Index

Note to reader: **Bolded** page numbers refer to main discussions and definitions. *Italicized* page numbers refer to illustrations.

Symbols
- # (pound) character, 53, 66, 649
- % (percent) wildcard character, 649
- & (ampersand) character, 531
- &nbsp: character, 850
- [ - ] wildcard character, 649
- [ ] wildcard character, 649
- [ ] (underscore) wildcard character, 649
- < (less than) character, XML, 531
- <%= and %> tags, XML, 599–600
- <!-- --> (comments), HTML, 851
- > (greater than) character, XML, 531
- ‘ (apostrophe) character, XML, 531
- ” (quotation mark) character, XML, 531

A
- ABC principles of WCF, 912
- Abort value, DialogResult enumeration, 228
- AcceptButton property, 205
- Accept/Reject Rule, foreign key constraints, 792, 793
- AcceptsReturn property, 164
- AcceptsTab property, 164
- access keys, 247–248
- Action attribute, HTML forms, 854
- action queries
  - defined, 973
  - deleting rows, 659–660
  - editing existing rows, 661
  - inserting new rows, 660–661
  - overview of, 658
- Activated event, 221–222
- Activation property, ListView control, 294, 294
- Active Server Pages (ASP), 863–864
- Add Connection dialog box
  - checking for sample databases, 634, 635
  - generating typed DataSet, 771–772, 772
- Add Data Source command, 770
- Add Item buttons, ListBox, 189
- Add method
  - adding nodes at runtime, 292–293
  - appending new items to collection, 81
- Controls collection, 234
- DateTime class, 479, 480
- Items collection, 185–186, 297
- TimeSpan class, 488–489
- Add New Item dialog box
  - accessing ASP.NET web services with AJAX, 911
  - adding ADO.NET Data Service, 921
  - creating master page, 876
  - creating new EDM, 734
  - working with LINQ to SQL classes, 615
- Add Reference command, 24–25, 334, 570
- Add Service Reference dialog box, 896
- Add Web Reference command, 24–25
- Add Web Reference dialog box, 908, 909
- AddDays method, 339
- AddExtension property,
  - OpenFileDialog/SaveFileDialog controls, 262
- AddRange method, collections, 503
- address, WCF web services, 912, 916–920, 917–920
- address visualization form, 833–843
  - assembling, 834, 835
  - composing web services, 832–834
  - constructing geocoding web service
    - URL/query parameters, 834–836
    - displaying coordinates on map, 838–842, 843
    - invoking geocoding web service, 836–837
    - overview of, 831–832
    - processing geocoding web service response, 837–838
  - AddToSuppliers method, ADO.NET Data Services, 933
- ADO.NET, programming with
  - action queries, see action queries
  - basic data-access classes, see classes, basic data access
  - databases, see databases
  - overview of, 631
- SQL, see SQL (Structured Query Language)
- storing data in relational databases, 972
- stream- vs. set-based data access, 662
ADO.NET Data Services, 920–938
  accessing methods of, 985
  building Windows client, 928–932, 929
  filtering, 926–928, 927
  overview of, 920–926, 922–924, 926
  performing transactions, 934–936
  real-world scenario, 937–938
  securing, 937–939
  submitting updates, 932–933
Advanced Options dialog box, TableAdapter
Configuration Wizard, 784, 784
Advanced submenu, Edit, 22
aggregates
  calculating with LINQ, 587–589, 592–593
  calculating with SQL, 651–653, 652
AJAX (Asynchronous JavaScript and XML)
  Extensions controls, 871, 872
  overview of, 863
  web services implementation, 910–912
algorithms, binary search, 498
align attribute, <img> tag, 850
alignment
  Label3D, 369, 371
  ListView, 293, 294
  LoanCalculator GUI controls, 135–136, 136
  MathCalculator GUI buttons, 146
All method, Controls collection, 234
AllowBrowserDrop property, WebBrowser, 818
AllowColumnReorder property, ListView, 294
AllowFullOpen property, ColorDialog, 257
AllowNavigation property, WebBrowser control, 817, 818
AllowScriptChange property, 259
AllowVerticalFonts property, 259
AllSystemSources property, 179
AllUrl property, 179
alt attribute, <img> tag, 850
Amazon.com, 894
American National Standards Institute (ANSI), 52
Amount argument, Pmt( ) function, 132–133
ampersand (&) character, 531
Ancestors methods, traversing XML documents, 543–544
Anchor property, forms, 216–217, 216–217
anonymous types, LINQ and, 583–584
ANSI (American National Standards Institute), 52
AnyColor property, ColorDialog, 257
apostrophe (‘) character, XML, 531
appearance, form
anchoring controls, 216–217, 216–217
Contacts project, 212, 212–213
docking controls, 216, 217–218, 218
elements of, 203–204, 204
form events, 221–223
Form object properties, 204–209, 205, 207
handling keystrokes, 213–214
placing controls, 209–210, 210
processing keystrokes from within code, 214–216
setting TabIndex property, 211, 211
splitting into multiple panes, 219–221, 220–221
Append/AppendFormat methods, 473–474
AppendText method, 443, 951–952
application generator, 242–243
Application object, 875
arguments, 106–125
  built-in functions, 111–113
  custom functions, 113–115, 114
  overloading functions, 118–125, 120, 123–124
  overview of, 106–108
  passing, 108–111, 115–118, 945
Array class
  additional operations of, 498–500
  ArraySearch application, 496, 496–497
  binary search algorithm, 498
  extension methods, 408–411
  limitations, 500
  overview of, 493
  searching arrays, 495–496
  sorting arrays, 494
  XML serialization of, 567–569
ArrayList class
  adding/removing items from, 502–505
  collection initializers, 505–506
  creating collections, 501–502
  extracting items from, 505–506
  iterating through collections, 509–510
  iterating with enumerators, 520–521
  searching collections in, 508–509
  serialization, 573–575
  sorting collections in, 507–508
arrays, 75–82
  collections vs., 81–82
defining, 75–76, 944
functions returning, 117–118
initializing, 76–78
multidimensional, 78–79, 78–81
ARRAYS, ADVANCED TOPICS • BUSINESS OBJECTS

overview of, 75, 75
searching, 964–965
arrays, advanced topics
additional operations, 498–500
ArraySearch application, 496, 496–497
binary search algorithm, 498
limitations, 500
overview of, 493
searching arrays, 495–496
sorting arrays, 494
As keyword, 39, 105
ASP (Active Server Pages), 863–864
ASP.NET
creating web applications, 875–879, 876–879
Development Server, 906
overview of, 864
Web Application option, 864
ASP.NET web service
adding simple AJAX implementation, 910–912
creating simple, 900–903, 902–903, 984
deﬁned, 983
overview of, 898–899
technologies, 899–900
asynchronous download, WebClient class, 830–831
Asynchronous JavaScript and XML. see AJAX
(asynchronous JavaScript and XML)
asynchronous methods, WebClient class, 828
Attribute property, XML documents, 544
attributes
adding images to web page, 850
entity, 726
HTML form, 854–855
HTML tag, 848
XHTML tag, 847
XML document, 532–534
audio, accessing using My component, 34
authentication, generating typed DataSet, 772
AutoCompleteSource property, TextBox controls, 179–180, 179–180, 947
AutoGenerateColumns property, 798
auto-implemented properties, 319
AutoSize property, 205
AutoSize property, 205, 205–206, 218–219
AutoSizeMargin property, 206
AutoSizeMinSize property, 206
AutoSizePosition property, 206
AutoSize property, 11, 14
auxiliary form, MultipleForms project, 229–233
AvailableFreeSpace property, 442, 444
Average extension method, 409
AVG() function, aggregates, 652
axis methods, 543–545
B
barcode reader, 691
baseline alignment, controls, 210, 210
BeginTransaction method, Connection object, 696
bidirectional data binding, 778
binary serialization, 560, 569–571
BinaryFormatter method, 569–571, 572
BinarySearch method, 965
ArraySearch application, 496, 496–497
binary search algorithm, 498
searching arrays, 495–496
searching collections, 508–509
BinaryWriter object, 960
binding. see data binding; data-bound applications
BindingSource class, 781–786
adjusting DataSet properties, 782–784, 783–784
designing data-driven interfaces, 788
handling Identity columns, 782
overview of, 781–782
using as data source, 794
block-level scope, 72, 72–73
body, HTML pages, 848
Booking.com, 894
Books and Authors EDM application, 764–766
Boolean operators, 49, 50
Boolean variables, 50
border attribute, <img> tag, 850
BorderStyle property, RichTextBox, 271, 273
boxing, 61, 392
brand independence, Entity Framework, 731
breakpoints, 27, 153
browser. see WebBrowser control
Brush class, 449
bulleted lists (<ol>...<ol>), displaying, 851
BulletIndent property, RichTextBox, 271, 273
business layer, 692–693
business objects, 687–701
application’s interface, 690–691, 690–693
implementing data-access layer with custom objects, 693–697, 697
implementing data-access layer with LINQ to SQL, 697–699
NWOrders application, 689
overview of, 687–689
using TransactionScope class, 699–701

Button control
adding to toolbar, 790
address visualization form, 834, 835
creating HTML forms, 855
first VB application, 14–17, 15
LoanCalculator GUI, 135
MathCalculator GUI, 145–146

