

# Index

---

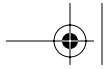
## Symbols

- <%#...%> syntax, 508–509
- ? (question mark), 140
- + (plus sign), 202

## A

- Absolute positioning, controls, 589–590
- AcceptChanges method
  - client-side transactions, 650–651
  - EndEdit method vs., 158
  - setting row state in data tables, 641
- accessDataSource control, 507
- Add Column dialog, 264–265, 276–277
- Add Connection dialog
  - adding new data sources and connections, 7–8
  - overview of, 182–183
  - setting up typed data set connections, 45–46
- Add method
  - adding columns to TextBox cell, 223
  - adding controls to Windows Forms, 552
  - adding rows to grids, 224
  - binding image columns to PictureBox, 142
  - binding numeric columns to TextBox, 145–147
  - IList interface, 300–301
  - inserting items in BindingSource list, 124–126
  - simple data binding, 91
- Add Query Wizard, 75–77
- add tag, connectionStrings, 58
- Add Web Reference dialog, 185–186
- AddBarChartControl method, 389–391
- AddControlToGrid method, 531
- AddCopies method
  - adding rows in virtual mode, 234
  - initializing grid, 238
  - working with unbound columns, 231
- AddCustomers method, 425
- AddHandler keyword, 575
- AddIndex method, IBindingList, 312
- AddingNew event, 125–126, 131
- AddNew method
  - creating custom collection type, 427
  - IBindingList, 312–313
  - ICancelAddNew interface, 426
  - using BindingSource as data storage container, 124–126
- AddNewCore method, 427–428, 441
- AddValueChanged method, 307
- AddWithKey property, 622
- ADO.NET, 601–670
  - client-side transactions, 650–651
  - creating data sets programmatically, 611–613
  - data set and data adapter events, 651–654
  - DataSet structure, 607–609
  - handling concurrency, 628–631
  - loading data into XPathDocument, 664–665
  - loading data sets from database, 613–619
  - loading data sets from files, 609–611





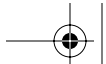
672 **INDEX**

- ADO.NET, *continued*
  - loading DataTable with DataReader, 619–621
  - master-details DataSets, 621–623
  - merging data from multiple data sets, 639–641
  - navigating XML document, 667–669
  - overview of, 601–603
  - querying XML data, 665–667
  - reading data into business objects, 654–658
  - relational data access, 603–607
  - retrieving data with stored procedures, 623–624
  - scoping transactions with
    - System.Transactions, 647–649
  - searching data sets, 637–638
  - updating database using data sets, 624–628
  - updating with data sets and stored procedures, 632–637
  - working with data views, 641–642
  - working with SmlDataDocument class, 659–663
  - working with transactions, 643–646
  - working with XPathDocument class, 663–664
  - XML data access, 658–659
- Aliases, 78–79
- AllCells value, DataGridViewAutoSizeMode, 260
- AllowEdit property, IBindingList, 133, 311, 313
- AllowNew property, IBindingList, 133, 311, 313
- AllowPaging property, DataGridView control, 511
- AllowRemove property, IBindingList, 133, 311, 313
- AllowUserToAddRows property, 221, 265
- AllowUserToDeleteRows property, 221, 265
- AllowUserToOrderColumns property, 266–269
- AlternatingRowsDefaultCellStyle property, 282
- Anchor property, controls, 590–592
- app.config, 58, 60
- Application class, defined, 572
- Application class, XAML, 538–539
- Application object, WinFx, 525
- ApplySort method
  - creating rich object collections, 426
  - IBindingList, 312, 317–318
  - IBindingListView, 324, 444
  - ApplySortCore method, 437
  - ApplySortInternal method, 437
- Architecture, Windows Forms, 567–570
- ArgumentException, 611
- ASP.NET, 487–519
  - data binding expressions, 508–509
  - data binding in ASP.NET 2.0, 498–499
  - DetailsView control, 512–514
  - FormView control, 514–515
  - GridView control, 509–512
  - hierarchical binding, 518
  - master-details data binding, 515–518
  - overview of, 487–488
  - page processing basics, 489
  - ASP.NET 1.X, 490–498
    - hooking up dynamic behavior in code-behind class, 492–495
    - overview of, 490–491
    - presenting tabular data in grid, 491–492
    - row selection in grid, 495–496
    - using expressions, 496–498
  - ASP.NET data sources, 499–508
    - AccessDataSource control, 507
    - ObjectDataSource control, 505–507
    - overview of, 499–500
    - SiteMapDataSource control, 508
    - SqlDataSource control, 500–505
    - XmlDataSource control, 507–508
- ASPX files, 489, 492–495
- Attributes
  - designer initialization of data binding, 359–360
  - specifying binding properties on controls, 360–362
  - XPathNavigator navigation methods, 669
- Authentication, 8–9
- AutoComplete feature
  - DataGridComboBoxColumn, 252
  - filter grid controls, 354–355
  - overview of, 160–162
  - TextBox control, 371–375
- AutoGenerateColumns property, 358
- Automatic formatting, 147–148
- AutoSizeMode property
  - DataGridView control, 259–261
  - DataGridViewColumn, 246, 375–376
- AutoValidate property, 461–462, 471–472
- Avalon code name, 521



**B**

- BackgroundWorker, 239
- BarChartControl
  - adding editing support to, 391–397
  - creating, 381–383
  - designing, 379–381
  - rendering bars, 383–384
- BarChartControl, coding, 384–391
  - defining custom control skeleton, 385–386
  - initializing and updating data bindings, 387–389
  - overview of, 384–385
  - using AddBarChartControl method, 389–391
- Batching, in database updates, 33
- BeginEdit method
  - binding controls to data sources, 157
  - IEditableObject interface, 329
  - transacted editing of objects, 417–420
- BeginInit method, ISupportInitialize, 335–337
- BeginTransaction method, table adapters, 64–65
- Bind method, ASP.NET 2.0, 509
- BindFilterFields method, 368
- Binding object
  - binding DateTime column to DateTimePicker, 142–144
  - binding DateTime column to TextBox, 144–145
  - binding image column to PictureBox, 141–142
  - overview of, 135–137
  - simple data binding with, 83–84, 90–92
- BindingComplete event, 131
- BindingContext, 20
- BindingContext property, 86, 96–97
- BindingEvents application. *See* Events, binding
- BindingList<T> class
  - creating rich object collections, 424–426
  - defined, 287
  - as generic collection class, 408
  - managing transacted additions to collections, 439–441
  - object edit notifications, 423–424
  - raised item changed events, 441–443
- BindingList<T> class, custom collections, 426–439
  - adding search functionality, 430–431
  - adding sorting capabilities, 432–439
  - getting test data, 429–430
  - overview of, 426–427
  - taking over construction process, 427–428
- BindingListView<T> class
  - adding search functionality, 430–431
  - adding sorting functionality, 435–437
  - using IBindingListView functionality, 445–453
- BindingNavigator control
  - adding navigation to control with, 11
  - defined, 151–152
  - master-details data binding with, 105–106
  - overview of, 103–104
- BindingSource component
  - data binding with, 97–99, 112–115, 219–220
  - defined, 15
  - events, 131–133
  - generating data-bound controls, 11, 195–196
  - implementing ITypedList with, 307–309
  - master-details data binding with, 105–106, 116–121
  - navigating data with, 121–122
  - overview of, 111–112
  - paging through data with, 99–104
  - populating with data reader, 126–128
  - position changes in, 365–366
  - restricting changes to data with, 133
  - sorting, searching, and filtering data with, 128–130
  - WinFx binding vs., 532
- BlinkRate properties, ErrorProvider, 466
- BlinkStyle properties, ErrorProvider, 466
- bool Contains method, IList, 300
- Borders, in DataGridView, 282–283
- Bound controls. *See* Data-bound controls
- Built-in columns, DataGridView, 241–255
  - DataGridViewButtonColumn, 250
  - DataGridViewCell properties, 242–246
  - DataGridViewCheckBoxColumn, 251
  - DataGridViewColumn properties, 246–249
  - DataGridViewComboBoxColumn, 252–255
  - DataGridViewImageColumn, 251–252
  - DataGridViewLinkColumn, 250
  - DataGridViewTextBoxColumn, 249–250
  - overview of, 241–242
- Business logic layer, 21–27, 459



