated expressions, 357–358
 - decision statements, 944–945
 - enumerated values, 358
 - To keyword, 356–357
 - syntax for, 355–356
- Select keyword, LINQ, 462–465, 1068, 1076**
- selection controls**
 - Windows Forms controls, 111–112
 - Windows Presentation Foundation, 179–180
- self-hosting, 882–883**
- separated validation, 145–146**
- Separator control, WPF, 178**
- sequential-file access, 853**
- Serializable attribute, 331**
- serializable objects, dragging, 526–528**
- SerialPort component, 107, 1013**
- service contract, WCF, 874**
- ServiceController component, 107, 1013**
- service-oriented architecture (SOA), 873**
- services, hosting, 882–883**
- set operations, LINQ, 471**
- SetAllCursors subroutine, 155**
- SetBlendTriangularShape property, HatchBrush, 714**
- SetClip method, Graphics object, 721–722**
- SetData method, 529–532**
- SetDataObject method, clipboard, 532**
- SetEnvironmentVariable method, 811**
- SetError method, ErrorProvider, 965**
- SetMeasurableCharacterRanges method, Graphics object, 737–739**
- SetPixel method, Bitmap class, 748, 753–754, 757**
- SetSigmaBellShape property, HatchBrush, 714**
- settings, defined, 238**
- Settings tab, 240**
- Settings.setting file, 237**
- shadowing parent features, derived controls, 504–505**
- Shadows keyword**
 - class declarations with, 566–567
 - event declarations with, 592
 - subroutines, 333
 - variable declarations with, 272–274
- shapes, Microsoft Power Packs, 1045**
- Shared keyword**
 - constant declarations, 300–301
 - shared methods, 600–602
 - shared subroutines, 332
 - shared variables, 599–600
 - variable declarations, 272
- shortcut keys**
 - commands, 15–16
 - My namespace, 886
- Show All command, 38**
- Show All Files command, 22**
- Show Data Sources command, 28**
- Show Hidden Files button, 237–238**
- ShowDialog method, 170**
- signature authorities, 516–517**
- signing, 893–897**
- single-document interface (SDI), 161, 164, 167**
- single-line If Then statements, 288–289, 353–354, 943–944**
- size properties, controls, 137–138**
- Size property, 128**
- sizing arrays, 623**

sizing controls

- icons in Windows Forms, 156–157
- layout controls, 179
- Windows Form Designer, 53
- WPF Designer, 67
- skip keyword, LINQ, 471–472**
- Skip While keyword, LINQ, 471–472**
- slider control, WPF, 179**
- smart tags, 50, 56**
- SmoothingMode property, 683, 720–721**
- snap lines, WPF, 66**
- snap-to-grid mode, WPF Designer, 62**
- snippets, code, 81–85**
- SOA (service-oriented architecture), 873**
- SolidBrush class, 706**
- Solution Explorer, 11**
- solutions, 6–7, 13**
- Sort Items Alphabetically command, 38**
- SortedList class, 639**
- Sorting property, ListView control, 981**
- Source Control, Options dialog box, 33**
- Source value, Exception class, 386**
- spaghetti code, 373**
- SpecialDirectoriesProxy object, 870**
- speed, anti-aliasing settings, 684**
- splash screens, Windows Forms, 153–154**
- Split command, 34**
- SplitContainer control, 107, 111, 1013–1015**
- Splitter control, 108, 1015**
- Spy++ tool, 921**
- SQL, LINQ to, 487–488**
- SQL statements, Data Adapter Configuration Wizard, 421–422**
- SqlCommand class, 424–425**
- SqlConnection object, 414–415**
- StackPanel control, WPF, 178**
- stacks**
 - collection classes, 640–641
 - performance, 574–575
- StackTrace method, Exception class, 386**
- StackTrace object, 386–387**
- standard format specifiers**
 - date and time, 1099–1100
 - numeric, 1103–1104
- Standard Report Creation Wizard, 783–794**
 - Fields page, 785–786
 - Group Sorting page, 788–789
 - Grouping page, 786–787
 - Link page, 784–785, 786
 - Record Selection page, 789, 791
 - Report Style page, 790, 792
 - Summaries page, 787–788
- standard windows commands, 20**
- Start Page, Visual Studio, 7–11**
- Start Wizard button, 231–233**
- StateChange event, connection object, 417**
- statements section, subroutines, 341**
- Static keyword, 271, 300**
- status area (system tray), 157–158**
- StatusBar control, WPF, 181**
- StatusBarItem control, WPF, 181**
- StatusStrip control, 108, 115, 1015–1016**
- StreamWriter class, 1136**
- Step Into command, Debug menu, 88**
- Step Out command, Debug menu, 88**
- Step Over command, Debug menu, 88**
- Stop Debugging command, Debug menu, 88**
- stop statement, and debugging, 398**
- storage, variable requirements, 263**
- Storyboards, XAML, 206–208**
- Stream class, 838–839**
- StreamReader class, 845–846, 1136**
- streams, 837–848, 1131–1137**
 - AppendText method, 846–848
 - BinaryReader and BinaryWriter, 841–843, 1133–1135
 - BufferedStream class, 841
 - console streams, 1136
 - CreateText method, 846–848
 - custom classes, 848
 - FileStream class, 839–840
 - MemoryStream class, 840–841
 - OpenText method, 846–848
 - overview of, 837–838, 1131–1132
 - Stream, 1132–1133
 - Stream class, 838–839
 - StreamReader and StreamWriter, 845–846, 1136
 - StringReader and StringWriter, 1136
 - text file methods, 844
 - TextReader and TextWriter, 843–844, 1135–1136
- StreamWriter class, 845–846**
- StringBuilder class, 316–318, 844–845**
- StringCollection class, 629**
- StringDictionary class, 639**
- StringFormat object**
 - FormatFlags property, 731–734
 - GetTabStops method, 734–735
 - properties and methods, 730, 1090–1091
 - SetTabStops method, 734–735
 - text formatting using, 728–729
 - Trimming property, 735–736
- StringReader class, 838, 844–845, 1136**
- StringWriter class, 838, 844–845**
- StringWriter class, 1136**
- stroke attribute, XAML, 209**
- stroke property, 183**