Byte variable, 49
ByVal keyword, 108–109

Cache-Control response header, 828
CalculateStatistics() function, 109–110
calculators. see LoanCalculator project; 
MathCalculator project
Cancel value, DialogResult, 228, 228–229
CancelAsync method, WebClient class, 829
CancelButton property, 205
CanUndo/CanRedo properties, RichTextBox, 276
Caption property
DataSet, 782
designing custom controls, 372
capturing
function keys, 177–178
keystrokes, 176–177
cardinality, 728
cart web form, 886–891, 891
Cascading Style Sheets (CSS), 848, 856–860, 
982–983
Case statements, 90–91
case-sensitivity
arrays, 496
first VB application, 17–18
SQL, 462
XHTML, 846
casting, 397–398, 417
Catch Exception block, 156–157
Categories table, 636, 636–637, 791–792, 791–794
cells, HTML table, 852
CellStyle Builder dialog box, 797, 797
Changed events, custom Windows controls, 
375–376
ChangeExtension method, 444, 445
ChangeInterceptor method, 938
Char class, 463–464, 464–465
Char data type, 51–52, 961
Character variables, 51–52
CharacterCasing property, 164
text management, 850
type, 45, 45
wildcard, 649, 649–650
CheckBox control
creating HTML forms, 856
designing LoanCalculator GUI, 135
designing Windows applications, 130
CheckBoxes property, TreeView, 290
Checked property, 246
CheckedListBox control
basic properties, 183–184, 184
customizing, 382–383
overview of, 182, 182–183
selecting items, 187–188
CheckFileExists property,
OpenFileDialog/SaveFileDialog, 262
CheckPathExists property,
OpenFileDialog/SaveFileDialog, 262
cart nodes, tree structure, 287–288
ChildKeyConstraint property rules, 716–717, 717
Choose Toolbox Items command, 25
Choose Your Database Objects page, Data Source 
Configuration Wizard, 772–773, 773
Class Diagram Designer, 403, 403, 429
class instantiation, 387
classes
ADO.NET Data Service, 921–922
building WCF service, 913–916
combining code with data, 307–308
defined, 306
objects vs., 305–307, 387–388
classes, basic data access
Command class. see Command class
Connection class, 663–665
data-driven applications, 680, 680–687
DataReader class. see DataReader class overview of, 662–663
SQL injection attacks, 673–674
classes, custom
adding code, 311–313, 312
auto-implemented properties, 319–320
building Minimal. see Minimal class
building own, 952–953
Class Diagram Designer, 429
combining code with data, 307–308
custom enumerations, 323–325, 325
customizing Equals method, 321–322
customizing ToString method, 320–321
extension methods, 407–411
firing events, 335–338, 336
inheritance and, 401–402
instance and shared methods in, 338–342
Is operator in, 322–323
nesting classes, 344–347
object constructors and, 331–333
objects and, 305–306
overloading operators, 347–351
property procedures, 313–314
read-only properties, 317–318
revising Contacts project, 325, 325–330
StringTools class, 342–344, 344
throwing exceptions, 315–316
unary operators, 351–352
using SimpleClass in other projects, 333–335
variants, 352–353
working with, 953–956
class-related modifiers, 418–420
cleanup statements, structured exception handlers, 156
Clear button, 146, 148
Clear method
  Controls collection, 234
  Items collection, 186, 297
clearDisplay variable, MathCalculator GUI, 148, 150–151
Click event handler, Button control
  LoanCalculator GUI, 136–137, 137
  MathCalculator GUI, 147–149
  programming, 15, 15–17, 19–20
Click event, ToolStripMenuItem, 949–950
Clickable value, HeaderStyle property, 294
client
  building Windows, 928–932, 929
  maintaining state, 871, 873
client variable, consuming web services, 897
Clipboard, 34
Close command, Control menu, 204, 204
Cloud tab, Start page, 5
CLR (Common Language Runtime), 65, 677–678
code
  adding to Minimal class, 311–313, 312
  data combined with, 307–308
  processing keys from within, 214–216
  programming first VB application, 14–18, 15
  reuse of, 306. see also inheritance
code editor, 15, 15
code snippets
  calculating monthly loan payment, 139–140
  in Framework, 434–436, 435
  Insert Snippet command, 23
  using My component, 33–34
  working with, 32–33
code-behind
  creating cart web form, 888–890
  defined, 864
  developing stand-alone web service, 909–910, 910
collection initializers, Array class, 505–506
collections
  adding/removing list items, 502–505
  additional, 519
  advanced array topics. see Array class
  collection types. see collections
  creating, 501–502
  Dictionary, 510–512, 965–966
  dynamic data storage, 81–82
  extracting items from, 506–507
  HashTable, 512–513
  IComparer interface. see IComparer interface
  IEnumerable interface. see IEnumerable interface
  implementing custom comparers, 966–967
  iterating through lists, 509–510
  List, 965
  overview of, 493, 500–501
  querying with LINQ, 584–586
  searching arrays, 964–965
  searching lists, 508–509
  serialization, 573–575
  SortedList, 518–519
  sorting lists, 507–508
  transforming into XML segments, 596–597
  WordFrequencies application, 513–517, 514
  XML serialization of, 567–569
color
  creating graphics, 448
  as important design element, 12
  MathCalculator GUI, 145
  prompting users for, 951
Color property
   ColorDialog, 257
   FontDialog, 259
ColorClick event, 377–378
ColorDialog control
   function of, 255
   prompting users for colors, 951
   properties, 256–258
ColorEdit control, 365, 365–368, 377–378
Colors project, 198, 198–200
Column attribute, LINQ to SQL, 611–613
column names, aliasing, 648–649
ColumnHeader Collection Editor dialog box, ListView, 295–296
columns
   creating HTML tables, 852
   customizing appearance, 794–798, 795–797
   data binding, 778–780, 779–780
   displaying lookup, DataGridView, 798, 798–799
   handling Identity, 716
   retrieving database update errors, 716
Columns collection, DataTable class, 707–708
ColumnType property, 795–796
Combine method, 444, 445
ComboBox control
   adding items at runtime, 195, 195–196
   creating user friendly application, 18–19, 18–21
   customizing, 382–383
   displaying options, 18
   FlexCombo project design, 196–197
   NWOrders application UI, 690–691
   overview of, 182, 182–183
   properties, 183–184, 184
   selecting items, 187–188
   types, 193–194, 193–195
   Windows application design, 130
Command class
   application of, 665–667
   empty strings vs. null values, 675
   executing selection queries, 667–668
   executing statements during transactions, 696
   executing stored procedures, 667
   handling special characters, 674–675
   retrieving multiple values from stored procedures, 670–673
   using Commands with parameters, 669–670
Command window, 24, 27–28
CommandBuilder class, 705–707
commands, Control menu, 204, 204
CommandText property, 973
CommandType property, 973
comments, 22, 851
Commit method, Transaction object, 696
Common Controls tab, Toolbox window, 8, 8, 26
Common Language Runtime (CLR), 65, 677–678
Compare method
   DateTime class, 479
   features of, 467–468
   implementing custom comparers, 966
CompareOrdinal method, 468
comparers, custom, 966–967
Complete Word, IntelliSense, 23
complex property, adding to entity, 747–748
complex types, Entity Framework, 729
compound controls
   ColorEdit, 365, 365–368
   mapping members to, 956–957
   overview of, 364–365
Concat method, 468
concatenation operator (&), variants, 60
conceptual model, EDM structure, 738
conditions, 86
Configuration Editor, WCF services, 917–919
   configuration wizard, 5
connection
   Data Source Configuration Wizard, 770–772, 771–772
   to web services, 908
Connection class, 663–665
connectionString setting, NWOrders application, 690
console application, building, 30–32
constants, 74
constituent controls, 364
constraints, aborting transactions, 697, 697
constructors, 331–333, 954–955
consuming web services
   ADO.NET Data service, 928–929, 929
   ASP.NET, 902–903, 984
   overview of, 894–898, 895, 897–898
contacts, 328–330
Contacts project
   revising for database, 680, 680–687
   revising with custom classes, 325, 325–330
   TabIndex property, 212, 212–213
Contains method, Items collection, 187
content pages, 874
context menus, 251, 818, 819
Context object, 875
Continue statement, 102
ContinueUpdateOnError property, DataAdapters, 788
contract, WCF web service, 912, 916–920, 917–920
Control menu commands, 204, 204
control structures, nested, 99–102
custom classes
adding code, 311–313, 312
auto-implemented properties, 319–320
building Minimal. see Minimal class
Class Diagram Designer, 429
combining code with data, 307–308
custom enumerations, 323–325, 325
customizing Equals method, 321–322
customizing ToString method, 320–321
data, 868, 869
dialog, 950
HTML, 871, 873
LoanCalculator GUI, 134, 134–135
login, 870, 871
manipulating in Intermediate window, 28
navigation, 868, 870
reporting, 871
Standard (web form), 867, 867–868
Text Box, 946–947
understanding. see Windows Form Designer
validation, 868, 869
Web Parts, 870, 872
Windows applications, 941–942
Controls collection, 234–236, 235, 949
custom classes
adding code, 311–313, 312
auto-implemented properties, 319–320
building Minimal. see Minimal class
Class Diagram Designer, 429
combining code with data, 307–308
custom enumerations, 323–325, 325
customizing Equals method, 321–322
customizing ToString method, 320–321
data, 868, 869
dialog, 950
HTML, 871, 873
LoanCalculator GUI, 134, 134–135
login, 870, 871
manipulating in Intermediate window, 28
navigation, 868, 870
reporting, 871
Standard (web form), 867, 867–868
TextBox, 946–947
understanding. see Windows Form Designer
validation, 868, 869
Web Parts, 870, 872
Windows applications, 941–942
Controls collection, 234–236, 235, 949
conversions
handling dates between locales, 54–55
of strings and numbers, 51
variable type, 62, 62–63
widening and narrowing, 64, 64–65
Convert class, 62, 62–63
Convert To Upper/Lower commands, 170
cookies
classes supporting, 828
managing state using, 871
managing with HttpWebRequest and
HttpWebResponse, 831
Copy command, 170–171, 170–171, 244
Copy method
Array class, 498–499
IO.File class, 442, 443
RichTextBox control, 276–277
CopyTo method
Array class, 498–499
manipulating Items collection, 187
COUNT() function, aggregates, 652
Count property
Controls collection, 234
Items collection, 186, 297
counter variable, 93–95
CountFiles( ) function, 121–124, 124–125
Create Directory For Solution check box, 6
Create/CreateText methods, IO.File class, 442,
443
CreateDirectory method, IO.Directory class, 441,
441
CreateEntityName method, EDM, 764
CreateGraphics method, 962–963
CreateQuery method, Windows client, 930–931
CRM system, Entity Framework, 729–731
C-S mapping content. EDM structure, 739
CSDL (Conceptual Schema Definition Language),
738
CSS (Cascading Style Sheets), 848, 856–860,
982–983
CType() function, 352–353
cultural differences, date formats, 477–478,
537–538
curly brackets, collection initializers, 502–505
cursor position, 167–168
objects and, 305–306, 307
overloading operators, 347–351, 351
overview of, 306–307, 953–956
property procedures, 313–314
revising Contacts project, 325, 325–330
StringTools class, 342–344, 344
throwing exceptions, 315–316
using SimpleClass in other projects, 333–335, 334–335
variants, 352–353

custom comparer function, 966–967
custom data types, 942–943
custom enumerations, Minimal classes, 323–325
custom events, 377–378
custom functions, 113–115
custom functions, LINQ to XML, 603
custom objects, XML serialization of, 563–567
custom sorting, IComparer interface, 522–523
custom Visual Studio IDE, 29
custom Windows controls. see Windows controls, building custom
CustomColors property, ColorDialog, 257–258
Customers table, Northwind database, 637, 638
CustomerTotals stored procedure, 670–673
CustomExplorer project, 299–300, 300
Customize command, 7
CustomSource property, AutoCompleteSource, 179
Cut command, 170–171, 170–171, 244
Cut method, RichTextBox, 276–277