Business objects, 399–456  
 adding `IBindingListView` to, 443–453  
 CustomBusinessObjects example. *See* CustomBusinessObjects example  
 defining and working with, 400–405  
 defining and working with collections, 405–406  
 edit notifications with  
     `INotifyPropertyChanged`, 423–424  
 edit notifications with property change events, 420–423  
 generic collection classes, 406–408  
 managing transacted additions to collections, 439–441  
 overview of, 399–400  
 raising item changed events, 441–443  
 reading data into, 654–658  
 setting textual data-binding behavior of, 415–416  
 through Data Sources window, 453–455  
 transacted object editing and, 416–420  
 using `BindingList<T>`. *See* `BindingList<T>` class  
 using `IDataErrorInfo`, 479–482

Button control  
 adding to Windows Forms, 550–552  
 adding to WinFx applications, 530–531  
 overview of, 580  
 using `DataGridViewButtonColumn`, 250

ButtonColumn, 495–496

Byte arrays, 141

## C

`CacheDisplayOrder` method, columns, 269

`CancelEdit` method, `IEditableObject`  
 binding controls to data sources, 157  
 overview of, 329  
 transacted additions to collections, 439–441  
 transacted editing, 417–420

`CancelEventHandler`, validating event, 460–461

`CancelNew` method, `ICancelAddNew`, 326, 425–426

Caption property, 47–48

Cell events, `DataGridView`, 277–281

Cell templates, 222–223

Cell types, built-in, 240

Cell types, custom, 270–273

`CellFormatting` event  
 controlling displayed cell content with, 232

creating grouped column `DataGridView`, 347–348

`DataGridView` validation with, 462  
 overview of, 227–231

`CellPainting` event, `DataGridView`  
 customizing cell painting process, 269  
 for header cells, 255–256  
 modifying cell painting behavior, 348–349

`CellParsing` event  
`DataGridView` validation, 462–463  
`DataGridViewTextBoxColumn`, 249–250  
 grid data edits, 256

Cells, errors in `DataGridView`, 467

`CellTemplate` property  
 customizing columns, 274–275  
`DataGridComboBoxColumn`, 252  
`DataGridViewColumn`, 246

`CellType` property, `DataGridViewColumn`, 246

`CellValidated` event, 462–463

`CellValidating` event, 462–463

`CellValueNeeded` event  
 overview of, 239  
 setting up virtual mode, 234, 236–237  
 virtual mode using, 240

`CellValuePushed` event, 240, 250

Chaining binding sources, 116–121

Change Data Source dialog, 44–45

Chart controls, data-bound. *See* `DataBoundBarChartControl`

CheckBox control  
 as default control type, 190  
 icon, 191  
 overview of, 580–581  
 using `DataGridViewCheckBoxColumn`, 251

`CheckChanged` event, CheckBox control, 580–581

`CheckChanged` event, `RadioButton` control, 581

`CheckState` property, CheckBox control, 580–581

`CheckState` property, `DataGridViewCheckBoxColumn`, 251

Child controls, 552, 567

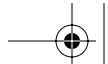
Child relations. *See* Parent-child data binding

`ChildObject`, 121

Choose Toolbox Items dialog, 199, 351

Classes  
 layered application architecture, 21–22  
 partial, 51, 545–565





- typed data sets as, 37
- Windows Forms architecture, 567–570
- Clear method, IList, 300–301
- ClickOnce, 27
- Client-side transactions, ADO.NET, 650–651
- Clone method, custom column types, 275
- Code
  - BarChartControl, 384–391
  - designer-generated, 202–205
  - generating data-binding code for existing controls, 199–201
  - setting control data binding in Properties window, 205–210
  - table adapter designer-generated, 55–56
  - typed data set-generated, 49–52
  - using typed data sets in your, 78–79
- Code-behind
  - hooking up dynamic behavior in, 492–495
  - XAML, 538, 540
- Code view, 526
- Collections
  - access control with ICollection, 295–298
  - adding controls to Windows Forms, 552
  - adding search functionality to, 430–431
  - adding sort functionality to, 432–439
  - binding to grid with templates, 541–543
  - business object, 405–408
  - customizing control mappings, 197–198
  - data binding with interfaces. *See* Interfaces
  - data sets and, 19
  - defined, 16
  - generating bound controls from data sources, 189–196
  - object, 237
  - using BindingList<T>. *See* BindingList<T> class
- ColumnChanged event, DataTable, 652
- ColumnChanging event, DataTable, 652
- ColumnCount property, TextBox, 224
- ColumnHeadersDefaultStyle property, DataGridView, 281–282
- ColumnIndex property, DataGridViewCell, 242
- Columns. *See also* Built-in columns, DataGridView
  - adding to data sets, 48–49
  - automatically sizing, 259–261, 375–376
  - built-in. *See* DataGridViewColumn
  - capturing data errors on, 477–479
  - cell-oriented grid features for, 277–281
  - customizing content with unbound columns, 226–232
  - DataGridViewButtonColumn, 250
  - DataGridViewCheckBoxColumn, 251
  - DataGridViewImageColumn, 251
  - DataGridViewLinkColumn, 250
  - DataGridViewTextBoxColumn, 249–250
  - defining grids in designer, 263–266
  - freezing, 262–263
  - grouped in DataGridView, 345–350
  - header cells for, 255–256
  - implementing custom type of, 269–270, 273–277
  - programmatically adding to grid, 222–224
  - reordering, 266–269
  - setting name of database, 39
  - updating database with data sets, 624–628
- ComboBox control
  - AutoComplete feature, 160–162, 371–375
  - binding data collections to, 88–90
  - dynamically populating, 368–369
  - handling grid data edits with, 257
  - icon, 192
  - .NET Framework 2.0 enhancements, 85–86
  - overview of, 584
  - Smart Tag, 213
  - using DataGridViewComboBoxColumn, 252–255
- CommandField, DataGridView control, 511
- CommandType property, stored procedures, 624
- CompareTo method, SortComparer, 435
- Complex data binding
  - customizing control mappings with, 198–199
  - to individual properties on controls, 134–141
  - .NET Framework 2.0 enhancements, 84–85
  - using IList, 302
- ComplexBindingProperties attribute, 361
- Component base class, Windows Forms, 567–568
- ComponentType property, PropertyDescriptor, 306
- Concurrency
  - ensuring data consistency with, 457
  - handling in ADO.NET, 628–631
  - resolving errors, 483–484
- .config extension, 60
- Configure Data Source wizard, 505