StrokeDashArray attribute, XAML, 210
StrokeEndLineCap attribute, XAML, 209
StrokeLineJoin attribute, XAML, 210
StrokeStartLineCap attribute, XAML, 209
StrokeThickness attribute, XAML, 209
StrokeThickness property, 183
strongly named assemblies, 514–516
strongly typed collections, 629–631, 643–645
strongly typed dictionaries, 636–638
structured error handling, 383–391

- building custom exceptions, 389–391
- classic vs., 396–397
- Exception objects, 385–386
- overview of, 383–385
- StackTrace objects, 386–387
- summary of, 949
- throwing exceptions, 387–389

structures

- boxing and unboxing, 580
- classes vs., 574–575
- constructors, 1065
- declaring, 573
- graphics namespaces, 1081–1084
- heap and stack performance, 577
- information about, 1064
- inheritance and, 574
- instantiation details, 582–584
- memory requirements, 575–576
- namespaces and, 613–614
- object assignment, 577–578
- overview of, 573–574
- parameter passing, 578–579
- System.Drawing namespace, 665
- as value types, 574–575

styles, ComboBox control, 959–960
styles, XAML, 201–202, 204
Sub Main procedure, 902
Sub statement, 327
subroutine parameters, 334–341

- arrays declared ByVal and ByRef, 335–337
- ByRef keyword, 335
- ByVal keyword, 334–335
- optional parameters, 338–339
- optional vs. overloading, 339
- parameter arrays, 339–341
- parenthesized parameters, 337
- passing, 578

subroutines, 327–342

- accessibility clause for, 333
- attribute list, 328–332
- declaring variables using Dim in, 275
- defined, 327
- defining code regions, 243
- editing Visual Basic code, 196

- Implements interface.subroutine, 341–342
- inheritance_mode, 332–333
- instantiating classes/structures using constructors, 581–584
- names, 334
- passing delegate variables into, 303
- property procedures for, 940
- statements in, 341
- syntax for, 939

subtrees (hives), Registry, 811–812
Summaries page, Standard Report Creation Wizard, 787–788
SurroundColors property, PathGradientBrush class, 715, 718
SurroundPoints property, PathGradientBrush class, 715, 719
System namespace