D

data
code combined with, 307–308
combining from multiple tables, 647
displaying delimited, 274
ListBox control for storage of, 331
persisting, 559
querying with LINQ. see LINQ (Language Integrated Query)
streaming in Framework, 445–446
data handling
data types. see data types
large sets of, 517
repeated items, 239–241, 240
variables. see variables
data, storing in collections
additional collections, 519
Array class. see Array class
collection types. see collections
Dictionary collection, 510–512
HashTable collection, 512–513
IComparer. see IComparer interface
IEnumerator. see IEnumerator interface
overview of, 493
SortedList collection, 518–519
WordFrequencies application, 513–517, 514
data binding, 778–786. see also data-bound applications
BindingSource class, 781–786, 783–784, 786
of DataGridView to DataTable, 779–780
overview of, 778–780
real-world scenario, 784–785
WCF web service, 912, 916–920, 917–920
data controls, 868, 869
Data Manipulation Language (DML), 643
data manipulation, XML
editing documents, 549–551
lambda expressions, 557–559, 559
locating information, 547–549
overview of, 546–547
segments as literals, 551–557, 554
Data menu, 25, 770
data model, and Entity Framework, 726–728, 727
data source
adding ADO.NET Data Service, 921–922
using BindingSource class as, 794
Data Source Configuration Wizard, 770–771, 771
Data Sources window
binding hierarchical tables, 791, 791–792
generating typed DataSet, 773, 774
Data tab, Start page, 5
data types
Boolean, 50
Byte, 49
CLR vs. SQL, 677–678
converting variable, 62, 62–63
creating custom, 68, 68–70, 70, 942–943
Date, 53–54
native, 942
now part of CLR, 65
numeric, 42, 42
objects vs. basic, 61
Signed Byte (SByte), 49
String, 50–51
user-defined, 68, 68–70, 70
variable, 37–38
XML as, 540, 540–542
data validation. see validating data
data-access classes
building data-driven applications, 680, 680–687
Command class. see Command class
Connection class, 663–665
DataReader class. see DataReader class overview of, 662–663
SQL injection attacks, 673–674
data-access layer
with DataContext object, 700–701
defined, 692
implementing with custom objects, 693–697, 697
implementing with LINQ to SQL, 697–699
DataAdapters
DataAdapter class, 703, 703–705
designing data-driven interfaces, 788
filling DataSets, 702–703
loading DataSets, 720–721
populating DataTables of typed DataSet with, 774–775
updating database, 715–716
Database Designer tool, 641
Database icon, Data Source Configuration Wizard, 770, 771
database management system (DBMS), 631
databases
classes for accessing, see data-access classes
connecting EDM to, 734
Contacts application with, 680, 680–687
generating structure with EDM Designer, 749–750, 750
obtaining Northwind/Pubs sample databases, 633–635, 635
overview of, 631–632
submitting changes in DataSet to, 974–975
understanding relations, 640–641, 641
using relational, 632–633, 633
data-bound applications
data binding, see data binding
designing and using typed DataSets. see typed DataSets overview of, 769, 975–976
data-bound applications, interface design
adjusting DataGridView appearance, 794–799, 795–798
binding hierarchical tables, 791–792, 791–794
building more-functional interfaces, 801–808, 805
data binding with LINQ, 808–810, 809
editing data in hierarchical tables, 799–801, 800
enhancing navigational tools, 789–791
overview of, 786–789, 787–789
DataReader class, 707–708
DataContext object
creating, 618
implementing data-access layer with, 700–701
role in LINQ to SQL, 609–610, 614
data-driven applications, 687–723
creating and populating DataSet, 974
creating simple DataSet, 717–723
defined, 687
establishing relations between tables in DataSet, 974
performing update operations, 714–717
storing data in DataSets. see DataSets using business objects. see business objects
data-entry
Contacts application, 212, 212–213
Enter keystroke and, 211–212
TextBox control and, 181
DataGridView control
adjusting appearance, 794–799, 795–798
binding hierarchical tables, 791–792, 791–794
binding to DataTable, 786–787, 787
building more-functional interfaces, 801–804
customizing column appearance, 794–798, 795–797
data binding, 778–780, 779–780
displaying lookup columns, 798, 798–799
enhancing navigational tools, 789–791
using BindingSource as data source, 794
viewing deleted rows, 804–808, 805
viewing update errors, 787–789, 789
dataGridview ButtonColumn, 796
dataGridView CheckBoxColumn, 796
dataGridView ComboBoxColumn, 796, 796
dataGridView ImageColumn, 797
dataGridView LinkColumn, 797
dataGridView TextBoxColumn, 797
DataReader class
application of, 678–679
CLR types vs. SQL types, 677–678
overview of, 676
properties/methods, 676–677
reading multiple result sets, 679–680
DataRow objects
    adding and deleting rows, 710–711
    navigation through DataSet, 711–712
    row states and versions, 712–714, 713
    working with rows, 708–709
DataRowState enumeration, 712–713, 713, 722
DataRowVersion enumeration, 714, 714, 722
DataService project. see ADO.NET Data Services
DataSet, LINQ to, 581
DataSet Designer window, 782–784, 783–784
DataSetOperations project, 770–774
    adding third form to, 786
    generating typed DataSet, 770–773, 771–774
    iterating through linked DataTables, 776, 776–777
    manipulating rows of typed DataTables, 777, 777–778
    operations on typed DataSet, 774–776, 775
DataSets, 701–714
    accessing tables, 707–708
    adding and deleting rows, 710–711
    creating and populating, 974
    creating SimpleDataSet project. see
    SimpleDataSet project establishing relations between tables in, 974
    handling null values, 709–710
    navigating through, 711–714, 713
    overview of, 701–702
    populating and displaying, 721
    submitting changes to database, 974–975
    typed. see typed DataSets
    typed vs. untyped, 769–770
    working with Entity Framework vs., 751
    working with rows, 708–709
DataSets, filling, 702–707
    CommandBuilder class, 705–707
    DataAdapter class, 703, 703–705
    overview of, 702–703
DataSource property, data binding, 779
DataSvcUtil.exe utility, 928–932
DataTables
    accessing DataTable’s tables, 707–708
    adding and deleting rows, 710–711
    data binding, 778–780, 779–780
    establishing relations between tables in
    DataTable, 974
    generating typed DataSet, 773, 774
    iterating through linked, 776–777, 776–777
navigation through DataSet, 711–714, 713
populating DataSet using Fill method, 703
populating typed DataSet, 774–775
updating row of typed, 777, 777
working with rows, 708–709
DataService class, 808
date formats, 477–478, 537–538
date property, 478
Date variables, 53–54, 54–55
DateTime class
date conversion methods, 483, 484
dates as numeric values, 483–485
defined, 962
for different cultures, 477–478
format characters, 482–483, 483
methods, 479–480
overview of, 476
properties, 478–479
standard format patterns, 480–481, 481
DateTime values, XML documents, 538
Day property, WeatherForecast class, 897–898
DayOfWeek/DayOfYear properties, 478
Day(s) property
dateTime class
    Date class, 478
    TimeSpan value, 487
DaysInMonth method
    Date class, 478
    as shared method, 339
dblProperty, Minimal class, 311–313
DBMS (database management system), 631
Deactivate event, 221–222
Debug menu, 25
debugging
    defined, 155
    MathCalculator GUI, 152–155
    by printing to Intermediate window, 27–28
    Web applications, 866
    Web applications using Fiddler, 826, 826
Debug.WriteLine method, 27
debug.WriteLine statement
Decision statements
defined, 44–46
If...Then, 86–87
If...Then...Else, 87–90
nested control structures, 99–102
overview of, 944
Select Case, 90–91
declarations
array, 75–76
constant, 74
variable, 38–40
declarative languages, 830
Default value, Alignment property, 294
DefaultCellStyle property, 797–798
DefaultText property, 262
DefaultValue attribute, 363
DefaultValue property, DataSet, 782
defered (lazy) loading, 762–763, 975
dehydration, 969, see also serialization
Delete method
entity instances, 764
of I/O.Directory class, 441, 441
of I/O.File class, 442, 443
rows, 710–711
Delete rule, foreign key constraints, 792, 793
DELETE statement
optimistic concurrency, 784
rows, 659–660
TableAdapter Configuration Wizard, 784, 784
delimited data, 274
delimiters, 470
deploying web service, 905–906, 906
dereferenceLinks property, 262–263
derived class keywords, 419, 420
Descendants methods, traversing XML documents, 543–544
deserialization, 559, 969
deserialize method
with collections, 574–575
with custom objects, 567
with individual objects, 563, 571–573
serialization process and, 560–561
Design Patterns: Elements of Reusable Object-Oriented Software (Gama et al.), 731
details property, WeatherForecast class, 897–898
details value, View property, 293
dialog boxes
forms vs., 226–229, 227–228
MultipleForms project, 229–233, 230
testing custom Windows controls, 374, 374–375
dialog controls
ColorDialog, 256, 256–258
FolderBrowser demo project, 267–269
FolderBrowserDialog, 266, 266–267
FontDialog, 258
FontDialog control, 258–261
MultipleFiles project, 265, 265–266
OpenDialog/SaveDialog, 261–265
OpenFileDialog/SaveFileDialog, 261
overview of, 254, 254–255
using, 255–256
DialogResult property, 228, 228–229
dictionary collection
creating, 81–82
populating, 965–966
storing data in, 510–512
XML serialization of, 567–569
DigitClick event handler, Button control, 148
dim statement
declaring arrays, 75–76
declaring variables, 39
declaring variants, 59
DirectCast( ) function, 63
Directory class, 441, 441–442
DirectoryInfo class, 443–444
DiscardChanges() function, 280
DISCO document, 908
DisplayStyle property, ComboBox control, 22
DISTINCT keyword, 649
Distinct method, LINQ, 589
<div> tag, CSS, 858–860, 860
<div> tag, HTML, 854
<div> tags, CSS, 854, 858–860, 860
DML (Data Manipulation Language), 643
Dock property
form design, 216, 217–218, 218
WebBrowser, 817, 823
docked windows, IDE, 9–10, 10
!DOCTYPE definition, 847, 848
Document Object Model (DOM), AJAX, 863
document property, WebBrowser, 820–821
documentation, overloaded functions and
definitions, 124–125
documentCompleted event, WebBrowser, 822
documentStream property, WebBrowser, 820
documentText property, WebBrowser, 819–820
do...Loop statement, 96–98
DOM (Document Object Model), AJAX, 863
_doPostBack function, 879
double data type, 41–44, 143, 143
double-precision numbers, 41–44
download methods, WebClient class, 828, 829
download*Async methods, WebClient class, 829,
830, 837
download*Completed events, WebClient class, 829
DownloadStringCompleted event handler, 837–838
drag-and-drop behavior, WebBrowser, 818
DrawImage method, 453, 461–462
drawing, in Framework. see also graphics
generating gradients, 453–454
generating printouts, 460, 460–462
Graphics object, 446–447
Image class, 454–455
methods, 449–453
overview of, 446
Pen/Brush classes, 449
Point/Rectangle classes, 447
Size/Color/Font classes, 448
DrawItem event handler, 958
DrawItem events, 382–383
DrawMode property, 383, 386
DrawMode property, 958
DrawString method, 451–452, 459
DriveFormat/DriveType properties, 442, 444
DriveInfo class, 442, 444
DropDownOpened event, 949–950
Due argument, Pmt( ) function, 132
dynamic content, adding, 599–601
dynamic forms
creating event handlers at runtime, 241–242
designing application generators, 242–243
DynamicForm project, 236–239, 237
Form’s Controls collection, 234
handling repeated data items, 239–241, 240
overview of, 233, 949
ShowControls project, 234–236, 235
Dynamic Help command, 26
dynamic queries, advanced search forms with
eSQL, 755–758
DynamicForm project, 236–239, 237

E
e argument, event handlers, 16
every-bound variables, 398–399
edge alignment, controls, 210, 210
Edit Columns dialog box, 795, 795–799, 798
Edit DataSet button, 717, 718, 722
Edit menu
  Advanced, 22
  captions and names, 243–244, 244
  IntelliSense, 22–23, 23

Outlining, 24
in RTFPad project, 281–283
editing
columns, 795, 795
commands, TextPad application, 169–171, 170–171
data in hierarchical tables, 799–801, 800
RichTextBox control, 275–276
TextBox control, 168
EditPad Pro, 848
EDM (Entity Data Model), 975
adding complex property to entity, 747–748
adding many-to-many association, 748–749, 975–976
adding one-to-many association, 748, 975–976
connecting to database, 734
creating entity, 735
creating new, 734
creating primary key, 735
creating scalar property, 735–736
deferred (lazy) loading, 759–763, 975
database inheritance, 743–747, 975
generating database structure, 749–751, 750
modifying data with Entity Framework, 763–767
overview of, 732–733
querying, 751
reverse-engineering, 767
using EDM Designer, 733, 733
using eSQL, 754–758
using LINQ to Entities, 752–754
using Query Builder methods and
ObjectQuery class, 758–759
EDM (Entity Data Model), structure, 736–743
C-S mapping content, 739
CSDL content, 738
SSDL content, 739
XML content, 736–738
EDM Designer, 733, 733–734
EDM diagram, 733, 733
Effect property, Label3D control, 370, 371
Element/Elements methods, 543
elements, XML documents, 530–534
elliptical controls, 379, 379–382
Else keyword, 87–90
ElseIf statement, 87–90
<embed> tag, media, 851
embedding media, HTML, 851
Employees table, Northwind database, 637
Empty property, TimeSpan, 486, 486
designing data, 726–727
evnt, 246
End Try statement, exception handling, 156, 158
error handling
only variable, For . . . Next loops, 93–95
types vs. errors, 155
EndsWith method, 468
overview of, 143–144
endpoints, WCF services, 918–919
updating database, 715
EnterFocusColor property, 360–361
writing robust applications, 946
adding complex property to, 747–748
Error List window, 28–29
event handlers
adding inheritance between, 743–747
changing name of controls after inserting code in, 16
creating at runtime, 241–242
creating first VB application, 16
declared, 14
designing Windows applications, 130
defined, 14
overview of, 942
programming MathCalculator GUI, 147–149
creating at runtime, 241–242
defining, 14
overview of, 942
programming loan calculator GUI, 136–140, 137
programming MathCalculator GUI, 147–149
adding inheritance between, 743–747
designing data, 726–727
EDM implementing, see EDM (Entity Data Model)
deleting, 764
creating new instances, 735
databasing, see EDM (Entity Data Model)
deleted, 631
creating to add to EDM, 735
defined, 632
deleting, 765
EDM defining, 732
establishing relationship between, 728, 765–766
events
identifying by primary key, 726
LINQ to, 581
one-to-many association between, 748
updated, 765
Entity Data Model. see EDM (Entity Data Model)
events
Entity Framework, 725–732
adding complex property to, 747–748
data store and brand independence, 731–732
adding inheritance between, 743–747
EDM defining, 732
establishing relationship between, 728, 765–766
events
EDM. see EDM (Entity Data Model)
EDM implementing, see EDM (Entity Data Model)
identifying by primary key, 726
LINQ to, 581
many-to-many association between, 748–749
one-to-many association between, 748
updating, 765
Entity Data Model Wizard, 734
Entity Framework, 725–732
data store and brand independence, 731–732
EDM defining, 732
establishing relationship between, 728, 765–766
identifying by primary key, 726
LINQ to, 581
many-to-many association between, 748–749
one-to-many association between, 748
updating, 765
Entity SQL (eSQL), 754–758, 977–978
ETag response header, 828
exceptions
changing name of controls after inserting code in, 16
creating at runtime, 241–242
creating first VB application, 16
defining, 14
designing Windows applications, 130
defined, 14
overview of, 942
programming events, 945–946
programming loan calculator GUI, 136–140, 137
programming MathCalculator GUI, 147–149
Events
Changed, 375–376
firing class, 335–338, 336
HTML, 861, 861–862
overview of, 942
programming, 945–946
raising custom, 377–378
WebClient class, 822
WebBrowser control, 822
WebClient class, 829
exceptions
errors vs., 155
MathCalculator project, 155–158
within property procedures, 315–316
ExecuteNonQuery method, 696, 973
ExecuteQuery method, 613–615, 973
ExecuteNonQuery method, 696, 973
ExecuteQuery method, 613–615, 973
Exists method
I0.Directory class, 441, 442
I0.File class, 442, 443
Exit command, 170, 244
Exit statement, 102
Exit Sub statement, 103–104
Expedia.com, 894
explicit loading, 761–762
Explicit option, 55–59, 942
Expression Blend, 12
expressions, 27, 91–93
extensible HTML (XHTML), 846–847
Extensible Markup Language. see XML (Extensible Markup Language)
extension methods, 407–411, 583–584
external events, 333
ExtractFileName method, 342
ExtractPathName method, 342