## 676 INDEX

- Connections
  - adding new, 6–8
  - choosing database, 181–184
  - data reader, 620–621, 654–655
  - using table adapters for, 58–62
- connectionStrings configuration, 58–62
- Constraints, typed datasets, 33
- Consumers
  - defined, 288
  - implementing IEnumerable for, 289–290
  - notifying of changes to collection, 313–315
- ContainerControl class, 569
- Contains method, BindingSource, 121
- ContainsListCollection property, GetList method, 304
- ContentBounds property, DataGridViewCell, 242
- ContextMenuStrip property, DataGridViewCell, 242
- ContextMenuStrip property, DataGridViewColumn, 246
- Control base class
  - common controls. *See* Controls, display
  - designing data-bound controls from, 20
  - handling control events, 574–576
  - Windows Forms architecture, 567–570
- Control composition model, WinFx, 523
- Control layout, 589–596
  - anchoring, 590–592
  - creating Windows Forms application, 551–552, 554
  - docking, 592–594
  - positioning and sizing, 589–590
  - setting tab order, 596–598
  - snap-lines simplifying, 557–558
  - using layout container controls, 595–598
  - WinFx applications, 526
- Controls. *See also* Data-bound controls
  - binding data to individual, 90–92
  - binding to data sources. *See* Data sources, binding controls to
  - creating custom user, 586–588
  - developing Windows Forms and, 548
  - handling events, 574–576
  - working with Toolbox, 555–557
- Controls, display, 578–586
  - Button, 580
  - CheckBox, 580–581
  - ComboBox, 584
  - DataGrid, 585–586
  - DataGridView, 586
  - DateTimePicker, 582
  - Label, 579–580
  - ListBox, 582–584
  - ListView, 584
  - overview of, 578–579
  - PictureBox, 585
  - RadioButton, 581
  - RichTextBox, 582
  - TextBox, 581
  - TreeView, 585
- ControlUpdateMode property, BindingSource, 162–163
- Converter property, PropertyDescriptor, 306
- ConvertEventArgs, 149, 154–155
- ConvertEventHandler, 149
- CopyTo method, ICollection, 296–297
- Count property, ICollection, 296
- CreateBindings method, 152–153
- CreateCells method, 225–226
- CreateNavigator, 666
- CTP (Community Technology Preview), 522
- CurrencyManager
  - changing current record in set of data with, 99
  - defining, 92
  - ICurrencyManagerProvider interface, 341
  - synchronizing data between controls, 96–97
- Current property
  - IEnumerator interface, 291
  - manipulating data through binding source, 122
- CurrentChanged event, 131–133, 169–172
- CurrentItemChanged event, 131
- Cursor, data reader as, 656–657
- Custom collections, 237–238
- CustomBusinessObjects example, 408–416
  - binding through Data Sources window, 453–455
  - customer and order classes, 408–410
  - forming controls, 410–413
  - generating test data, 413–414
- Customer objects
  - binding to form controls, 410–413
  - implementing data-bound business objects, 408–410
  - PropertyChanged events, 420–423
- CustomerIDChanged event, PropertyChanged events, 421–423





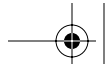
CustomerNameChanged event, Property-Changed events, 421–423  
 CustomersRow class, 38–40, 52  
 CustomersTableAdapter class, 47, 63–64

**D**

Data access layer  
 adding helper methods in, 66–67  
 layered application architecture, 21–27  
 table adapters and, 67  
 Data adapters  
 events, ADO.NET, 651–654  
 loading data sets from database and, 616–618  
 updating database with, 626–628  
 using Update method on, 33  
 Data-bind method, ASP.NET, 496  
 Data binding  
 adding to custom controls, 357–359  
 building applications with Windows Forms.  
*See* Windows Forms applications,  
 building data-bound  
 business objects. *See* Business objects  
 concepts, 82–84  
 data paths within data sources, 92–96  
 defining, 2–3, 81–82  
 designer initialization of, 359–360  
 expressions, 508–509  
 to grid, 86–87  
 IList enabling, 298–303  
 to individual controls, 90–92  
 key interfaces involved in. *See* Interfaces  
 master-details, 104–106  
 to multi-valued controls, 88–90  
 .NET Framework 2.0 and, 84–86  
 paging through data, 99–104  
 smarter data containment and, 97–99  
 synchronizing data between controls, 96–97  
 updating data sources through, 106–108  
 Data-binding expressions, ASP.NET,  
 496–498, 508–509  
 Data-bound columns, 264–265  
 Data-bound controls  
 ASP.NET 1.X, 490–491  
 binding customers and orders to form,  
 410–413  
 data paths within data sources and, 92–96  
 data source mappings and, 10–11  
 generating, 11–12

.NET Framework 2.0 and, 84–86  
 overview of, 20–21  
 simple data binding and, 83–84  
 synchronizing data between, 96–97  
 validation, 459, 463, 473  
 with Visual Studio Designer. *See* Visual  
 Studio Designer, bound controls  
 WinFx, 524, 530–535, 543–545  
 XAML, 539–540  
 Data-bound controls, customizing, 343–398  
 adding data-binding capability, 357–359  
 autocomplete, 371–375  
 autosizing columns, 375–376  
 building from scratch, 379  
 chart controls for decision support, 379–384  
 coding, 384–391  
 delayed initialization, 362–366  
 designer initialization, 359–360  
 developing, 353–354  
 editing, 391–397  
 filtered grid control, 354–356, 377–378  
 grouped columns, 345–350  
 overview of, 343–345  
 specifying binding properties, 360–362  
 specifying data source properties, 367–371  
 test container, 352–353  
 using, 350–352  
 Data concurrency, 457, 483–484  
 Data connections. *See* Connections  
 Data contexts, 536, 543  
 Data display controls  
 Button, 580  
 CheckBox, 580–581  
 ComboBox, 584  
 DataGrid, 585–586  
 DataGridView, 586  
 DateTimePicker, 582  
 label, 579–580  
 ListBox, 582–584  
 ListView, 584  
 overview of, 578–579  
 PictureBox, 585  
 RadioButton, 581  
 RichTextBox, 582  
 TextBox, 581  
 TreeView, 585  
 Data items  
 defined, 16  
 defining, 286–287  
 PropertyDescriptors and, 305–307



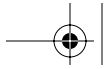


## 678 ■ INDEX

- Data members
  - customizing control mappings, 196, 197–199
  - defined, 7
  - generating data-binding code for existing controls, 201
  - specifying for bound controls, 94–95
- Data objects, 8–10, 16–18
- Data persistence layer (data tier), 15
- Data providers, 6–7, 604–607
- Data readers
  - defined, 607
  - filling binding source with, 126–128
  - loading DataTable with, 619–621
  - reading data into business objects, 654–658
- Data relations, typed datasets, 33
- Data rows
  - capturing errors on, 475–479
  - data sets containing, 31
  - moving between DataRowView and, 123–124
- Data Set Designer
  - adding tables to data sets, 48–49
  - creating typed data sets, 43–45
  - setting up connections, 45–48
- Data sets
  - capturing data errors on, 475–479
  - choosing custom collections vs., 237
  - as collection of collections, 303–304
  - defined, 31
  - as fancy collection class, 405–406
  - overview of, 31–32
  - using in this book, 18–20
- Data sets, ADO.NET
  - creating master-details, 621–623
  - creating programmatically, 611–613
  - data adapter events and, 651–654
  - loading DataTable with data reader, 619–621
  - loading from database, 613–619
  - loading from file, 609–611
  - master-details, 621–623
  - merging data from multiple, 639–641
  - searching, 637–638
  - structure of, 607–609
  - updating with, 624–628, 632–637
- Data Source Configuration wizard
  - adding data sources to project, 179–180
  - adding database data source, 181–184
  - adding new data source and data connection, 6–8
  - adding object data source, 186–189
  - adding Web service data source, 185–186
  - choosing type of data source, 180
  - creating new data sources in property editor, 207
  - designer-generated code files and, 205
  - selecting data objects, 8–10
- Data source control, ASP.NET 2.0
  - DataGrid control working with, 510
  - defined, 498–499
  - using master-details data binding, 515–518
- Data sources
  - adding new, 6–8
  - binding attributes, 360–362
  - binding hierarchical, 518
  - choosing typed data set, 44–45
  - creating bound controls. *See* Data Source Configuration wizard
  - customizing control mappings, 10–11
  - data binding through BindingSource, 97–99
  - data contexts and, 536
  - data paths within, 92–96
  - data providers, 604–605
  - defined, 31
  - designer-generated code files, 202–205
  - dynamically determining properties of, 367–371
  - ErrorProvider extracting information from, 466–467
  - overview of, 15–16
  - paging through data, 99–104
  - selecting in Property window, 207
  - in simple data binding, 91
  - updating through data binding, 106–108
  - updating typed data sets with table adapters, 57–58
  - using designer to define grids, 263–266
- Data sources, ASP.NET, 499–508
  - AccessDataSource control, 507
  - ObjectDataSource control, 505–507
  - overview of, 499–500
  - SiteMapDataSource control, 508
  - SqlDataSource control, 500–505
  - XmlDataSource control, 507–508
- Data sources, binding controls to, 111–176
  - AutoComplete feature, 160–162
  - automatic formatting and parsing, 147–148
  - binding events, 148–154