- overview of, 606
- providing integer data types, 263
- second-level namespaces within, 887–888

system tray (status area), 157–158
System.Drawing namespace, 663–665, 1081–1082
System.Drawing.Drawing2D namespace, 665–668, 1082–1083
System.Drawing.Imaging namespace, 668–669, 1083–1084
System.Drawing.Printing namespace, 672–673, 1084
System.Environment object, 809–811
System.Security namespace, 893–897
System.Windows.Controls namespace, WPF controls, 187, 1035
System.Windows.Forms namespace, 187
System.Xml namespace, 891–893

T

Tab Order command, 20
TabControl control

- defined, 108, 111
- MDI container for, 167
- useful properties, 1016–1019
- WPF, 178

TabItem control, WPF, 178
TableAdapter object, 411, 413–414, 452
TableAdapterManager, 411
TableLayoutPanel control, 108, 110, 1019–1021
TableMappings property, data adapters, 421
tabs

- Error List window, 10–11
- TabControl control, 1016–1019
- ToolBox, 953–954

Take keyword, LINQ, 471–472

Take While keyword, LINQ, 471–472

TARGET constant, 247

TargetSite method, Exception class, 386
templates

Windows Application, 497

Windows Control Library, 494

XAML, 202–205

Test menu, 33–34

testing

catching bugs, 379

controls, 498

custom controls, 495–497

QuoteService, 880–881

unplanned conditions not caught in, 380

text, 727–743

drawing, 728

font metrics, 739–743

MeasureString method, 736–739

overview of, 727

WPF controls for, 180

Text Editor, Options dialog box, 33

text formatting

FormatFlags property, 731–734

overview of, 728–731

tab stops, 734–735

Trimming property, 735–736

Text Only (Always) command, 46

Text property, groups of controls, 55

TextBlock control, WPF, 181

TextBox control

in complex data binding, 453

defined, 108

entering data, 113

for login forms, 154

overview of, 1021–1023

WPF, 180

TextReader class, 843–844, 1135–1136

TextRenderingHint, 670–671, 683, 727

TextureBrush class, 706–708

TextWriter class, 843–844, 1135–1136

thin user interface applications, 188

third-party controls, obtaining, 118–119

ThreadException events, 833–835

ThreadExit events, 833–834

threads, defined, 92

Threads window, Debug menu, 92–93

ThreadStaticAttribute, 331

**ThreeD Computer Graphics, Third Edition
(Watt), 908**

Throw statement, 949–950

throwing events, 139

throwing exceptions, 387–389, 949–950

Timer component

creating splash screens using, 153–154

defined, 108

initiating action, 116

overview of, 1023

TimeSpan operator, 318–320, 935–936

To keyword, Select Case statement, 356

TodayDate property, MonthCalendar control, 988

Toggle Bookmark tool, 73–74

Toggle Breakpoint command, Debug menu, 90

ToList function, LINQ, 474, 489–490

ToolBar control, WPF, 182

toolbars

containing menu commands, 16

customizing appearance of, 4

customizing commands on, 43–47

IDE after starting new project, 10

working with, 35–36

Toolbars submenu, 20

ToolBarTray control, WPF, 182

Toolbox

adding control to, 497

adding control to form from, 124

adding WPF control to, 176

adding WPF Designer controls to, 64

features of, 37–38

setting icon for controls, 495

starting new project, 10–11

tools, 953–954

ToolboxBitmap attribute, 331, 495

ToolBoxBitmap attribute, 500–501

Tools menu, 34–35

commands, 29–30

Macros submenu, 30–32

menus, 29–33

Options submenu, 32–33

Tools.Alias command, Debug menu, 98

ToolStrip control, 108, 116, 1023–1024

ToolStripContainer control, 108–109, 1025

ToolStrip control

defined, 108

overview of, 1026–1027

providing feedback to user, 114, 181

tooltips, Visual Basic code editor, 75–76, 80

ToString method, 298, 386

TRACE constant

catching bugs using, 379

defined, 247

working with, 250

TrackBar control

defined, 108

initiating action, 116

overview of, 1027–1028

as selection control, 112

Transaction property, command objects, 425

transactions, 417–419

transformations, 684–691

- advanced, 688–691
- code, 685–688
- mathematics, 684–685
- Pen object, 703–705
- saving and restoring graphics state, 692–693
- XAML, 205

TransformGroup element, XAML, 205, 207

TranslateTransform method, Graphics object, 685–688, 773

transparency, Windows Forms, 150–153

TransparencyKey property, Windows Forms, 151–153

TreeNode object, 1031–1032

TreeView control

- defined, 108
- displaying data, 113
- displaying data, WPF, 181
- overview of, 1028–1032