F
fetch translators, 753–754
Fiddler tool, 826, 826
FieldCount property, DataReader object, 676
fields
 calculated, 651
 hidden in web page, 873
 properties vs., 314, 395
 relational database, 633
 selection list, 646
 TimeSpan object, 486
File class, 442, 443
File menu
 captions and names, 243–244, 244
 overview of, 21–22
 in RTFPad project, 279–281
filenames, prompting users for multiple, 950
FileName(s) properties, 263, 264, 950–951
files
 handling with My component, 959–960
 manipulating in IO namespace, 960–961
 writing data with IO namespace to, 960
FileStream object, 960
FileSystem property, 179
Fill method
 adding Refresh Data button to call, 790
 populating and displaying DataSet, 703, 721
 populating typed DataSet with proper TableAdapters, 775–776
Filter() function, Array class, 499–500
Filter property, BindingSource class, 781
Filter/FilterIndex properties, 263
filtering
 ADO.NET Data Services and, 926–928, 927
 with LINQ vs. eSQL dynamic query, 977–978
 product data with LINQ, 809, 809–810
Finally block, 156, 158
Find & Replace command, 170
Find button, adding to toolbar, 790
Find method, BindingSource class, 781–782
Find/Find Next buttons, TextPad application, 172–175
FindString method, ListBox control, 947
FindStringExact method, ListBox control, 947
FindStringFindStringExact methods, 191
FixedPitchOnly property, FontDialog control, 259
FixedToolWindow value, 207, 207
FlexComboBox project, 195, 196–197
flow format, HTML, 854
flow-control, defined, 944
flow-control statements, 85–102
 Continue statement, 102
 Do...Loop loop, 96–98
 For Each...Next loop, 95–96
 Exit statement, 102
 For...Next loop, 93–95
 If...Then statements, 86–87
 If...Then...Else statements, 87–90
 nesting, 99–102
 overview of, 85–86
 Select Case statements, 90–91
 short-circuiting expression evaluation, 91–93
 While...End While loop, 98–99
FocusedTextBox control
 adding functionality, 359–362
 building, 357, 357–359, 359
 classifying properties, 363–364
 inheritance and, 401–402
 testing, 362, 362–363
FolderBrowserDialog control
 defined, 951
 function, 255
 properties, 266, 266–267
 sample application, 267–269
folders
 creating in Solution Explorer window, 26
 creating/removing project, 6
 manipulating in IO namespace, 960–961
Font class, 448
Font command, 170
font families, 11–12
font property
 FontDialog control, 259–260
Label3D control, 369
FontDialog control
function, 254, 255
overview of, 951
properties, 258, 258–261
FontMustExist property, FontDialog control, 260
fonts
creating first VB application, 14
as design element, 11–12
designing LoanCalculator GUI, 133–135, 134
designing MathCalculator GUI, 145
HTML, text management tags, 849
footers, master page setup, 874
For Each...Next loop statement, 95–96, 708
For Next loop statement, 708
forecast variable, consuming web services, 897
foreign key constraints, 793, 793
foreign keys, relational databases, 640
form design
anchoring controls, 216–217, 216–217
Contacts project, 212, 212–213
docking controls, 216, 217–218, 218
dynamic forms, 949
elements of, 203–204, 204
form events, 221–223
Form object properties, 204–209, 205, 207
handling keystrokes, 213–214
multiple forms, 948–949
placing controls, 209–210, 210
processing keys from within code, 214–216
setting TabIndex property, 211, 211
splitting into multiple panes, 219–221, 220–221
working with, 948
Form Designer, 7
<form> tag, HTML, 854–855
<form>...<form> tags, HTML, 854–855
Format menu, 25
Format menu commands, 171–172, 283–284
formats
microformats on websites, 863
number, 65–68, 66–67
pages in HTML, 853–854, 854
pages with CSS, 858–860, 860
formatString argument, 65–66
FormBorderStyle property, 207, 207
FormClosed event, 222
FormClosing event, 222
fonts
building dynamic, see dynamic forms
controls, see controls
designing menus, see menus, designing
events triggered by Form objects, 221–223
HTML, see HTML forms
HTML element events, 861
overview of, 203
forms, loading and showing
forms vs. dialog boxes, 226–229, 227–228
multiple instances of single form, 226
MultipleForms project, 229–233, 230
overview of, 223–224
sharing variables, 225–226
Startup Form, 224–225
form-wide variables, 150
For...Next loop statement, 93–95, 101
fragment caching, 864
frames, HTML, 854
Framework
drawing and painting, see graphics
extending classes, 410–411
handling dates, see DateTime class
handling files with My component, 436–440, 959–960
handling strings and characters, see strings and
characters
handling time, see TimeSpan class
10 namespace, see IO namespace
manipulating folders and files, 960–961
overview of, 433–434
printing controls and dialog boxes, 963
using code snippets, 434–436, 435
front-end applications, 687
full joins, in SQL, 654
FullRowSelect property
ListView control, 295
TreeView control, 290
function keys, 153
Function keyword, 105
functional construction, 539
Function...End Function statements, 104–106
functions
accepting arguments, see arguments
built-in, 111–113
custom, 113–115, 603
documenting overloaded, 124–125
overloading, 118–125, 120, 123–124
passing arguments to, 945
returning arrays, 117–118
future value argument, \( PMT( ) \) function

- returning multiple values, 109–110
- subroutines vs., 103
- validating user input, 71–72
- writing and using, 104–106, 944–945

FutureValue argument, \( Pmt( ) \) function

G

- Garbage Collector (GC), 393–394
- geocoding service. see Yahoo! Geocoding API
- Go navigation methods, WebBrowser, 821, 821
- Google Maps service, 833–834, 838–842, 843
- graphical user interface. see GUI (graphical user interface) design
- graphics
  - drawing methods, 449–453, 453, 962–963
  - gradients, 453–454
  - Graphics object, 446–447
  - Image class, 454–455
  - overview of, 446
  - Pen/Brush classes, 449
  - Point/Rectangle classes, 447
  - printing, see printing
  - printouts, 460, 460–462
  - Size/Color/Font classes, 448
- Graphics object, 446–447
- greater than (\( > \)) character, XML, 531
- grid control, adding, 881–883
- Gridlines property, ListView, 295
- grids, customizing appearance, 795
- GROUP BY clause, rows, 656–658
- grouping
  - with LINQ, 591–592
  - rows with SQL, 656–658
- Group property, ListView, 295
- GUI (graphical user interface) design
  - alignment of controls on form, 945
  - building calculator. see MathCalculator project
  - building loan calculator. see LoanCalculator project
  - designing Windows applications, 129–130
  - NWOrders application, 690–691, 690–693
- overview of, 945
- programming events, 945–946

H

- Handles keyword, 16
- HasErrors property, updating database, 715
- HashTable collection
  - iterating with enumerators, 521–522
  - storing data in, 512–513
  - for WordFrequencies project, 513–517, 514
- HasRows property, DataReader, 676
- HAVING clause, 658
- hCard specification, 863
- headers
  - HTML page. see controls
  - master page, 874
Headers property, WebClient class, 828
HeaderText property, ListView, 294
headings
  HTML table, 852
  HTML text management tags, 849
height attribute, &lt;img&gt; tag, 850
Hello method, ASP.NET web service, 901–902, 902
HelloWorld web service, 901–903, 902–903
Help menu, 26
hidden fields, 873
HideSelection property, 168, 290
hierarchical tables
  binding, 791–792, 791–794
  editing data in, 799–801, 800
hierarchy of classes, 403, 403
HistoryList property, 179
horizontal partitioning, 732
horizontal resolution, Image class, 454
horizontal rules (&lt;hr&gt; tag), HTML, 850
HotTracking property, TreeView, 290
Hour(s) property, DateTime class, 478
Hour(s) property, TimeSpan value, 487
href attribute, &lt;a&gt; tag, 850
HTML (Hypertext Markup Language)
  comments, 851
  creating basic page, 981
  defined, 871
  embedding media, 851
  forms and form elements, 854–856
  horizontal rules, 850
  images, 850
  links, 850–851
  lists, 851
  overview of, 847, 873
  page construction, 848–849, 849
  page formatting, 853–854, 854
  RTF comparison, 270–271
  scripts, 851
  tables, 852–853, 853
  tags for transforming XML documents, 601–605, 602
  text management, 849–850
  upgrading to XHTML from, 847
  web development and, 845–846
HTML forms
  elements, 854–856
  embedding Yahoo! Finance badge, 824–825, 825
  overview of, 854
  WebBrowser Document property, 820–821
HTTP
  debugging proxy, 826, 826
  publishing services on Internet, 827
  web services using, 917
HttpWebRequest class, 831
HttpWebResponse class, 831
I
IComparer interface
  custom sorting, 522–523
  implementing, 523–525
  implementing custom comparers, 966–967
  implementing multiple comparers, 525–527
  overview of, 519–520
IDE (integrated development environment), 3–15
  font as design element, 11–12
  overview of, 3–4
  rearranging windows, 9
  Start Page, 4, 4–5
  starting new project, 5–7, 6–7
  using Windows Form Designer, 7–13, 8–11, 13
IDE (integrated development environment) components, 5–15
  Command and Immediate windows, 27–28
  components, 941
  Error List window, 28–29
  menus, 22–26
  Output window, 27
  overview of, 941
  Properties window, 26–27
  Solution Explorer window, 26
  Toolbox window, 26
Identity columns
  adjusting DataSet properties, 783
  BindingSource class and, 782
  data-driven applications, 716–717, 717
IEnumerable interface
  LINQ and, 584–586
  sample application, 589–591
IEnumerator interface
  enumerating collections, 520–522
  overview of, 519–520
iFrames (inline frames), 854
If...Then statement, 86–87, 101–102
If...Then...Else statement, 87–90
Ignore value, DialogResult, 228, 228
IIf() function, 89
IIS (Internet Information Services), 863, 905–906, 906
Image class, 454–455
images (<img> tag), 850, 853, 853
Immediate window
debugging by printing to, 27–28, 155
defined, 24
Inches application, TrackBar, 200–201
Include method, EDM, 759–761
increment variable, For...Next loops, 93–95
Indent property, TreeView, 290
Index command, 26
IndexOf method
applying to any string, 149
features of, 469
searching arrays, 495–496
searching collections, 508–509
IndexOfAny method, 469
Infer option, 55–59, 942
Infinity value, 46–49
inheritance
adding between entities, 743–747
class-related modifiers and, 418–420
designing with, 402–407, 403
Entity Framework and, 743–747, 975
finding entity of specific type, 753–754
how to apply, 401–402
InheritanceKeywords application, 420–423
overview of, 400
sample application, 423–428
understanding, 959
InheritanceKeywords project, 420–423
Inherits statement, 419, 959
InitialDirectory property, 263–264
initialization
ArrayList collection, 505–506
arrays, 76–78
multidimensional arrays, 79–80
variables, 40, 74
inline frames (iFrames), 854
inner joins, SQL, 654
<input> tag, HTML, 856
Insert method
adding items to collection, 503
features, 470, 474
manipulating Items collection, 186
Insert Snippet, IntelliSense, 23, 32–33
INSERT statement, rows, 660–661
InsertRange method, collections, 503
instance methods
custom classes, 338–342
shared methods vs., 395–396
Integer values, Unicode or ANSI, 52
Integer variables, 41
IntegralHeight property, 183
integrated development environment. see IDE (integrated development environment)
IntelliSense submenu, Edit, 22–23, 23
interceptors, 937
InterestRate argument, Pmt( ) function, 132–133
interface
data-bound applications. see data-bound applications, interface design
defined, 519–520
designing modern, 12
GUI. see GUI (graphical user interface) design
Internet Explorer
script debugging, 866, 866
WebBrowser and, 816
Internet Information Services (IIS), 863, 905–906, 906
Interval methods, TimeSpan, 488
intervals, TimeSpan, 486, 487
IO namespace
Directory class in, 441, 441–442
manipulating folders and files in, 960–961
overview of, 440, 441
streaming data with, 445–446
writing data to file with, 960
I0. Directory class, 441, 441–442
I0. DirectoryInfo class, 443–444
I0. DriveInfo class, 442, 444
I0. File class, 442, 443
I0. Path class, 444, 445
irregularly shaped controls, 379, 379–382
Is operator, Minimal class, 322–323
ISBNCheckDigit( ) function, 113–115
IsContactNameNull1 method, DataRows, 709
IsDate( ) function, validating user input, 71
IsDbNull method, DataReader, 677
IsLeapYear method, 479
IsLower/IsUpper methods, 464, 464
ISNULL( ) function
DataRows, 709
selection queries, 650
T-SQL, 709
IsNumeric() function, validation, 71
IsOnDropDown property, 246
_IsPostBack() function, 879
IsPunctuation/IsSymbol/IsControl methods, 464, 465
IsWebBrowserContextMenuEnabled property, 818, 819
Item property, Items collection, 297
Items collection, 184–187, 297
Items Collection Editor, 245, 245
Items property
ComboBox control, 18, 19
defined, 185
populating control at design time, 183

J
JavaScript
AJAX, see AJAX (Asynchronous JavaScript and XML)
conforming to XHTML case syntax, 847
displaying code errors in WebBrowser, 818–819, 819
embedding into HTML pages, 861–862, 861–863
JavaScript Bible, Sixth Edition (Goodman and Morrison), 862
JavaScript Object Notation (JSON), 827
Join method, 470
join table, 730
joins, 653–656
JSON (JavaScript Object Notation), 827

K
key fields, 633
keyboard events, 861
keyboard shortcuts, 153, 818
KeyPress event, 15–16, 152, 176–177
KeyPress property, 152, 208–209
keys
capturing, 176–177
capturing function, 177–178
Dictionary collection, 510–512
foreign, 633, 633
handling, 213–214
processing from within code, 214–216
SortedList collection and, 518–519
using Access and Shortcut, 247–248
keystrokes, at form level, 152
KeyUp/KeyDown events, 177–178, 181–182, 213–214
known issues, 155