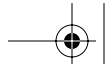
- BindingSource, as data storage container, 124–126
- BindingSource, chaining for master-details, 116–121
- BindingSource, filling with data reader, 126–128
- BindingSource, manipulating data through, 122–124
- BindingSource, navigating data through, 121–122
- BindingSource, overview of, 111–112
- BindingSource, sorting, searching and filtering with, 128–130
- child-parent data binding, 163–165
- complex types, 134–141
- data binding lifecycle, 162–163
- Date/Time column to Date/Time Picker, 142–144
- Date/Time column to Text Box, 144–145
- editing completion, 157–160
- Format event, 154–155
- image column to Picture Box, 141–142
- monitoring with events, 131–133
- multiple copies of data, 165–168
- nullable types, 140–141
- numeric column to Text Box, 145–147
- Parse event, 156–157
- restricting changes to data, 133
- simple data binding, 112–115
- synchronizing many-to-many related collections, 172–176
- type conversion process and format providers, 138–139
- updating to parent data-bound controls, 168–172
- Data Sources window
  - adding data sources, 179–180
  - adding object data sources, 188–189
  - adding outside data sources, 202
  - binding to business objects in, 453–455
  - creating typed data sets in, 41–42
  - customizing control mappings in, 196–199
  - generating bound controls from data sources in, 189–196
  - generating data-binding code for existing controls on, 199–201
  - working with, 177–179
- Data storage container, BindingSource as, 124–126
- Data structures, 405
- Data tables
  - binding to Data View when binding to, 113
  - capturing data errors on, 475–479
  - defined, 609
  - form controls and components from, 192–193
  - loading with Data Reader, 619–621
- Data tier, 15
- Data type drop-down list, 197–199
- Data UI Customization settings, 199
- DataBind, ASP.NET, 494
- DataBinder.Eval method, ASP.NET, 509
- DataBindingComplete event, 221, 376
- DataBindings property, 90–92, 208–209
- DataBoundBarChartControl
  - adding editing support to, 391–397
  - class definition for, 385–386
  - implementing BarChartControl, 381–383
  - overview of, 380–381
- DataBoundBarChartControl, coding, 384–391
  - defining custom control skeleton, 385–386
  - initializing and updating data bindings, 387–389
  - overview of, 384–385
  - using AddBarChartControl method, 389–391
- DataColumn instances, 31
- DataError event, 132
- DataErrorEventArgs, DataGrid View, 468–471
- DataFile property, AccessDataSource control, 507
- DataGrid control
  - ASP.NET, 490–492, 494, 509–510
  - DataGrid View vs., 220
  - limitations, 217
  - overview of, 585–586
- DataGrid View control, 217–284
  - automatic column sizing, 259–262
  - basic data binding with, 219–221
  - binding data collections to grid, 86–87
  - built-in header cells, 255–256
  - calling EndEdit in, 159
  - column and row freezing, 262–263
  - column reordering, 266–269
  - controlling modifications, 221–222
  - custom cell types, 270–273
  - custom column content with unbound columns, 226–232





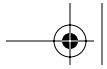
- DataGridView control, *continued*
  - customizing control mappings, 198
  - DataError event, 468–471
    - as default control type, 190
  - defined, 586
  - defining custom column type, 273–277
  - deriving controls from, 21
  - formatting with styles, 281–283
  - grid data edits, 256–259
  - grouped columns, 345–350
  - master-details data binding, 104–106
  - overview of, 218–219
  - programmatically adding columns, 222–224
  - programmatically adding rows, 224–226
  - row and cell error displays, 467–468
  - transacted object editing, 417–420
  - using Visual Studio Designer, 263–266
  - utilizing cell-oriented grid features, 277–281
  - validation events, 462–463
- DataGridView control, built-in column types, 241–255
  - DataGridViewButtonColumn, 250
  - DataGridViewCell properties, 242–246
  - DataGridViewCheckBoxColumn, 251
  - DataGridViewColumn properties, 246–249
  - DataGridViewComboBoxColumn, 252–255
  - DataGridViewImageColumn, 251–252
  - DataGridViewLinkColumn, 250
  - DataGridViewTextBoxColumn, 249–250
- overview of, 241–242
- DataGridView control, virtual mode, 233–241
  - choosing data sets or custom collections, 237–238
  - initializing grid, 238–239
  - overview of, 233–234
  - setting up, 234–237
  - summary, 240–241
  - understanding behavior of, 239–240
- DataGridView icon, 191
- DataGridViewAutoSizeMode, 259–260
- DataGridViewButtonColumn, 250
- DataGridViewCell class
  - built-in cell types derived from, 240
  - cell templates as instances of, 222
  - properties, 242–246
  - working with built-in header cells, 256
- DataGridViewCellStyle, 282
- DataGridViewCell.CellStyle property, 282
- DataGridViewCellStyle object, 226
- DataGridViewCheckBoxColumn, 251
- DataGridViewColumn, 240, 246–249
  - DataGridViewColumn.DefaultCellStyle property, 281–282
- DataGridViewColumnHeaderCell, 255
- DataGridViewComboBoxColumn, 252–255
- DataGridViewImageColumn, 251–252
- DataGridViewLinkColumn, 250
- DataGridViewRow.DefaultCellStyle property, 282
- DataGridViewRowHeaderCell, 255
- DataGridViewSelectionMode, 277–280
- DataGridViewTextBoxCell class, 223–224
- DataGridViewTextBoxColumn, 223, 249–250
- DataGridViewTextBoxEditingControl, 249–250
  - dataMember parameter, Binding class, 136
- DataMember property
  - DataBoundChartControl, 386
  - DataGridView, 219–221
  - master-details data binding, 105–106
- DataMemberChanged event, 132, 221
- DataPropertyName property, 227, 246
- DataRowState, 625
- DataRowView class
  - implementing IEditableObject, 157
  - moving between DataRow and, 123–124
  - supporting transacted object editing, 417–420
- DataSet
  - ADO.NET, 607–609
  - icon, 191
- DataSet property, XmlDocument, 659
- .datasource files, 202–203
- dataSource parameter, Binding class, 136
- DataSource property
  - adding columns to grid, 264–265
  - adding rows to grid, 227
  - binding data collections to grid, 86–87
  - complex data binding using, 84
  - creating Windows Forms application, 553–554
  - data binding with DataGridView, 220–221
  - DataGridComboBoxColumn, 253
  - defining custom control skeleton, 386
  - master-details data binding, 105–106
  - .NET Framework 2.0 and, 85
  - simple data binding, 113–114
- DataSourceChanged event, 132, 221