Trigger element, XAML, 203, 206–207

Trimming property, StringFormat object, 735–736

Try block, structured error handling, 383–384

Try Catch statements, block scope, 288–289

Try statements, error handling, 383–384, 396–397

TryCast statement, 269

TypeOf.. operator, 309–310

U

UAC (User Account Control) security, 537–544

- designing, 538–541
- elevating programs, 542–543
- new in Windows Vista, 4
- overview of, 537–538

unboxing, 580

Uncomment Selection command, 254

UnhandledException event handler, 382, 383

Unindent methods, Debug object, 249

UniqueConstraint, 439, 441–442

UnlockBitmap method, Bitmap object, 756

unplanned conditions, 378, 380–381

Until keyword, Do Loop, 370

updates, table adapter, 411, 419

User Account Control. See UAC (User Account Control) security

user interface, WPF, 188–189, 213–218

UserControl Test Container, 495–496, 498–499

users, elevated privileges, 542

UseWaitCursor property, Application object, 831, 833

Using statement

- block scope, 288–290
- garbage collection using, 725
- variable declarations, 924

V

valid identifiers, 278

Validated event, 145–146

Validating event, 143–147

validation events, 143–147

value types, structures as, 574–575

variable declarations, 270–288

- accessibility clause in, 271–272
- attribute list in, 270–271
- bounds_list clause in, 279–280
- With clause, 924
- As clause in, 280–281
- Dim keyword in, 275
- enumerated types, 924–925
- inferred types in, 280–281
- information about, 923–924
- initialization, 281–285
- initialization expressions, 924
- multiple, 285–286
- name clause, 278
- New keyword in, 280
- Option Explicit and Option Strict, 286–288, 926
- Option Infer, 926–927
- ReadOnly keyword in, 274–275
- Shadows keyword in, 272–274
- Shared keyword in, 272
- syntax for, 270
- Using statement, 924
- WithEvents keyword in, 276–277
- XML variables, 925

variables, 261–305

- anonymous types, 298–299
- constants, 300–301
- data type characters, 264–266
- data type conversion, 266–269
- data types, 262–263
- delegate, 302–303
- enumerated data types, 295–298
- environment, 807–811
- naming conventions, 303–304

variables (continued)

- nullable types, 299–300
 - overview of, 261–262
 - parameter declarations, 292–294
 - property procedures, 294–295
 - scope, 288–292
 - shared, 599–600
 - Visual Basic code editor tooltips, 75–76
- VBC_VER constant, 247**
- versions, Visual Studio, 5**
- Vertical Spacing command, 29**
- View menu, 20–21**
- Viewbox control, WPF, 178**
- VirtualizingStackPanel control, WPF, 178**
- visibility, 262**
- Visible property, 129, 510–511**
- Visual Basic**
- building WPF interface with, 213–218
 - development settings, 5–6
 - editing code, 194–197
- Visual Basic Classic Error Handling. See classic error handling**
- Visual Basic code editor, 71–86**
- code coloring and highlighting, 79–81
 - code snippets, 81–85
 - IntelliSense, 77–79
 - margin icons, 72–74
 - other Visual Studio editors vs., 71
 - outlining, 74–75
 - overview of, 71–72
 - at runtime, 85–86
 - tooltips, 75–76
- Visual Basic Power Packs, 1043–1046**
- Visual Studio**
- adding code to WPF Designer controls, 68
 - free Express editions, 5
 - icon editor, 157
- Visual Studio Tools for Office (VSTO), 762**
- VScrollBar control**
- defined, 108
 - initiating action, 116
 - overview of, 1033
 - as selection control, 112
- VSTO (Visual Studio Tools for Office), 762**