L
Label control
address visualization form, 834, 835
consuming web service, 907
LoanCalculator GUI, 134, 134
MathCalculator GUI, 145
Label3D control
Changed events, 375–376
designing, 370–374
in other projects, 378, 378
overview of, 368–369, 369
specifications, 369–370
testing, 374, 374–375
LabelEdit property, ListView, 295
lambda expressions, 557–559, 559
Language Integrated Query. see LINQ (Language Integrated Query)
language variable, 17, 17
LargeChange property, ScrollBar, 198
LargeChange property, TrackBar, 200–201
LastIndexOf method
features, 469
searching arrays, 495–496
LastIndexOf method, collections, 508–509
late-bound variables, 398–399
leaf nodes, tree structure, 287–288
left joins, SQL, 653
Left value, Alignment property, 294
LengthUnits project, 348–351
LengthUnits sample project, 351
less than (<) character, XML, 531
lifetime, variable, 73–74
lightweight web services, 827
LIKE operator, 649, 649–650
line break tag (<br>), 850
Lines property, 166
&lt;link&gt; tag, HTML, 857
LinkedDataTables project, 791–792, 791–794, 800–801
links, 850–851. see also HTML (Hypertext Markup Language)
LINQ (Language Integrated Query)
building Windows client, 930–932
components, 580–581
creating cart web form, 886–888
data binding with, 808–810, 809
to Dataset, 581
to Entities, 581, 752–754
filtering data with eSQL dynamic query vs., 977–978
overview of, 577
performing simple queries, 970
queries, 578–579
LINQ to Objects
aggregating with, 587–589
anonymous types/extension methods, 583–584
examples, 589, 589–591
grouping and aggregating, 591–593
LINQ component, 580
overview of, 581–582
querying arbitrary collections, 584–586
transforming objects, 593–597, 596
LINQ to SQL
accessing tables. see tables, accessing with LINQ
Classes component, 580–581, 615–618, 615–618
DataContext object, 618
implementing data-access layer, 697–699
processing relational data with, 971–972
retrieving data with ExecuteQuery method, 613–615
sample application, 609–613
LINQ to XML
adding dynamic content to XML, 599–601
consuming web services, 895, 895–896
LINQ component, 580
overview of, 597–598
transforming XML documents, 601–605, 602
working with XML files, 605–609, 606, 970–971
List class
adding/removing items from, 502–505
collection initializers, 505–506
creating collections, 501–502
extracting items from, 505–506
iterating through collections, 509–510
sorting collections in, 507–508
XML serialization of, 567–569
list controls, customizing, 382–385, 383
List Members, IntelliSense, 22–23, 23
list structure, ListView, 288–289, 289
List value, View property, 293
ListBox control
adding, editing and deleting contacts from, 328–330
ArraySearch application, 496, 496–497
customizing rendering of items, 958–959
for data storage/as navigational tool, 331
Demo application, 188, 188–191
designing owner-drawn, 382–385, 383
ListBoxFind application, 191, 191–193
locating item in, 947
manipulating Items collection, 184–187
overview of, 182, 182–183
properties, 183–184, 184
revising Contacts project, 325–328
selecting items, 187–188
ListBox Demo project
Add Item buttons, 189
moving items between lists, 190
overview of, 188, 188–189
removing items from two lists, 189–190
searching ListBox, 191
ListBoxFind application, 191, 191–193
lists, HTML, 851, 855–856
ListView control
Columns collection, 295–296, 296
CustomExplorer application, 299–300, 300
items and subitems, 296, 296–297
Items collection, 297
list structure, 289
nested loop example, 101
populating with data, 952
processing selected items, 299
properties, 293–294, 293–295
reference types, 392–394
SubItems collection, 298–299
tree and list structures, 288–289
with TreeView control, 286, 286–287
ListViewItem Collection Editor dialog box, 296, 296–297
Load DataSet button, 718, 719–721
Load event handler, 20, 787
Load method, EDM, 761–762
LoadFile method, RichTextBox, 274–275, 275
loading, querying Entity Data Model, 761–763
loading forms
forms vs. dialog boxes, 226–229, 227–228
multiple instances of single form, 226
MultipleForms project, 229–233, 230
overview of, 223–224
sharing variables, 225–226
Startup Form, 224–225
LoadProperty method, Windows client, 930
LoanCalculator project
building loan calculator, 130–131
designing user interface, 133–136
how it works, 131–133, 132
programming, 136–140, 137
validating data, 140, 140–144, 143
local variables, lifetime, 73, 73–74
Location property, 209
Locked property, 164
logical model, EDM structure, 739
login controls, 870, 871, 874
long menus, 249, 249
Long variables, 41
lookup columns
displaying in DataGridView, 798, 798–799
editing in hierarchical tables, 799–801, 800
loop statements
Do...Loop, 96–98
For Each...Next, 95–96
For...Next, 93–95
nested control structures, 99–102
using Exit and Continue statements, 102
While...End While, 98–99

M
Macros command, 25
Main( ) subroutine, 31
Mandatory/MandatoryColor properties, controls, 360–361
many-to-many association
between entities, 748–749
Entity Data Model, 975–976
Entity Framework, 730, 730
generating database structure, 751
many-to-many relationship
defined, 639
relating two tables with, 972
mapping, Entity Framework inheritance, 744–747
Mapping Details window, EDM diagram, 733, 733
Mapping Specification Language (MSL), EDM structure, 739
/master filename, 874
master pages, 874–875, 983
Mastering Integrated HTML and CSS (DeBolt), 848, 860
materialize, EDM, 759–760
math operations, coding, 149, 149–152
MathCalculator project, 144–158
basic debugging, 152–155
coding math operations, 149, 149–152
designing user interface, 145–146
exception handling, 155–158
overview of, 144–145, 145
programming, 147–149
MathOperator variable, 149–152, 154–155
MAX() function, aggregates, 652
Maximize command, 204, 204
MaximizeBox property, 207
MaximizeSize property, 207–208
Maximum property, ScrollBar, 197–198
MaxLength property, 162–163
MaxSize property, FontDialog, 260
MaxValue property, TimeSpan, 486, 486
MeasureItem event handler, 382–383, 958
MeasureString method, 451–452, 459–460
media, embedding in web page, 851
members
BindingSource class, 781
variable, 61
WeatherForecast class, 897–898
memory leak problem, exception handling, 156–157
menus, designing
adding/removing commands at runtime, 250–251
creating context menus, 251
creating short and long, 249, 249
menu editor, 243–246, 244–245
overview of, 243
programming menu commands, 246–247
with ToolStripMenuItem events, 949–950
with ToolStripMenuItem properties, 246
using access and shortcut keys, 247–248
menus, IDE, 22–26
Data, 25
Debug, 25
Edit, 22–24
File, 21–22
Format, 25
Help, 26
Project, 24–25
Tools, 25
View, 24
Window, 25
Menus & Toolbars tab, Toolbox window, 26
/meta tag, HTML, 849
Method attribute, HTML forms, 854
METHODS • MY.WEBSERVICE COMPONENT

methods
  custom class, 954
  data-type conversion, 62, 62–63
documenting overloaded, 124–125
  Request object, 876
  Response object, 877
  Server object, 878
  Session object, 878
  Trace object, 879
  variable, 61
WCF web service, 914–915
WebBrowser control, 821–822
WebClient class, 828–829
MetricUnits class, 348–351, 351
microformats, 863
Microsoft IIS 7 Implementation and Administration (Mueller), 905
MicrosoftReportViewer, 871

Millisecond(s) property
  DateTime class, 478
  TimeSpan value, 487

Min( ) function
  calculating aggregates, 652
  implementing arguments, 106–108
  overloaded implementations of, 119–120

Minimal class
  adding code to, 311–313
  auto-implemented properties, 319–320
  building, 308–311, 309
  custom enumerations, 323–325, 325
  customizing Equals method, 321–322
  customizing ToString method, 320–321
  implementing read-only properties, 317–318
  Is operator, 322–323
  shared variable, 341–342
  throwing exceptions, 315–316
  using property procedures, 313–314
Minimize command, Control menu, 204, 204
MinimizeBox property, 207
MinimumSize property, 207–208
Minimum property, ScrollBar, 197–198
MinSize property, FontDialog, 260
Minute(s) property
  DateTime class, 478
  TimeSpan, 487
MinValue property, TimeSpan, 486, 486
missing information, Nullable variables for, 391
mobile phones, accessing web, 846
Model Browser window, EDM diagram, 733

modifiers, class-related, 418–420
module-level scope, 72
monthly payments
  building loan calculator GUI, 132–133
  calculating using code snippet, 139–140
  programming loan calculator GUI, 136–140, 137
  validating data, 140, 140–144, 143
MonthName( ) built-in function, 112
mouse events, HTML, 861
Move command, 204, 204
Move method, 441, 442, 442–443
MoveNext method, collections, 520–522
MsgBox( ) function, 17
MSL (Mapping Specification Language), EDM structure, 739

MultiColumn property, 183–184
multidimensional arrays, 78–79, 78–81
MultiExtended property, SelectionMode, 184
Multiline property, 12–13, 13
MultiLine property, 162
multiple forms, 948–949
multiple result sets, 679–680
MultipleFiles project, 265, 265–266
MultipleForms project, 229–233, 230
MultiSelect property
  ListView control, 295
  OpenFileDialog control, 950
  OpenFileDialog/SaveFileDialog controls, 264
  SaveFileDialog control, 264
MultiSimple property, SelectionMode enumeration, 184
MustInherit keyword, 418
MustOverride keyword, 419
My component
  handling files, 959–960
  how to apply, 439–440
  overview of, 436
  subcomponents of, 436–439, 438
  using, 33–36

My.Application component, 33–34, 436
MyBase keyword, 419, 422–423
MyClass keyword, 422–423
My.Forms component, 35, 438
My.Settings component, 35, 438, 438–439
My.User component, 35, 439
My.WebService component
  adding code-behind, 909–910, 910
adding simple AJAX implementation, 910–912
adding web reference, 908–909
building, 904–905
consuming, 906–907
defined, 35
deploying, 905–906
functions of, 439
overview of, 903

N

name attribute, 〈a〉 tag, 851
Name attribute, HTML forms, 854
named arguments, 116–117
Named Pipes, 917
NamedColor property, ColorEdit control, 366–367
naming conventions
changing name of controls, 16
costants, 74
functions, 105
LoanCalculator TextBox control, 134, 134
typed DataSets, 773
variables, 37, 39–40
WCF web service, 918, 918–919
NaN value, 46–48
narrowing conversions, 64–65
Navigate method, WebBrowser, 817, 821
Navigating event, WebBrowser, 822
navigation
accessing tables with LINQ, 620–624
defined, 868
enhancing tools, 789–791
ListBox control, 331
master page, 874
overview of, 870
through DataSet, 711–714
WebBrowser events, 822
WebBrowser control, 979–981, 980
WebBrowser methods, 821–822
NegateNumber method, Minimal class, 311–313
nested control structures, 99–102
nesting classes, 344–347
New( ) subroutine, 954–955
New command, 170, 244
New Connection button, Data Source
Configuration Wizard, 770–771
New keyword
creating ArrayList class, 501–502
creating variables as true objects, 81
declaring object variables, 388–389
with Minimal class, 312–313
multiple instances of single form, 226
new order, databases, 625–628
New() procedure, 388
New Project dialog box
building console application, 30–32
choosing ASP.NET web application, 864, 865
creating new EDM, 734
starting new project, 6, 6
New() subroutine, 331–333
New Web Site dialog box
choosing ASP.NET website, 864, 865
consuming web service, 907
developing stand-alone web service, 904
setting up ASP.NET web service, 901
New Web Site option, File menu, 864
newOrder variable, 695–697
Next method, System.Random class,
adding at design time, 290–291
adding at runtime, 291–293
in tree structure, 287, 287–288
Nodes collection, 292–293
NonClickable value, HeaderStyle property, 294
nondeterministic lifetime, objects, 393
None value
DialogResult enumeration, 228
FormBorderStyle enumeration, 207, 207
HeaderStyle property, 294
NonRectangularControl project, 379–382
〈NonSerialized〉 attribute, XML, 562–563
normalization, 632
Notepad2, 848
NotInheritable keyword, 418
NotOverridable keyword, 419
Now( ) function, 53
null values
in data-driven applications, 709–710
vs. empty strings, 675
in SQL, 650
Nullable variables, OOP, 390–391
NullValue property, DataSet, 782
Num2String method, 342
Number Lines command, 170, 172
numbered lists (〈li〉...〈/li〉), 851
numbers
formatting, 65–68
in XML documents, 537–538
numeric values, dates as, 483–485
numeric variables, 41–49
  Boolean, 50
  Byte, 49
  Decimal data type, 44–46
double-precision numbers, 41–44
formatting numbers, 65–68, 66–67
Infinity and NaN, 46–49
Integer, 41
overview of, 41
Signed Byte, 49
single-precision numbers, 41–44
type characters, 45, 45
types of, 42
NWOrders application
  application’s interface, 690–691, 690–693
developing, 689
  implementing with custom objects, 693–697, 697
  implementing with LINQ to SQL, 697–699
TransactionScope class and, 699–701