- DataSourceID property
    - for ASP.NET data-bound controls, 500
    - data-binding expressions, 509
    - DataGrid control, 510–511
  - dataSourceUpdateMode parameter, Binding class, 136
  - DataSourceUpdateMode property, 163, 209–210
  - DataTable
    - events, 651–652
    - overview of, 31–33
    - using DataView and, 123
  - DataTextField property, ASP.NET Drop-DownList control, 497
  - DataValueChanged event, 397
  - DataValueField property, ASP.NET Drop-DownList control, 497
  - DataView class
    - binding to data tables really binding to, 113
    - defining, 123
    - implementing ITypedList, 307–309
    - as key class for data binding, 287
    - working with in ADO.NET, 641–642
  - DateTime column, DateTimePicker, 142–144
  - DateTime column, TextBox, 144–145
  - DateTimeFormatInfo provider, 145
  - DateTimePicker control
    - binding DateTime column to, 142–144
    - as default control type, 190
    - icon, 192
    - limitations of, 144
    - overview of, 582
  - DBConcurrency exception, 483
  - DBNull, 140–141, 155
  - Debugging
    - running FirstDataApp, 13–14
    - in Visual Studio, 563
  - Declarative programming, design-time, 378
  - DefaultBindingProperties attribute, 361
  - DefaultCellStyle property, 246, 281–282
  - DefaultNewRowValue property, DataGridViewCell, 242
  - DefaultStatus property, custom columns, 274–276
  - Definitions, data set tables, 44
  - Delegate inference, 575
  - Delete method, table adapters, 54
  - DeleteCommand property, database updates, 626
  - Design tools, 36
  - Designer. *See* Visual Studio Designer, bound controls
  - Designer view, 526
  - .Designer.cs extensions
    - adding database data sources, 184
    - defined, 49–50
    - example, 50–52
  - .Designer.vb extensions, 49–50
  - DesiredType property, ConvertEventArgs, 149–150
  - Details view, Data Sources window, 191, 195–196
  - DetailsView control, ASP.NET, 512–514, 515
  - Direction, in data binding, 82–83
  - DirectX graphics engine, in WinFx, 523
  - Disconnected data access, ADO.NET, 603
  - Displayed property, DataGridViewCell, 243
  - DisplayIndex property, 247, 267–269
  - DisplayMember property
    - ComboBox control, 85–86, 88–90
    - DataGridComboBoxColumn, 253
    - setting in Properties window, 207–208
  - DisplayStyleForCurrentCellOnly property, DataGridComboBoxColumn, 253
  - Dispose method
    - IDisposable interface, 293
    - loading DataTable with data reader, 620
    - Windows Forms architecture, 570
  - DividerWidth property, DataGridView, 247, 262–263
  - Dock property
    - DataGridView Smart Tag, 265
    - docking controls, 592–594
  - Drag and drop process. *See* Mouse clicks
  - DropDownList control, ASP.NET, 497
  - DropDownStyle, ComboBox, 584
  - DropDownWidth property, DataGridViewComboBoxColumn, 253
  - DumpList method, IBindingList, 320
- ## E
- Edit
    - adding to custom data bound controls, 391–397
    - binding controls to data sources, 157–160
    - DataGridViewTextBoxColumn, 249–250
    - FormView, 515
    - grid data, 256–259
    - INotifyPropertyChanged, 423–424



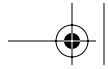


- Edit, *continued*
  - PropertyChanged events, 420–423
    - supporting transacted object, 416–420
    - using Data Sources window toolbar, 179–180
  - Edit Columns dialog, 263–264, 276–277
  - EditedFormattedValue property, DataGridViewCell, 243
  - EditingControlShowing event, 257–258
  - EditType property, DataGridViewCell, 243
  - EnableVisualStyle method, Windows Forms, 572
  - EndEdit method
    - ISupportInitialize, 335–337
    - ISupportInitializeNotification, 338–341
    - updating bound controls, 170
    - using BindingSource as data storage container, 125–126
  - EndEdit method, IEditableObject
    - overview of, 157–159
    - transacted additions to collections, 439–441
    - transactional editing of objects, 328–329, 417–420
  - EndNew method, ICancelAddNew, 326, 425–426
  - Error handling, 457–485
    - capturing errors on data sets, 475–479
    - data concurrency resolution, 483–484
    - DataGridView, 467–468
    - DataGridView DataError event, 468–471
    - IDataErrorInfo interface for, 330–331, 479–482
    - overview of, 457–458
    - validation controls, 474–475
    - validation down control hierarchy, 472–473
    - validation events, handling, 459–462
    - validation events, with DataGridView, 462–463
    - validation up control hierarchy, 463
    - validation, with AutoValidate property, 471–472
    - validation, with ErrorProvider, 464–467
    - Windows Forms validation, 458–459
  - Error property, IDataErrorInfo, 331
  - ErrorIconBounds property, DataGridViewCell, 243
  - ErrorProvider control, 459, 464–467
  - ErrorText property, DataGridViewCell, 243
  - Esc key, transacted object editing, 417, 420
  - Eval method, data-binding expressions, 509
  - EventHandler
    - control events, 574–576
    - creating Windows Forms with Visual Studio, 561–563
    - PropertyChanged events, 421–423
  - Events
    - binding, 148–154
    - BindingSource, 131–133
    - control, 574–576
    - controlling modifications to data in grid, 221–222
    - data sets and data adapter, 651–654
    - DataGridView cell, 277–281
    - DataGridView DataError, 468–471
    - DataGridView validation, 462–463
    - DataTable, 651–652
    - IBindingList, 312–313
    - paging through data with, 101–103
    - populating column cells through, 227
    - typed data table, 38–39
    - validation, 459–462
    - validation based on, 459
    - virtual mode, 234
    - Windows Forms architecture, 570
  - ExecuteNonQuery method, SqlCommand, 646–647
  - ExecuteReader method, 621
  - ExecuteScalar method, SqlCommand, 646
  - Expressions
    - adding tables to data sets, 48–49
    - ASP.NET data-binding, 496–498, 508–509
  - Extended provider control, defined, 464
  - Extensible Application Markup Language. *See* XAML (Extensible Application Markup Language)

**F**

  - Fat client. *See* Smart client applications
  - Fault tolerance, 25
  - Fields, 7
  - Fill, AutoSizeMode, 260–261
  - Fill method
    - filling typed data sets with table adapters, 56–57
    - loading data sets from database, 616–618
    - untyped data sets, 32–33
    - Update method vs., 33
  - FillError event, SqlDataAdapter, 654





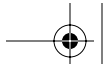
FillWeight property, AutoSizeMode, 261  
 FillWeight property, DataGridViewColumn, 247  
 Filter property  
   IBindingListView, 323–325, 444  
   setting on binding source, 130  
   synchronizing many-to-many relationships, 175–176  
 Filtered grid control  
   adding data-binding capability, 357–359  
   autocompleting input, 371–375  
   autosizing columns in grid, 375–376  
   building, 354–356  
   delayed initialization, 362–366  
   determining properties of data source, 367–371  
   specifying binding properties on controls, 360–362  
   summary, 376–378  
 FilterExpression property, SqlDataSource control, 515–517  
 Filters  
   data view supporting, 642  
   IBindingListView supporting, 323–325, 443–453  
   master-details binding supporting, 516–517  
   removing from collections, 325  
 Find method  
   BindingSource, 121, 129–130  
   creating rich object collections, 426  
   IBindingList, 312  
 FindCore method, 430–431  
 FlatStyle property, DataGridViewComboBoxColumn, 253  
 FlowDirection property, FlowLayoutPanel control, 595  
 FlowLayoutPanel control, 595–597  
 Font, BarChartControl, 383  
 foreach operator, IEnumerable and IEnumerator, 292–293, 294  
 Form class, 15, 569  
 Format event  
   event binding and, 149–150, 153–154  
   handling, 154–155  
 Format providers, 138–139, 147–148  
 FormatException, 148  
 formatInfo parameter, Binding class, 137  
 FormatString parameter, 137, 145–147

FormattedValue property, DataGridViewCell, 243  
 FormattedValueType property, DataGridViewCell, 243  
 Formatting  
   the data, 135  
   data binding lifecycle, 163  
   with styles, 226, 281–283  
 Formatting and Advanced Binding dialog, 209–210  
 FormattingApplied property, 230, 260  
 FormattingEnabled parameter  
   binding DateTime column to TextBox, 145  
   binding numeric column to TextBox, 146  
   overview of, 136–138  
 FormView control, ASP.NET, 514–515  
 Frozen property  
   column and row freezing with, 262–263  
   DataGridViewCell, 243  
   DataGridViewColumn, 247

## G

Generic collection classes, 406–408  
 Generic data classes, ADO.NET, 604  
 Generics, defined, 287  
 Get Visited Count button, virtual mode, 240  
 GetColumnsInError method, 477–479  
 GetCustomers method, objectDataSource control, 506–507  
 GetEnumerator method, IEnumerable, 290, 292  
 GetEnumerator method, IEnumerator<T>, 293  
 GetFormattedValue method, custom cell types, 272–273  
 GetItemProperties method, ITypedList, 308–309  
 GetManifestResourceStream method, custom cells, 273  
 GetOrdinal method, data reader, 657  
 GetProperties method, 367  
 GetRegions method, 633–634  
 GetString method, data reader, 656–657  
 GetTestCustomers method, 413–414  
 GetValue method, PropertyDescriptor, 306  
 GetXML method, loading data sets, 609  
 Global exception handlers, 573–574  
 Graphics, WinFx, 523