W

- WaitForExit method, UAC, 540**
- Watch window, Debug menu, 89–90, 94**
- WCF (Windows Communication Foundation), 873–884**
- building initial service, 875–878
 - building QuoteClient, 881–882
 - building QuoteService, 878–880
 - concepts, 874
 - early version warning, 874
 - example of, 875
 - hosting service, 882–883
 - overview of, 873
 - testing QuoteService, 880–881
- WCF Service Host, 880–881**
- WCF Test Client, 880–881**
- WebBrowser control, 108, 1033**
- WFD (Windows Form Designer), 49–59**
- adding code to controls, 57–59
 - adding controls, 51
 - arranging controls, 54
 - copying controls, 53
 - moving and sizing controls, 54
 - selecting controls, 51–53
 - setting options, 49–51
 - setting properties, 53–56
 - using command verbs, 56–57
 - using smart tags, 56
- Where clause, LINQ syntax, 461–462, 466, 1067–1068, 1076**
- While End loop, 371**
- While End loop, 948**
- While keyword, Do Loop, 370**
- While loops, 288–289, 362**
- widening conversions, data types, 269**
- widow lines/words, 762**
- Window class, WPF, 189, 223–225**
- Window menu, 34–35**
- windows, 36–41**
- docking, 36–37
 - Properties window, 38–41
 - recognizing WPF Designer, 63
 - Toolbox window, 37–38
- Windows command, 34, 35**
- Windows Communication Foundation. See WCF (Windows Communication Foundation)**
- Windows Form Designer. See WFD (Windows Form Designer)**
- Windows Forms, 149–173**
- About forms, 153–154
 - dialog boxes, 170–171
 - icons, 156–158
 - IDE in WPF vs., 190–191
 - login forms, 153–154
 - MDI events, 165–166
 - MDI features, 162–165
 - MDI vs. SDI, 167
 - Most Recently Used lists, 168–170
 - mouse cursors, 154–156
 - overriding WndProc, 159–161
 - properties adopted by child controls, 158–159

- property reset methods, 159
- SID and MDI, 161–162
- splash screens, 153–154
- transparency, 150–153
- wizards, 172–173
- WPF applications vs., 187–188
- Windows Forms controls and components**
 - adding to containers, 124
 - BackgroundWorker component, 955
 - BindingNavigator control, 955
 - BindingSource control, 955
 - Button control, 956
 - CheckBox control, 956–957
 - CheckedListBox control, 957–958
 - choosing restrictive, 109
 - ColorDialog component, 958–959
 - ComboBox control, 959–961
 - containing and arranging, 109–110
 - ContextMenuStrip component, 961
 - controls and components, 121–123
 - controls vs. components, 121–123, 494
 - creating at design time, 123–124
 - creating at runtime, 124–126
 - DataGridView control, 961–962
 - DataSet component, 962
 - DateTimePicker control, 962–964
 - DirectoryEntry component, 964
 - DirectorySearcher component, 964
 - displaying data, 113–114
 - displaying dialog boxes, 117
 - displaying graphics, 116
 - DomainUpDown control, 964–965
 - entering data, 113
 - ErrorProvider component, 965–966
 - event handlers, creating at design time, 139–140
 - event handlers, setting at runtime, 141–142
 - event handlers, WithEvents, 141
 - EventLog component, 966
 - events, control array, 142–143
 - events, overview of, 139
 - events, validation, 143–147
 - FileSystemWatcher component, 966
 - FlowLayoutPanel control, 967
 - FolderBrowserDialog component, 967–968
 - FontDialog component, 968–970
 - GroupBox control, 970
 - HelpProvider component, 970–971
 - HScrollBar control, 971–972
 - ImageList component, 972
 - initiating action, 115–116
 - Label control, 973
 - LinkLabel control, 973–974
 - list of most useful, 954
 - ListBox control, 975–976
 - ListView control, 976–979
 - ListView control, custom sorting, 981–982
 - ListView control, helper code, 979–981
 - MaskedTextBox control, 982–984
 - MenuStrip control, 985–986
 - MessageQueue component, 987
 - methods, 138–139
 - MonthCalendar control, 987–990
 - NotifyIcon component, 990–991
 - NumericUpDown control, 991–992
 - OpenFileDialog component, 992–994
 - overview of, 103–109, 951–953
 - PageSetupDialog component, 994–995
 - Panel control, 995–996
 - PerformanceCounter component, 996
 - PictureBox control, 996–998
 - pointer, 954
 - PrintDialog component, 998–1000
 - PrintDocument component, 1000–1001
 - PrintPreviewControl control, 1001–1002
 - PrintPreviewDialog component, 1003–1004
 - Process component, 1004
 - ProgressBar control, 1005
 - properties, at design time, 127–132
 - properties, at runtime, 132–133
 - properties, overview of, 126–127
 - properties, position and size, 137–138
 - properties, useful, 133–137
 - PropertyGrid control, 1005–1006
 - providing feedback, 114–115
 - purposes of components, 953–954
 - RadioButton control, 1006–1007
 - ReportViewer control, 1007
 - RichTextBox control, 1007–1012
 - SaveFileDialog component, 1012
 - selection controls, 111–112
 - SerialPort component, 1013
 - ServiceController component, 1013
 - SplitContainer control, 1013–1015
 - Splitter control, 1015
 - StatusStrip control, 1015–1016
 - supporting other controls, 117
 - TabControl control, 1016–1019
 - TableLayoutPanel control, 1019–1021
 - TextBox control, 1021–1023
 - third-party controls, 118–119
 - Timer component, 1023
 - ToolStrip control, 1023–1024
 - ToolStripContainer control, 1025
 - ToolTip component, 1026–1027
 - TrackBar control, 1027–1028
 - TreeView control, 1028–1032
 - VScrollBar control, 1033
 - WebBrowser control, 1033