Object icon, Data Source Configuration Wizard, 770, 771
Object Services, Entity Framework, 739–743
  object variables (variants)
    objects vs., 388–389
    overloading of, 352–353
    overview of, 59–60
ObjectContext class, 763–766
ObjectQuery class, 758–759
  objects
    ASP.NET, 875–876
    basic data types vs., 61
    building Minimal class, see Minimal class
classes vs., 305–308, 387–388
  constructors, 331–333
  Garbage Collector removing, 393–394
  nesting classes, 344–347
  object variables vs., 388–389
  operator overloading. see operator overloading
  overview of, 305
  serialization of individual, 571–573
StringTools class, 342–344, 344
  transforming collections of, 593–597, 596
  variables as, see variables as objects
  XML serialization of, 562–563
  Objects, LINQ to
    aggregating with, 587–589

anonymous types/extension methods, 583–584
  examples, 589, 589–591
grouping and aggregating, 591–593
Linq component, 580
overview of, 581–582
querying arbitrary collections, 584–586
transforming objects, 593–597, 596
offending operation (statement), 158
Office tab, Start page, 5
OfType operator, 753–754
OK value, DialogResult enumeration, 228, 228–229
One property, SelectionMode enumeration, 184
OneClick value, Activation property, 294
one-to-many association
  defined, 639
  Entity Data Model, 975–976
  establishing between entities, 748
online ordering application, 879–891
  creating cart web form, 886–891, 891
  creating Products web form, 880–883, 882
  creating project, 880
  creating Quantity web form, 883–886, 885
  overview of, 879
online references
  AJAX tutorial, 863
  ASP.NET AJAX Control Toolkit, 871
  Cascading Style Sheets, 860
eSQL language, 755
Expression Blend, 12
Google Maps service, 833
GUI guidelines for Windows 7 and Windows Vista, 133
HTML, 848
HTML text editors, 848
HTML text management tags, 850
IIS, 905
Internet Information Services, 905
JavaScript tutorial, 863
microformats, 863
request/response headers, 828
Yahoo! badges, 823
Yahoo! Geocoding API, 832
OnPaint method, controls, 372–373
OnQueryOrders function, 937–938
OOP (object-oriented programming)
  Class Diagram Designer, 429
  classes vs. objects, 387–388
class-related modifiers. see class-related modifiers
early vs. late binding, 398–399
extension methods, 407–411
inheritance. see inheritance
InheritanceKeywords project, 420–423
objects vs. object variables, 388–389
polymorphism. see polymorphism
properties vs. fields, 395
reference types, 392–394
shared vs. instance members, 395–396
type casting, 397–398
uninitialized and Nullable variables, 390–391
value types, 391–392
variable types, 399–400
Open command, 170, 244
Open dialog box, 265, 265–266
OpenFile method, 264–265
OpenFileDialog control
function of, 254, 254
overview of, 261, 261–262
prompting users for filenames with, 950
properties, 262–265
OpenRead method, WebClient class, 829
OpenReadAsync method, WebClient class, 829
OpenRead/OpenText methods, 442, 443
Operand1 variable, 149–152, 153–154
Operand2 variable, 149–152, 153–154
operator overloading
addition and subtraction operators, 348–351, 351
overview of, 347–348
unary operators, 351–352
variants, 352–353
optimistic concurrency, 784, 784–785
Option Explicit statement, 56
Options command, 25
Options dialog box
customizing work environment, 29–30, 29–30
setting variable-related options, 56–57, 57
order, new, 625–628
ORDER BY clause, rows, 650–651
Order class, 688
Order Details table, 637, 638
Orders table, 637, 638
Others window command, View, 24
Outlining submenu, Edit, 24
output, sending to Intermediate window, 27
Output window, 24, 27
overflow error, 142–143, 143
overloaded functions, 945
overloading operators, 118–125, 120, 123–124, 955–956
Overrideable keyword, 414–415, 419
Overrides keyword, 69, 320–321, 420
owner-drawn controls, 382–385, 383
P
padding, HTML tables, 852
PadLeft/PadRight methods, 472
Page Color command, 170
page construction, HTML, 848–849, 849
page formatting, HTML, 853–854, 854
page geometry, printing, 458–459
PageSetupDialog control, 457–458
Paint event handler, 223, 380–381, 963
painting, Framework
drawing methods, 449–453, 453
generating gradients, 453–454
generating printouts, 460, 460–462
Graphics object, 446–447
Image class, 454–455
overview of, 446
Pen/Brush classes, 449
Point/Rectangle classes, 447
Size/Color/Font classes, 448
ParamArray keyword, 115–116
Parameter class, 669–670
Parameter Info feature, IntelliSense, 23, 133
parameterized constructors, 331–333, 388
parameterized queries, 674–675
parent class keywords, 422
parent class member keywords, 419–420
parent nodes, 288
Parse method, 53–54, 477–478
passing arguments
functions returning arrays, 117–118
named arguments, 116–117
unknown number of, 115–116
by value vs. reference, 108–111
PasswordChar property, 164
passwords, typed DataSet, 772
Paste command, 170–171, 170–171, 244
Paste method, RichTextBox control, 276–277
Path class, 444, 445
PathSeparator property, TreeView, 290
PDAs, accessing web, 846
Pen class, 449
percent (%) wildcard character, 649
Period button, 146, 149
persisting data, 559
picture Box control, 218–219
pixels, creating graphics, 454–455
PlainText format, 275
Play method, My component, 33–34
Plus button, 149–150
Pmt( ) function as built-in function, 111–112
loan calculator GUI, 136–140
MathCalculator GUI, 147–149
writing and using procedures, 102–106
programming, with ADO.NET
action queries, see action queries
basic data-access classes, see data-access classes
databases, see databases
overview of, 631
SQL, see SQL (Structured Query Language)
stream- vs set-based data access, 662
progress events, 333
Project menu, 24–25
Project Properties command, 24
projects
building own classes, 952–953
creating first VB application, 13–18, 14–15
creating user friendly application, 18–19, 18–21
starting new, 5–7, 6
Projects and Solutions options, 29–30, 30
properties
adding to entities, 747–748
autocomplete, 179–180
auto-implemented, 319–320
binding hierarchical tables, 791–792
calculating values on fly, 318
column appearance, 795
count appearance, 795
control appearance, 8–11, 10
custom classes, 954
DataColumn class, 707–708
DataReader object, 676–677
DataSet state members, 712–713, 713
DateTime class, 478–479
displaying in Solution Explorer window, 26
fields vs., 314, 395
FocusedTextBox control, 363–364
Form object, 204–209, 205, 207
implementing read-only, 317–318
implementing with procedures, 313–314
items in EDM diagrams, 733, 733
ListBox/CheckedListBox/ComboBox control, 183–184, 184
ListView control, 293–294, 293–295
Request object, 876
Response object, 877
Server object, 877
Session object, 878
SqlDataAdapter object, 703, 703
TextBox control, 947
TextBox control basic, 162–164
text-manipulation, 165–166
text-selection, 167–168
TimeSpan class, 486–488
Trace object, 879
TreeView control, 289, 289–290
variable, 61
WebBrowser, 816–821
WebClient class, 827–828
Properties window (Properties Browser), 26–27
property procedures
auto-implemented properties, 319–320
defined, 954
implementing properties with, 313–314
implementing read-only properties, 317–318
throwing exceptions, 315–316
Protected/ProtectedFriend keywords, 419, 420
prototypes, 131
proxy, Windows client, 928–930, 929
Public keyword, 225–226, 419
Public statement, 75–76
Public variables, 73
Publishers table, 640
Pubs database, 633–635, 635, 638–640, 639
pushpin icon, 11

Q
Quantity.aspx web form, 883–886, 885
queries
action. see action queries
Entity Data Model, with eSQL, 754–758, 755
Entity Data Model, with LINQ to Entities, 752–754
Entity Data Model, with ObjectQuery class, 758–759
Entity Framework options, 731
selection. see selection queries
types of SQL, 643
using Query Designer of Visual Studio, 644–645
using SQL Server Management Studio, 643–644, 644
Query Builder tool, 641
Query Designer of Visual Studio, 644–645, 645
QueryInterceptor attribute, 937–938
QueryString property, WebClient class, 827
Queue collection, 519
Quick Info, IntelliSense, 23
quotation mark (”) character, XML, 531