**Grids**

- adding to Windows Forms, 550–552
- binding customers and orders objects to form controls, 410–413
- binding data collections to, 86–87
- binding through BindingSource, 112–115
- handling data edits, 256–259
- handling row selecting in, 495–496
- implementing data-bound business objects, 408–410
- master-details data binding in, 104–106
- presenting tabular data. *See* DataGridView control
- templates in XAML binding collections to, 541–543
- WinFx, 526, 530–531
- GridView control, ASP.NET, 509–512, 513–514
- Grouped columns, 345–350

**H**

- HasErrors boolean property, 477
- HasStyle property, DataGridViewCell, 244
- Header cells, built-in, 255–256
- HeaderCell property, DataGridViewColumn, 247
- HeaderText property, DataGridViewColumn, 247
- “Hello World” application, WinFx, 526–528
- Heterogenous columns, DataGridView, 224–225
- Hierarchical binding, ASP.NET, 518
- Hierarchy, Windows Forms, 567–570
- Hollis, Billy, 475
- Hot cells, 281
- HTML, ASP.NET Web forms, 491
- HTTP POST, ASP.NET, 489

**I**

- IBindableComponent interface
  - data-bound controls and, 20–21
  - .NET Framework 2.0 and, 85
  - simple data binding with, 84–85
- IBindingList interface, 310–322
  - binding to data tables, 113
  - BindingList<T> implementation, 425
  - exercising change notifications, 316–317
  - filter function, 128–130
  - notifying consumers of changes, 313–315

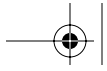
- overview of, 310–313
- restricting changes to data, 133
- search function, 128–130, 322
- sort function, 128–130, 317–321
- using for IList interface limitations, 303
- IBindingListView interface
  - adding functionality of, 443–453
  - multi-property sorting with, 318
  - overview of, 323–325
  - sorting with, 129–130
- ICancelAddNew interface
  - creating custom collection types, 425–426
  - overview of, 325–326
  - using BindingList<T>, 425–426
- ICollection interface, 295–298
- ICollection<T> interface, 298
- IComparer<T> interface, 433–435
- Icons, data sources controls, 189–192
- Icons, ErrorProvider control, 466
- IConvertible interface, 148
- ICurrencyManagerProvider interface, 341, 365
- ICustomTypeDescriptor interface, 332–334
- id attribute, ASP.NET controls, 492
- Id property, virtual mode, 237–238
- IDataErrorInfo, 330–331, 483–484
- Identity values, 633
- IDisposable interface
  - IEnumerator<T> class deriving from, 293
  - implementing BarChartControl, 383
  - Windows Forms architecture, 570
- IEditableObject interface
  - calling EndEdit, 159
  - DataRowView class implementing, 157
- ICancelAddNew interface vs., 325
  - managing transacted additions to collections, 439–441
  - overview of, 328–329
  - transacted editing of objects with, 416–420
- IEnumerable interface, 289–295, 301
- IEnumerator<T> class, 293
- IList interface
  - complex data binding with, 85–86
  - defined, 288–289
  - List<T> implementation, 411
  - overview of, 298–303
- IListSource interface, 303–305
- IList<T> interface, 303
- Image column, PictureBox control, 141–142





- Image property, PictureBox/TreeView control, 585
- Images
  - DataGridViewImageColumn for, 251–252
  - PictureBox control for, 580, 585
- Implementers, 288
- Indexers, data readers, 656–657
- IndexOf method, BindingSource, 122
- IndexOf method, IList, 300
- Indigo (Windows Communication Foundation), 521
- InheritedAutoSizeMode property, DataGridViewColumn, 247
- InheritedStyle property, DataGridViewCell, 244
- InheritedStyle property, DataGridViewColumn, 247
- InitData method, 118, 237–238
- Initialization
  - coding custom-bound controls, 387–389
  - delayed, data-bound controls, 362–366
  - ISupportInitialize, 334–337
  - ISupportInitializeNotification, 337–341
  - Main method and, 573
  - using designer, 359–360
- InitializeComponent method
  - defining grids in designer, 266
  - designer-generated code, 202
  - ISupportInitialize, 335–336
  - overview of, 565–566
  - in XAML, 540
- Initialized event, DataSet, 651
- INotifyPropertyChanged interface
  - business object edit notifications with, 423–424
  - overview of, 329–330
  - raising item changed events, 439
  - using for IList interface limitations, 303
- INotifyPropertyCollectionChanged interface, WinFx, 536
- Input validation, 474–475
- Insert method, IList, 300
- Insert method, table adapters, 54
- INSERT stored procedure, 637
- InsertCommand property, updating with data adapters, 626
- InsertTemplate, FormView, 515
- IntelliSense, 36
- Interfaces, 285–342
  - data binding and, 286–289
  - IBindingListView, 323–325
  - ICancelAddNew, 325–326
  - ICollection, 295–298
  - ICurrencyManagerProvider, 341
  - ICustomTypeDescriptor, 332–334
  - IDataErrorInfo, 330–331
  - IEditableObject, 328–329
  - IEnumerable and IEnumerator, 289–295
  - IList, 298–303
  - IListSource, 303–305
  - implementing with data providers in .NET, 605
  - INotifyPropertyChanged, 329–330
  - IRaiseItemChangedEvents, 327
  - ISupportInitialize, 334–337
  - ISupportInitializeNotification, 337–341
  - ITypedList, 307–309
  - overview of, 285–286
  - property descriptors, 305–307
  - WinFx, 536
- Interfaces, IBindingList, 310–322
  - exercising change notifications, 316–317
  - notifying consumers of changes, 313–315
  - overview of, 310–313
  - search function, 322
  - sorting function, 317–321
- IntiGrid method, 238–239
- IRaisedItemChangedEvents, 441–443
- IRaiseItemChangedEvents interface, 327
- Is<ColumnName>Null method, 140–141
- IsDataBound property, DataGridViewColumn, 248
- IsFixedSize property, IList, 299, 301
- IsInEditMode property, DataGridViewCell, 244
- IsInitialized property, ISupportInitializeNotification, 339–341
- IsMdiContainer property, MDI, 577–578
- Isolation levels, transaction, 645
- IsPostBack property, 495
- IsReadOnly property, IList, 299, 301
- IsSorted property, IBindingList, 311
- IsSynchronized property, ICollection, 297–298
- ISupportInitialize
  - DataBoundBarChartControl implementing, 386
  - delayed initialization with, 362–366
  - overview of, 334–337





ISupportInitializeNotification interface, 337–341  
 Items property, DataGridComboBoxColumn, 254  
 ItemsSource property, WinFx data binding, 535  
 Iteration, IEnumerable and IEnumerator, 290–292  
 IListView interface, 307–309

## L

Label controls  
   Detail view control mapping with, 194  
   generating for data-bound controls, 11–12  
   icon, 191  
   overview of, 579–580  
 Layered application architecture, 15–16, 21–27  
 Layers, 23–26, 29  
 Layout, control. *See* Control, layout  
 Lifecycle, data binding, 162–163  
 Lifecycle, Windows Forms, 569–570  
 LinkLabel icon, 192  
 Links, DataGridViewLinkColumn, 250  
 Links, defining grids in designer, 263–265  
 List-based binding, 84  
 ListBox control  
   ASP.NET, 497  
   binding data collections to, 88–90  
   icon, 192  
   overview of, 582–584  
   Smart Tag, 213–214  
   WinFx, 535  
   XAML, 540  
 ListChanged events  
   BindingList<T> supporting, 425  
   defining, 132  
   FilteredGrid control, 370–371  
   IBindingList, 312  
   INotifyPropertyChanged, 329–330, 424  
   IRaiseItemChangedEvents of, 327  
   ListChangedEventArgs properties of, 314–315  
   ListChangedType enumeration values, 315  
   synchronizing many-to-many related collections, 174–175  
 ListChangedEventArgs properties, 314–315  
 ListChangedType property, ListChangedEventArgs, 314