Windows Presentation Foundation controls. See **WPF (Windows Presentation Foundation) controls**

Windows Presentation Foundation Designer. See **WPF (Windows Presentation Foundation) Designer**

Windows submenu, Debug menu, 88, 91–95

Windows Vista

- Aero appearance in, 4
- graphics API in, 727
- setting environment variables in, 808
- underlying graphics model, 662
- User Account Control in, 4

Windows XP, 808

WindowsApplication1 Properties command, 24

WinZip, 118

With clause, 582, 924

WithEvents keyword

- catching events with, 125, 593–595
- variable declarations using, 276–277
- WithEvents event handlers, 141

wizard applications, WPF, 231–235

WizardData class, 233–235

wizards, Windows Forms, 172–173

WndProc, 159–161

WorkItem property, controls, 498–499

WPF (Windows Presentation Foundation) controls, 175–222

- containing and arranging, 176–179
- digital ink controls, 184–185
- displaying data, 180–181
- documents, 218
- drawing objects, 208–213
- editing Visual Basic code, 194–197
- editing XAML, 190–194
- entering data, 180
- hierarchies, 189
- initiating action, 182
- list of, 1035–1042
- making selections, 179–180
- managing documents, 184
- overview of, 175–176
- presenting graphics and media, 182–183
- procedural WPF, 213–218
- providing navigation, 184
- separation of user interface and code, 188–189
- Windows Forms applications vs., 187–188
- XAML animations, 206–208
- XAML objects, 197–199
- XAML resources, 199–201
- XAML styles, 201–202
- XAML templates, 202–205
- XAML transformations, 205

WPF (Windows Presentation Foundation) Designer, 61–69

- adding code to controls, 68–69
- adding controls, 63
- copying controls, 65
- early version warning, 61–62
- moving and sizing controls, 66–67
- recognizing windows, 63
- selecting controls, 63
- setting group properties, 67
- setting properties, 67

WPF (Windows Presentation Foundation) windows, 223–236

- browser applications, 226–227
- frame applications, 227–229
- overview of, 223
- Page applications, 225
- PageFunction applications, 229–231
- Window applications, 223–225
- wizard applications, 231–235

WrapPanel control, WPF, 178

Write methods, Debug object, 250

write-only property procedure, syntax for, 940

write-only variables, 295

WYSIWYG (what you See is what you get) design surface, 49, 61–64

X

XAML (Extensible Application Markup Language)

- animations, 206–208
- editing Visual Basic code for, 194–197
- graphical editor, 190–194
- objects, 197–199
- resources, 199–201
- styles, 201–202
- templates, 202–205
- transformations, 205
- user interface and code separated in, 188
- WPF with, 175

XAML code editor, 62–64, 68–69

XBAP (XAML Browser Application), 226–227

xElement object, 284–285, 1074

XML (Extensible Markup Language)

- comments, 254–258, 330
- creating snippets, 83–85
- DataSet object and, 428–429
- initializing XML variables, 284–285
- LINQ into XML, 482–483, 1073
- LINQ out of XML, 483–487, 1073–1075
- LINQ to XML, 481, 1073
- snippets stored as, 81

System.Xml namespace, 891–893
variable declarations, 925

XML literal, 481–483

XML Paper Specification documents (XPS). WPF, 221–222

XmlTextReader class, 891–893

XmlTextWriter class, 891–893

Xor operator, 311, 313

XPS (XML Paper Specification) documents, WPF, 221–222

Y

Y2K bug, 377

Z

ZIP files, and XPS documents, 221