R
dradio buttons, 856
Read method, DataReader object, 676
Read Products and Related Tables button, DataSets, 721
ReadOnly property
functions of, 164
implementing, 317–318
LoanCalculator TextBox control, 134
NamedColor property, 367
ReadOnlyChecked property, 264
Recent Projects tab, Start page, 5
RecentlyUsedList property,
AutoCompleteSource, 179
Rectangle class, 447
Redo command, 175–176, 244
Redo/RedoActionName properties, RichTextBox, 276
refactoring code, 110–111
reference
adding to remote web service, 896–897, 897
developing stand-alone web service, 908–909, 909
exploring types, 392–394
passing arguments by, 108–111
referential integrity, 640–641
Refresh Data button, 790
Refresh method, WebBrowser, 821–822
Reject Changes button, 718, 718
relational databases, 632–633, 633, 972
relationships
binding hierarchical tables, 792, 792–793
entity, 728
Entity Data Model defining, 732
establishing between entity instances, 765–766
Remove method
Controls collection, 234
features of, 471, 474
Items collection, 81, 187, 297, 504
SortedList collection, 519
Remove Selected Item(s) buttons, 189–190
RemoveAt method, 187, 710–711
Replace method, 471, 474–475
Replace/Replace All buttons, TextPad application, 172–175
reporting controls, 871
Representational State Transfer (RESTful) web
service, 832
request headers, WebClient class, 828
Request object, 875, 876
Reset method, 520–522
Resize/ResizeBegin/ResizeEnd events, 222–223
Response object, 875, 877
ResponseHeaders property, WebClient class, 828
RESTful (Representational State Transfer) web
service, 832
Restore command, Control menu, 204, 204
RestoreDirectory property, 264
result variable, 151, 154
Retry value, DialogResult property, 228, 228
return values
functions, 104–106
passing arguments, 115–118
Reverse method, Array class, 498
reverse-engineering, EDM, 767
ReverseString method, Minimal class, 311–313
Rich Text format (RTF)
basics, 270–271
RTFPad application. see RTFPad project
RichTextBoxStreamType enumeration, 274–275,
275
right joins, SQL, 653–654
Rollback method, Transaction object, 696
root nodes, tree structure, 287, 287
RootFolder property, FolderBrowserDialog, 267
RoundButton control, 379, 379–382
rows
adding and deleting DataSet, 710–711
creating HTML tables, 852
deleting, 659–660
editing existing, 661
grouping, 656–658
inserting new, 660–661
limiting data access, 937–938
navigation through DataSet, 711–712
rejecting changes in all, 718, 718
retrieving database update errors, 715–716
sorting with ORDER BY, 650–651
states and versions, 712–714, 713
viewing deleted, 804–808, 805
viewing update errors on, 789
working with DataSet, 708–709
RTF (Rich Text format)
basics, 270–271
RTFPad application. see RTFPad project
RTFPad project
Edit menu, 281–283
File menu, 279–281
Format menu, 283–284
overview of, 277–279
Search & Replace dialog box, 284, 284–286
runtime
adding nodes at, 291–293
building dynamic forms at. see dynamic forms
creating event handlers at, 241–242
DynamicForm project and, 236–239, 237
manipulating menus at, 248–251
S
Save button, toolbar, 787
Save DataSet button, 718, 719
SaveChanges method, 764–766, 933, 936
SaveContact function, 329–330
SaveFileDialog method, 264–265, 274
SaveFileDialog control
function of, 255
overview of, 261, 261–262
prompting users for filenames, 950
SaveOrder method, 695–696
Save/Save As commands, 170, 244
SByte data type, 42, 51
scalar property, 735–736
scancode, 177
schema, XML document, 534–537, 537
scope (visibility)
  constants, 74
  variables, 71–73, 72
<script> tag, HTML, 851
<...script> tag, HTML, 851
ScriptErrorsSuppressed property, WebBrowser, 818, 819
scripts, inserting non-HTML code into pages, 851
Scroll event, 223
Scroll method, 206
Scrollable property, ListView, 295
ScrollBar control
  Colors application, 198, 198
  events, 199–200
  function of, 197–198
  overview of, 197
  providing visual feedback to user, 947–948
ScrollBars property, 12, 163
ScrollBarsEnabled property, WebBrowser, 818, 823
Seago font, 12, 134
Search and Replace operations
  RTFPad project, 284, 284–286
  TextPad application, 172–175
Second(s) property
  DateTime class, 478
  TimeSpan value, 487
security
  ADO.NET Data Services, 937–939
  WebBrowser and, 816
Select All command, 170, 244
Select Case statement, 90–91
SELECT statement
  Data Source Configuration Wizard, 783, 783
  querying databases with SQL, 972–973
  for selection queries. see selection queries
<select> tag, HTML, 855–856
SelectedIndex property, ComboBox control, 20
SelectedListItemCollection collection, ListView, 299
SelectedPath property, FolderBrowserDialog, 267
SelectedText property
  functions of, 167
  RichTextBox, 271, 272
selection list, 646
selection queries
  aliasing column names, 648–649
  aliasing table names, 647–648
  calculating aggregates, 651–653, 652
  combining data from multiple tables, 647
  executing with Command class, 667–668
  grouping rows, 656–658
  LIKE operator, 649, 649–650
  limiting with WHERE, 646
  not joining tables with WHERE clause, 655–656
  null values and ISNULL() function, 650
  SELECT statement in, 645–646
  skipping duplicates with DISTINCT, 649
  sorting rows with ORDER BY, 650–651
  submitting queries to database in ADO.NET, 973
  using SQL joins, 653–654
  working with calculated fields, 651
SelectionAlignment property, RichTextBox, 271, 272
SelectionBullet property, RichTextBox, 271, 273
SelectionHangingIndent property, RichTextBox, 271, 272–273, 273
SelectionIndent property, RichTextBox, 271, 272–273, 273
SelectionLength property, 167–168, 271, 272
SelectionMode property, 184, 184, 187–188
SelectionRightIndent property, RichTextBox, 271, 272–273, 273
SelectionStart property, 167–168, 271, 272
SelectionTabs property, RichTextBox, 271, 273
selectors, 409–410
Select/SelectAll methods, RichTextBox, 275
sender argument, event handlers, 16
serialization
  of collections, 573–575
  of collections of objects, 567–569
  of custom objects, 563–567
  defined, 330
  of individual objects, 562–563
  other serialization types, 560
  overview of, 559
  process of, 560–561
  types, 560, 569–571
  with XMLSerializer class, 969
Serialize method, 560–561
server, ADO.NET Data Service, 923, 923
Server Explorer tool, 641
Server object
defined, 875
methods, 878
properties, 877
server-side
maintaining state, 873–874
technologies, 863–864
Service option, Data Source Configuration Wizard, 770, 771
Session object, 875, 878
session variables, 873, 874
Set section
auto-implemented properties, 319–320
ColorEdit control, 366–367
property procedures, 313–314, 395
set-based data access, 662
Set-Cookie response header, 828
SetColorDirectory method, 441, 442
SetEntitySetAccessRule method, 937
SetRange method, 504
SetServiceOeprationAccess Rule method, 937
Shape classes, 413, 413–417
Shared Members sample project, 339–340
shared methods
custom classes, 338–342
instance methods vs., 395–396
SharePoint tab, Start page, 5
Shippers table, Northwind database, 638
short menus, 249, 249
Short variables, 41
short-circuiting expression evaluation, 91–93
shortcut keys, 247, 248
Show Deleted Rows button, 804–807, 805
Show Edits button, 717–718, 718, 722–723
Show navigation methods, WebBrowser, 822, 822
ShowApply property, FontDialog, 260
ShowColor property, FontDialog, 259
ShowColor() subroutine, ColorEdit, 366–368
ShowControls project, 234–236, 235
ShowDataSet( ) subroutine, 721
ShowDialog method, 950–951
ShowEffects property, FontDialog, 260–261
ShowLines property, TreeView, 290
ShowNewFolderButton property,
FolderBrowserDialog, 267
ShowPlusMinus property, TreeView, 290
ShowReadOnly property,
OpenFileDialog/SaveFileDialog, 264
ShowRootLines property, TreeView, 290
Signed Byte (SByte) data type, 42, 51
SilverLight applications, 894
Simple ComboBox style, 193–194, 193–195
SimpleClass project, 333–335, 334–335
SimpleDataSet project, 717–723
populating and displaying DataSet, 721
populating DataSet at client, 717, 718
saving and loading DataSet, 719, 719
setting up DataAdapters, 720–721
viewing changes in client DataSet, 717, 718
viewing edited rows, 722–723
viewing rows failing to update, 719–720, 720
single data type, 41–44
single quotes, 674–675
single-precision numbers, 41–44
SizableSizableToolWindow values, 207
of FormBorderStyle enumeration, 207
Size class, graphics, 448
Size command, Control menu, 204, 204
Size property, 209
SizeGripStyle property, 209
Skip/SkipWhile methods, LINQ, 589
SmallChange property
ScrollBar, 198
TrackBar, 200–201
SmallIcon value, View property, 293
snap lines, 14, 145–146
SnapToGrid value, Alignment property, 294
snippets, see code snippets
SOAP (Simple Object Access Protocol)
ASP.NET web services, 899–900
Discovery, 900
serialization, 560, 569–571
SoapFormatter method, 570, 572–573
Social Security number, as primary key, 726–727
SolidColorOnly property, ColorDialog, 258
Solution Explorer window, 26, 908
solutions, starting new project, 6
Sort method, 494, 507–509
Sorted property, 184, 290
SortedList collection, 518–519
Sorting property, ListView, 295
special characters
handling, 674–675
text management, 850
Split method, 470–471, 514–516
SplitContainer control, 219–221, 220–221
SQL (Structured Query Language)
  data types, 677–678
  executing statements, 643–645, 644–645
  overview of, 642–643
  processing relational data with LINQ.
  see LINQ to SQL
  querying databases with, 972
  selection queries. see selection queries
  wildcard characters, 649, 649–650
SQL, LINQ to
  accessing tables. see tables, accessing with LINQ
  DataContext object, 618
  as LINQ component, 580–581
  LINQ to SQL classes, 615–618
  retrieving data with ExecuteQuery method, 613–615
  sample application, 609–613
SQL Server
  extracting data in XML format from, 605–609, 606
  Management Studio, 643–644, 644
  NWOrders application, 690
  SQL injection attacks against, 673–674
SQL Server Profiler, 751
SQL statement, Data Source Configuration Wizard, 783, 783
SqlConnection object, 973
src attribute, <img> tag, 850
SSDL (Store Schema Definition Language), 739
Stack collection, 519
stand-alone web services, 903–910
  adding code-behind, 909–910, 910
  adding web reference, 908–909, 909
  building MyWebService, 904–905
  consuming MyWebService, 906–907, 907
  deploying MyWebService, 905–906, 906
  overview of, 903
  real-world scenario, 906
Standard (web form) controls
  creating web applications, 867, 867–868
  defined, 867–868
  overview of, 867
Standard value, Activation property, 294
Start Page, 4, 4–6
  start variable, For...Next loops, 93–95
  StartPosition property, 209
  StartsWith method, 468
Startup Form combo box, 224–225
state, maintaining, 871, 873–874
statements, executing, 27
statements block, exception handling, 156–157
states, row, 712–713, 713
Static keyword, preserving variable lifetime, 73, 73–74
Statistics( ) function, 117–118
Stock Quotes project, 823–825
stocking products, 344–345
Stop method, WebBrowser, 821–822
StopWatch class, 489–490
storage
  data. see DataSets
  EDM schema model, 732
  Store Schema Definition Language (SSDL), 739
  stored procedures, 667, 670–673
  stream-based data access, 662
  streaming, 445–446
  StreamWriter object, 960
  Strict option, 55–59, 942
  strictly typed variables, 57–58
String class
  methods, 467–472
  overview of, 466
  properties, 466–467
String Collection Editor dialog box, 19, 19
StringBuilder class
  efficiency of, 475–476
  manipulating large or dynamic strings, 961–962
  overview of, 472–473
  properties and methods, 473–475
strings and characters
  Char data type, 463–466, 464–465, 961
  overview of, 463
  String class, 466–472
  StringBuilder class, 472–476, 961–962
StringTools class, 342–344, 344
strProperty, Minimal class, 311–313
structured exception handlers, 156, 946
structures
  creating custom data types, 68, 68–70, 70, 942–943
  using array of, 78
  <style> tag, CSS, 857–858
styles
  formatting with CSS, 857–858
  HTML text management tags, 849–850
Sub...End Sub statements, subroutines, 103–104
Submit Edits button, 718, 719–720, 723
SubmitChanges method, TransactionScope, 699–700
subroutines
accepting arguments. see arguments functions gradually replacing, 105
functions vs., 103
passing arguments to, 945
writing and using, 103–104, 944
Subtract method
DateTime, 480
TimeSpan, 489
Sum extension method, 409–410
SUM() function, aggregates, 652
Suppliers table, Northwind database, 636, 636
SubmitChanges method, TransactionScope, 699–700
Submit Edits button, 718, 719–720, 723
SubmitChanges method, TransactionScope, 699–700
subroutines accepting arguments. see arguments functions gradually replacing, 105
functions vs., 103
passing arguments to, 945
writing and using, 103–104, 944
Subtract method
DateTime, 480
TimeSpan, 489
Sum extension method, 409–410
SUM() function, aggregates, 652
Suppliers table, Northwind database, 636, 636
SubmitChanges method, TransactionScope, 699–700
Submit Edits button, 718, 719–720, 723
SubmitChanges method, TransactionScope, 699–700
subroutines accepting arguments. see arguments functions gradually replacing, 105
functions vs., 103
passing arguments to, 945
writing and using, 103–104, 944
Subtract method
DateTime, 480
TimeSpan, 489
Sum extension method, 409–410
SUM() function, aggregates, 652
Suppliers table, Northwind database, 636, 636
SubmitChanges method, TransactionScope, 699–700
Submit Edits button, 718, 719–720, 723
SubmitChanges method, TransactionScope, 699–700
Subtract method
DateTime, 480
TimeSpan, 489
Sum extension method, 409–410
SUM() function, aggregates, 652
Suppliers table, Northwind database, 636, 636
SubmitChanges method, TransactionScope, 699–700
Submit Edits button, 718, 719–720, 723
SubmitChanges method, TransactionScope, 699–700
Subtract method
DateTime, 480
TimeSpan, 489
Sum extension method, 409–410
SUM() function, aggregates, 652
Suppliers table, Northwind database, 636, 636
SubmitChanges method, TransactionScope, 699–700
Submit Edits button, 718, 719–720, 723
SubmitChanges method, TransactionScope, 699–700
Subtract method
DateTime, 480
TimeSpan, 489
Sum extension method, 409–410
SUM() function, aggregates, 652
Suppliers table, Northwind database, 636, 636
SubmitChanges method, TransactionScope, 699–700
Submit Edits button, 718, 719–720, 723
SubmitChanges method, TransactionScope, 699–700
TableAdapters, 774–776, 788
table-per-hierarchy inheritance, 745
table-per-type inheritance mapping, 744–745
tables
accessing DataSet, 707–708
adding ADO.NET Data Service, 922, 922–923
adding and deleting rows, 710–711
aliasing names in, 647–648
binding hierarchical, 791–792, 791–794
combining data from multiple, 647
editing data in hierarchical, 799–801, 800
establishing relations between DataSet, 974
formatting pages in HTML, 854
generating typed DataSet, 773, 774
HTML, 852–853, 853
navigation through DataSet, 711–714, 713
not joining with WHERE clause, 655–656
rejecting changes in all, 718, 718
working with rows, 708–709
tables, accessing with LINQ, 632–633
creating a new order, 625–628, 627
navigation methods, 620–624
overview of, 619–620
updates, 624–625, 625
Tables Designer tool, 641
tags
applying styles to HTML, 857
HTML page construction, 848–849
HTML text management, 849–850
inserting dynamic content, 599–600
table, 852
transforming XML documents, 601–605, 602
XHTML, 846–847
XML documents, 530–534
Take(N), TakeWhile(Expression) method, 588
target attribute, <a> tag, 851
<td> tag, HTML, 852
<td>...</td> tags, HTML, 852
testing
ASP.NET web service, 901–902
ColorEdit control, 368
custom comparers, 526–527
custom Windows controls, 374, 374–375
FocusedTextBox control, 362, 362–363
for Infinity or NaN value, 48–49
Shape classes, 416–417
short code segments with console applications, 32
StringTools class, 343–344, 344
using Boolean variables in, 50
WCF web service, 916, 919–920, 920
web services, 905, 906
text
formatting on web. see HTML (Hypertext Markup Language)
managing with HTML tags, 849–850
manipulation properties, 165–166
selection properties, 167–168
String data type storing only, 50–51
Text Color command, 170
Text control, 855
text editors, 163–164, 848
Text property
ComboBox, 21
functions of, 165–166
ListBox/CheckedListBox/ComboBox, 184
TextAlign property, 162
<textarea> tag, HTML, 855
TextBox control
  autocomplete properties, 179–180, 179–180
  basic properties, 162–164
  capturing function keys, 177–178
  capturing keystrokes, 176–177
  data entry with, 181, 181–182, 946–947
  designing address visualization form, 834, 835
  designing dynamic forms, 949
  designing LoanCalculator GUI, 134, 134–135
  designing Windows applications, 130
  enhancing, 401–402
  NWOrders application UI, 690–692
  overview of, 161, 162
  replacing Button controls on toolbar with, 790
  as text editor, 163–164
  text-manipulation properties, 165–166
  TextPad project. see TextPad application
text-selection properties, 167–168
  undoing edits, 168
  Undo/Redo commands, 175–176
  writing contents to file with IO namespace, 960
TextChanged event, dynamic forms, 949
TextPad application
  editing commands, 169–171, 170–171
  overview of, 168–169, 169
  Process and Format menus, 171–172
  Search and Replace, 172–175
TextTextOLEObjs format, 275
<th> tag, HTML, 852
<th>... </th> tags, HTML, 852
thin client, 894
TickFrequency property, Inches application, 200–201
Ticks property
  DateTime, 479
  TimeSpan, 487
  TicksPerDay/TicksPerHour properties, 486, 486
Title value, View property, 293
time events, 333
TimeOfDay property, 478
TimeSpan class
  defined, 53–54
  methods, 488–489, 489
  overloading operators in, 347–348
  overview of, 485–486
  properties, 486–488
StopWatch class, 489–490
timing operation with, 962
<title> tag, HTML, 849
Title/TitleAuthor tables, 638, 639, 639
Today( ) function, 53
ToUpper/ToLowerCase methods, 464, 465
tools
  designing data-driven interfaces, 787–787
  enhancing navigation, 789–791
  toggling IDE, 7
Toolbox, Windows Forms, 7–8, 8, 22
Toolbox window, 26, 733
Tools menu, 25
ToolMenuStripItem class, 246
ToolMenuStripItem object, 949–950
Top value, Alignment property, 294
TopMost property, 209
ToString method
  Char data type, 464, 465
  customizing, 320–321
  DateTime class, 480
  features of, 475
  formatting numbers, 65–66, 66
  overriding default implementation of, 69
  standard numeric format strings, 66, 66–67
  TotalSize/TotalFreeSize properties, 442, 444
  ToTraceString method, ObjectQuery, 758–759
<tr> tag, HTML, 852
<tr>... </tr> tags, HTML, 852
Trace object, 875, 879
TrackBar control
  defined, 947
  function of, 200
  Inches application, 200–201
  overview of, 197
transactions
  ADO.NET Data Services, 934–936
  implementing with custom objects, 693–697, 697
  implementing with LINQ to SQL, 697–699
  implementing with TransactionScope, 699–701
  using in Entity Framework, 766
TransactionScope class, 699–701, 766
tree structure, TreeView, 287–288, 287–288
TreeNode Editor dialog box, TreeView, 290–291, 291
TreeView control
  adding nodes at design time, 290–291, 291, 952
  adding nodes at runtime, 291–293
CustomExplorer application, 299–300, 300
  customizing, 382–383
  with ListView control, 286, 286–287
  properties, 289, 289–290
  tree structure in, 287–288, 287–288
TryCast() function, 397, 399
  TwoClick value, Activation property, 294
type casting, 397–398, 417
type characters, 45, 45
type inference, 58–59
typed collections, 501
typed DataSets, 769–778
  basic operations, 774–776, 775
  binding Windows forms to, 977
  creating, 976–977
  filtering data with LINQ query vs. eSQL dynamic query, 977–978
generating, 770–774, 771–774
  iterating through linked DataTables, 776, 776–777
  manipulating rows of typed DataTables, 777, 777–778
  untyped vs., 702, 769–770