Lists, defined, 16  
 ListSortDescriptionCollection, IBindingListView, 444  
 List<T> generic type  
   adding searching to collections, 431  
   adding sorting to collections, 432–434  
   IList interface using, 302  
   overview of, 407–408  
 ListView control, 584  
 ListViewItem class, 584  
 Load event, 570  
 LoadDataRow method, 641  
 Loading, ADO.NET  
   data into XPathDocument, 664–665  
   data sets from database, 613–619  
   data sets from files, 609–611  
   DataTable with DataReader, 619–621  
 localhost, 8  
 Location property, 589–590  
 Logical units, WinFx, 523  
 Lookup complex data binding, 199  
 LookupBindingProperties attribute, 361

## M

Main method, 549, 570–574  
 many-to-many relationships, synchronizing, 172–174  
 MarshalByRefObject, 567–568  
 MaskedTextBox control, 144  
 Master-details data binding  
   ASP.NET, 515–518  
   chaining BindingSources for, 116–121  
   generating controls with designer, 214–216  
   implementing data-bound business objects, 408–410  
   overview of, 104–106  
   reversing, 168–172  
 Master-details data sets, ADO.NET, 621–623  
 MaxDropDownItems property, DataGridComboBoxColumn, 254  
 MDI (multiple-document interface) applications, 577–578  
 MdiParent property, MDI applications, 578  
 Member controls, Windows Forms, 567  
 MemberDescriptors, 305–306  
 Members, adding, 557–561  
 Memory. *See* DataGridView control, virtual mode  
 MenuStrip control, 598–600





Merge method, ADO.NET, 639–641  
 MergeFailed event, DataSet, 651  
 MFC (Microsoft Foundation Class), 49–50  
 Microsoft SQL Server  
   as data provider in .NET, 604  
   loading data sets from database, 613–619  
   working with, 606–607  
 MinimumWidth property, DataGridView-Column, 248  
 MissingSchemaAction property, 622, 634  
 Mouse clicks  
   configuring table adapters to use stored procedures, 76  
   generating data-bound controls, 11, 200–201  
   generating form controls and components, 193–194  
   working with Data Sources window, 177  
 MoveFirst method, BindingSource, 122  
 MoveLast method, BindingSource, 122  
 MoveNext method  
   IEnumerator, 290, 292, 294  
   navigating data through BindingSource, 122  
   XPathNavigator query methods, 666  
 MovePrecious method, BindingSource, 122  
 MoveToXXX methods, XPathNavigator, 668–669  
 Multiline property, TextBox control, 581  
 multiple-document interface (MDI) applications, 577–578  
 Multithreaded programming, ICollection, 297  
 m\_Visited collection, 239

## N

Name property, DataGridViewColumn, 248  
 Name property, PropertyDescriptor, 306  
 Namespace, icon, 191  
 Namespace, XPathNavigator, 669  
 Naming conventions  
   adding tables to data sets, 48–49  
   configuration files at runtime, 60  
   data-bound controls, 11  
   generating bound controls from data sources, 194  
   hooking up event handler to get different, 561–563  
   loading data sets from database, 616–619

  setting control data binding in Properties window, 206  
 typed data sets, 39, 47–48  
 Navigation  
   with BindingNavigator. *See* BindingNavigator control  
   BindingSource properties and methods, 121–122  
   of XML documents. *See* XPathNavigator  
 XML documents, 667–669  
 .NET Framework 2.0  
   data binding enhancements, 84–86  
   data sets, 19–20  
   generic collection classes, 406–408  
 .NET Framework Data Provider, 45  
 Never value, DataSourceUpdateMode property, 163  
 New Connection button, 6–7  
 New Project dialog, 4–5, 554–555  
 NewIndex property, ListChangedEventArgs-Args, 314  
 No mapping icon, 191  
 NodeType property, XPathNavigator, 667  
 NotSortable property, DataGridView, 280  
 Null value  
   binding numeric column to TextBox, 146  
   customizing bindings in Properties window, 209  
   handling Format event, 155  
   overview of, 140–141  
 Nullable types, 140  
 nullValue parameter, Binding class, 137  
 Numeric columns, binding to TextBox, 145–147  
 NumericUpDown control icon, 192

## O

Object collections, 237  
 Object reference, 129–130  
 ObjectDataSource control, 505–507, 518  
 Objects  
   data-bound business. *See* Business objects  
   overview of, 16–18  
   performing data binding against, 186–189  
 ObservableCollection<T> class, 536  
 ODBC, 604  
 Offline applications, 27, 603  
 OldIndex property, ListChangedEventArgs, 314



OLE DB, 604  
 OLE images, conversion handling, 142  
 OnAddingNew event handler, 125–126  
 OnCellFormatting, DataGridView override, 347–348  
 OnCellPainting method, DataGridView override, 348–349  
 OnCellValueNeeded handler, virtual mode, 239  
 OnCountryComboChanged event, 258–259  
 One-way data binding, 82  
 OnFilterClicked event handler, 374  
 OnFormLoad method, paging through data, 101–102  
 OnGetList method, bound controls, 412  
 OnGetVisitedCount button Click handler, virtual mode, 239  
 OnLoadData handler, 552–553  
 OnPositionChanged event handler, 102–103  
 OnPropertyChanged value, DataSourceUpdateMode property, 163  
 OnSelectedIndexChanged event, 495  
 OnStartingUp method, WinFx, 529–530  
 OnValidating value, DataSourceUpdateMode property, 163  
 Optimistic currency, 33  
 Oracle, 605  
 Order objects, 410–413  
 OwningColumn property, DataGridViewCell, 244  
 OwningRow property, DataGridViewCell, 244

**P**

Page processing, ASP.NET  
 handling row selecting in grid, 495–496  
 hooking up dynamic behavior in code-behind files, 492–495  
 overview of, 489  
 Page\_Load event handler, 494  
 Paging, DataGridView control, 511  
 Paging, FormView, 515  
 Painting, custom and row headers, 255–256  
 Parent-child data binding  
 master-details data binding using, 117–121  
 overview of, 163–165  
 updating to parent data-bound controls, 168–172

Parsing  
 data binding lifecycle, 163  
 event binding with Parse event, 153–154  
 handling Parse event, 156–157  
 overview of, 147–148  
 Parse event, 149  
 Partial classes, 51, 545–565  
 Passing the data, 135  
 PasswordBox control, 530–531  
 Performance, 25, 68  
 PictureBox control  
 binding image column to, 141–142  
 customizing data source control mappings, 10  
 generating data-bound controls, 12  
 icon, 191  
 label control vs., 579  
 overview of, 585  
 Pixel-based programming, WinFx vs., 523  
 plus sign (+), 202  
 Position property, 99–100, 121  
 PositionChanged event, 101–103, 132  
 Postback handling, DataGridView control, 510  
*Pragmatic ADO.NET* (Wildermuth), 602  
 PreferredSize property, DataGridViewCell, 244  
 Presentation layer  
 of layered application architecture, 21–27  
 using data readers in, 607  
 using data sets in, 18–19  
 Preview Data option, Smart Tag, 212–213  
 Programmatically constructing data sets, 611–613  
*Programming .NET Components* (Löwy), 407, 574  
*Programming Windows Presentation Foundation* (Sells and Griffiths), 522  
 Programming, Winfx UI, 522–524  
 Properties  
 data-bound business object, 404  
 DataGridViewComboBoxColumn, 252–254  
 DataGridView, 218  
 DataGridView cell styles, 281–283  
 DataGridViewCell, 242–246  
 DataGridViewColumn, 246–249  
 defined, 7  
 designing own data-bound controls from, 21  
 dynamically determining data source, 367–371