U
UDDI (Universal Description, Discovery, and Integration), ASP.NET, 900
UI layer, 692
  unary operators, 351–352
  underscore (_) wildcard character, 649
  Undo command, 170, 175–176, 244
  Undo/UndoActionName properties, RichTextBox, 276
Unicode, Integer values, 52
  UnicodePlainText format, 275
  uninitialized variables, OOP, 390–391
Universal Description, Discovery, and Integration (UDDI), ASP.NET, 900
  untyped collections, 501
  untyped DataSets, 702
  Update method, 778
  Update method, DataAdapter, 703, 715
  Update rule, foreign key constraints, 792, 793
  UPDATE statement, 661, 784, 784, 784
  UpdateRule property, ChildKeyConstraint, 716–717, 717
updates
  accessing tables with LINQ, 624–625, 625
  ADO.NET Data Services, 932–933
  with DataAdapter, 715–716
  entity instances, 765
  handling Identity columns, 716–717, 717
  hierarchical DataSet, 800–801
  overview of, 714–715
  submitting changes in DataSet, 974–975
  viewing errors on form control, 787–789, 789
Upload methods, WebClient class, 829
  Upload*Async methods, WebClient class, 829
  Upload*Completed events, WebClient class, 829
URL (Uniform Resource Identifier), 817
  configuring ADO.NET Data Service, 923–925, 924
  filtering data through, ADO.NET Data Service, 926–928, 927
  incorporating state information in, 873
URL property, WebBrowser, 817, 824
  URL query, 832–833
  usemap attribute, <img> tag, 850
user-defined data types, 68, 68–70, 70
user-drawn controls
  Changed events in, 375–376
  control specifications, 369–370
  creating, 957–958
  designing, 370–374
  overview of, 368–369, 369
  testing, 374, 374–375
  using in other projects, 378, 378

V
validating data
  HTML forms, 820–821
  LoanCalculator project, 140, 140–144, 143
  writing well-behaved applications, 142
validation controls, 868, 869
Value property, ScrollBar, 197–198
value types, 391–392
values
  dates as numeric, 483–485
  DialogResult property, 228, 228–229
  Dictionary collection, 510–512
  editing DataTable rows, 708–709
  FormBorderStyle property, 207, 207
  handling Identity columns, 716–717, 717
  multiple, retrieving from stored procedures, 670–673
  null, 650, 675
  Parameter class, 669–670
variables

arrays, see arrays
Boolean, 50
Byte, 49
coder, 51–53
coding math operations, 149, 149–152
constants, 74
creating first VB application, 17, 17
date, 53–55
declaring, 38–40
declaring and using, 942
form-wide, 150
initialization, 40
naming conventions, 39–40
numeric, see numeric variables
object, 59–60
Object, 388–389
overview of, 37–38
sharing public, 225–226
Strict, Explicit, and Infer options, 55–59, 942
string, 50–51
uninitialized and Nullable, 390–391
value, 391–392
variables as objects, 60–74
basic data types vs. objects, 61
casting variable types, 397–398
converting variable types, 62, 62–65, 64
discovering variable types, 399–400
early vs. late binding of, 398–399
examining variable types, 70–71
formatting numbers, 65–68, 66–67
lifetime, 73–74, 73–74
overview of, 60–62
scope, 71–73, 72
user-defined data types, 68, 68–70, 70
variants, see object variables (variants)
VB axis properties, 544–545
Verdana font, 12, 134
versions, row, 713–714
vertical partitioning, 732
vertical resolution, Image class, 454
View menu, 24
View property, ListView, 293, 293
view state, 864

VARIABLES • WEB SERVICES 1021

ViewAddressOnAMap project, see address
visualization form
ViewState settings, 874
Visible property, 246
visual database tools, 641, 641
VolumeLabel property, 442, 444

W

WCF (Windows Communication Foundation) web services, 912–920
building, 912–916, 985
configuring, 916–920, 917–920
defined, 894
overview of, 899, 912
WCF Test Client Utility, 916, 919–920, 920
WcfSvcHost utility, 916
weather forecast web services, 895, 895–898, 897–898
WeatherData property, 897–898
WeatherImage property, 898
WeatherService project, 896–898, 897
Web access, 815–843
address visualization, see address visualization form
HttpWebRequest and HttpWebResponse classes, 831
using Fiddler, 826
WebBrowser control, 816–821, 979–981
WebBrowser control methods, 821–822
WebClient class, 827–831, 978
Web applications, creating, 845–892, see also online
ordering application
AJAX, 863
with ASP.NET intrinsic objects, 983–984
ASP.NET objects, 873–879, 876–879
Cascading Style Sheets, 856–860, 982
controls, 867–871, 867–871
developing for web, 845–846
JavaScript, 861–862, 861–863
maintaining state, 871–874, 872–873
master page, 874–875, 983
microformats, 863
overview of, 864–866, 865–866
postback, 879
server-side technologies, 863–864
understanding HTML and XHTML, 846–847
working with HTML, see HTML (Hypertext
Markup Language)
web services, 893–938
ADO.NET Data Services. see ADO.NET Data Services
AJAX implementation, 910–912
ASP.NET, 898–903, 902–903, 984
composing address visualization form, 832–834
consuming, 894–898, 895, 897–898
consuming ASP.NET, 902–903, 984
technologies associated with, 899–900
types of, 893–894
understanding, 894
using HTTP to transport data in lightweight, 827
WCF, 899, 912–920, 917–920, 985
Web Site Administration tool, 870, 870
Web tab, Start page, 5
WebBrowser control
adding functionality of, 979–981, 980
events, 822
methods, 821–822
overview of, 816
properties, 816–821
Stock Quotes project, 823–825
WebBrowserShortcutsEnabled property, 818
WebClient class
accessing Web with, 827–831, 978–979, 979
asynchronous download, 830–831
defined, 827
events, 829
methods, 828–829
properties, 827–828
Web.config file, 866
WebParts controls, 870, 872
WebRequest object, 896
WebResponse object, 896
WeekDayName( ) function, 112–113
Welcome tab, Start page, 5
WHERE clause, 646, 655–656
Where extension method, 409–410, 557–558
Where operator, 931
white space, 22, 850
widening conversions, 64, 64–65
width attribute, <img> tag, 850
wildcard characters, 649, 649–650
window events, HTML, 862
Window menu, 25
windows, rearranging IDE, 9
Windows applications
basics, 941
console applications vs., 31
consuming web services, 898, 898
designing, 129–130, 941–942
Windows authentication, 772
Windows Communication Foundation. see WCF (Windows Communication Foundation) web services
Windows controls
anchoring, 216–217, 216–217
Checked ListBox. see CheckedListBox control
ComboBox. see ComboBox control
common dialog. see dialog controls
docking, 216, 217–218, 218
ListBox. see ListBox control
ListView. see ListView control
overview of, 161, 253
placing on forms, 209–210, 210
RichTextBox. see RichTextBox control
ScrollBar, 197–200
TextBox. see TextBox control
TrackBar, 197, 200, 200–201
TreeView. see TreeView control
Windows controls, building custom compound controls, 364–368, 956
customizing list controls, 382–385, 383
irregularly shaped, 379, 379–382
overview of, 955–956
raising custom events, 377–378
rendering of items, 958
user-drawn. see user-drawn controls
user-drawn controls. see user-drawn controls
Windows controls, enhanced
adding functionality, 359–362, 956
building FocusedTextBox control, 357, 357–359, 359
classifying properties, 363–364
overview of, 355–357
testing FocusedTextBox control, 362, 362–363
Windows Form Designer
building interface, 7–13, 8–11, 13
first VB application, 13, 15
Windows forms. see forms
Windows tab, Start page, 5
With statement, My component, 34
word wrap, 22, 170
WordWrap property, 12, 163, 364–365
WSDL (Web Services Description Language), 900
X
XAttribute, 538–539, 545–546
XComment, 538
x-coordinate, Point class, 447
XDocument, 538
XElement class
  accessing contents of XML document, 968–969
  editing XML documents, 545–546, 549–550
  Element method and, 543
  function of, 538–539
  saving/loading XML documents and, 542–543
  using lambda expressions, 557–559, 559
XHTML (extensible HTML), 846–847
XML (Extensible Markup Language)
  accessing contents of document, 968–969
  consuming web services, 895, 895–896
  creating and processing files with LINQ. see LINQ to XML
  creating documents, 967–968
  as data type, 540, 540–542
  editing XML documents, 545–546
  EDM structure, 736–738
  introduction to, 530, 530–534
  LINQ to XML component. see LINQ to XML
  manipulating, see XML data manipulation
  numbers and dates, 537–538
  overview of, 529
  saving and loading, 542, 542–543
  schema, 534–537, 537
  serialization. see XML serialization
  transforming collections into, 596–597
  traversing, 543–545
  web services. see ASP.NET web service
XML data manipulation
  editing documents, 549–551
  locating information, 547–549
  overview of, 546–547
  using lambda expressions, 557–559, 559
  using segments as literals, 551–557, 554
XML serialization
  of collections of objects, 567–569
  of custom objects, 563–567
  of individual objects, 562–563
  other serialization types, 560
  overview of, 559, 969
  process of, 560–561
XMLSerializer class, 560–563, 568–569, 969
Y
Yahoo! badges, 823–825, 825
Yahoo! Geocoding API
  constructing URL and query parameters, 834–836
  implementing address visualization form, 832–833
  invoking web service, 836–837
  processing response, 837–838
y-coordinate, Point class, 447
Yes value, DialogResult, 228, 228
Z
Zero property, TimeSpans, 486, 486