- exposing data-binding, 307–309
- setting control data binding, 206–210
- setting values with designer-generated code, 565
- sorting on multiple. *See* `IBindingListView` interface
- transacted object editing, 416–420
- typed data set, 37–38, 47–48
- Properties window
  - adding member to Windows Forms, 557–560
  - DataGridView settings in, 266
  - hooking up event handler in, 561–563
  - setting control data binding in, 205–210
- Property change events, 420–424
- `PropertyChangedEventHandler`, 330
- `PropertyDescriptor`
  - determining properties of data sources, 367–371
  - `ListChangedEventArgs`, 315
  - manipulating data through binding source, 124
  - overview of, 305–307
- `propertyName` parameter, Binding class, 136
- `PropertyType` property, `PropertyDescriptor`, 306
- `PropertyUISettings`, data source files, 204
- `ProposedValue` property, `DataTable`, 652–653
- Provider model, ASP.NET, 499–500
- Provider-specific classes, ADO.NET, 604
- Pseudo-properties, Properties window, 206–210

## Q

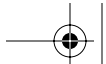
- Queries
  - loading data sets from database, 616–617
  - table adapters configured for custom, 69–72
  - table adapters configured for stored procedures, 74–77
  - using Smart Tags, 210–214
  - writing SQL statements with Query Builder, 72–73
  - against XML data, 665–667
- Query Builder, 72–73, 212
- question mark (?), 140

## R

- RadioButton control, 581
- `RaiseItemChangedEvents` property, 439–441

- Raster graphics, 523
- Read method, data reader, 656
- Readers. *See* Data readers
- `ReadOnly` property
  - controlling modifications to grid data, 221
  - `DataGridView Smart Tag`, 265
  - `DataGridViewButtonColumn`, 250
  - `DataGridViewCell`, 244
  - `DataGridViewColumn`, 248
  - `DataGridViewTextBoxColumn`, 249–250
  - in virtual mode, 240
- `ReadXml` method
  - creating Windows Forms application, 552
  - loading data sets from file, 609–611
  - working with `XmlDataDocument`, 661–662
- `Reference.map` item, 203
- `RejectChanges` method, client-side transactions, 650–651
- Relational data access, ADO.NET, 603–607
- Relational data, defined, 286
- Relations, typed data sets, 51
- Remove method, `IList`, 300–301
- `RemoveAt` method, `IList`, 300–301
- `RemoveFilter` method, `IBindingListView`, 324, 325, 444
- `RemoveIndex` method, `IBindingList`, 312
- `RemoveSort` method, `IBindingList`, 312, 317
- `RemoveSortCore` method, `IBindingList`, 438–439
- Renderers, .NET 2.0, 600
- Rendering, `BarChartControl`, 381–383
- Rendering, custom data structure for, 383–384
- Reset method, `IEnumerator<T>` class, 293
- Resizable property, `DataGridViewCell`, 244
- Resizable property, `DataGridViewColumn`, 248
- Rich Text Format (RTF), 582
- `RichTextBox` control, 582
- Rollback transactions, 645–646
- `RowAdded` event, `DataGridView`, 281
- `RowChanged` event, `DataTable`, 652
- `RowChanging` event, `DataTable`, 652
- `RowDeleted` event, `DataTable`, 652
- `RowDeleting` event, `DataTable`, 652
- `RowFilter` property, data views, 642
- `RowIndex` property, `CellFormatting` event, 230
- `RowIndex` property, `DataGridViewCell`, 244





**Rows**

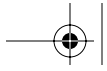
- adding in virtual mode, 234
- errors in DataGridView, 467
- freezing, 262–263
- header cells for, 255–256
- programmatically adding to grid, 224–226
- selecting in grid, 495–496
- updating database with data sets, 624–628
- RowsAdded event
  - controlling stored cell content with, 232
  - defined, 227
  - using, 231–232
- RowsHeadersDefaultCellStyle property, 281–282
- RowState flag, 33
- RowState property, DataSet class, 625
- RowStateFilter property, DataView, 642
- RowUpdated event, SqlDataAdapter, 654
- RowUpdating event, SqlDataAdapter, 654
- RowValidated event, DataGridView, 462–463
- RowValidating event, DataGridView, 462–463
- RTF (Rich Text Format), 582
- Run method, 527, 572
- runat = "server" attribute, ASP.NET, 492
- Running, data-bound applications, 13–14
- Runtime, data binding, 37

**S**

- Saving, connection strings, 8–9
- Scalability, 25
- Scoping transactions, ADO.NET, 647–649
- ScrollableControl class, 569
- Scrollbars property, TextBox control, 581
- Scrolling, while freezing rows/columns, 262–263
- Search Criteria Builder dialog, 211
- Search functionality
  - adding to custom collections, 430–431
  - adding to rich object collections, 426
  - with BindingSource, 128–130
  - data sets, in ADO.NET, 637–638
  - with IBindingList, 322
- SearchControl user control, 588–589
- Security
  - basing table adapters on stored procedures and, 68
  - dividing applications into tiers for, 25
  - IEnumerable and IEnumerator interfaces and, 294

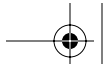
- smart clients and, 28
- Select method, overloads, 638
- SELECT query, in database updates, 627–628
- SelectCommand property
  - calling update stored procedures, 634
  - SqlDataSource control, 500, 515
  - updating database with data sets, 626, 627
- Selected property, DataGridViewCell, 244
- SelectedIndexChanged event
  - ComboBox control, 584
  - grid data edits, 258
  - ListBox control, 583
  - row selection in grid, 495
- SelectedItem(s) property, ListBox control, 583
- SelectedValue property, ComboBox, 88–90
- SelectedValue property, updating
  - data-bound controls, 170
- SelectIndex property, ListBox control, 583
- SelectionMode property, DataGridView, 277–280
- SelectionMode property, ListBox control, 583–584
- SELECT...WHERE clause, DetailsView control, 514
- Server Explorer, 44–45
- SetBinding method, WinFx, 535
- SetColumnError method, DataRow class, 476
- SetConnection method, 62
- SetDisplayOrder method, columns, 269
- SetError method, ErrorProvider, 465–467
- Settings class, connection management, 58–60
- SetValue method, IRaiseItemChanged-Events, 327
- SetValue method, PropertyDescriptor, 306–307
- Show method, displaying other forms, 576–577
- Show method, WinFx, 527
- ShowDialog method, displaying other forms, 577
- ShowXXXButton properties, DataGridView control, 511
- Simple data binding
  - with binding sources, 112–115
  - customizing control mappings using, 198–199
  - overview of, 83–84
  - performing, 90–92
  - using IList interface, 302





- Site property, DataGridViewColumn, 248
- SiteMapDataSource control, 508
- Size
  - controls on forms, 589–590
  - DataGridViewCell, 245
  - PictureBox, 585
  - TableLayoutPanel control, 595–596
  - Windows Forms, 551, 554
- Smart client applications
  - configuring, 24–25
  - defining, 27–28
  - separate layers in, 29
- Smart Tags
  - adding columns to grids, 264–266
  - generating data bindings, 210–214
  - working with, 556
- SmlDataDocument class, ADO.NET, 659–663
- Snap-lines, 557–558
- SnycRoot property, ICollection, 297–298
- Solution Explorer
  - creating typed data sets, 43, 49–50
  - XSD files hidden in, 42
- Sort functionality
  - adding to collections, 432–439
  - with BindingSource, 128–130
  - data views, 642
  - with DataGridView control, 511
  - with IBindingList, 317–321
  - with IBindingListView, 323–325, 443–453
  - with IEnumerable, 291–292
  - rich object collections, 426
- Sort property, data views, 642
- SortComparer, 433–434
- SortComparer, adding sorting to collection, 437–438
- SortDescriptions property, IBindingListView, 323–324, 444
- SortDirection property, IBindingList, 311
- SortDirection property, IBindingListView, 324
- Sorted property, DataGridViewComboBoxColumn, 254
- SortExpression property, DataGridView control, 511
- SortMode property, DataGridView, 280
- SortMode property, DataGridViewColumn, 248
- Source property, database column names, 39
- Source property, typed data sets, 47–48
- SplitContainer control, 595–597
- Splitter control, SplitContainer control, 595
- Spreadsheets, creating, 218
- SQL Express engine, 614–615
- SQL Server, 45–46
- SQL statements
  - adding custom query to table adapters, 69–72
  - configuring table adapter to use stored procedures, 74–77
  - writing with Query Builder, 72–73
- SqlCommand object, 615–616, 646
- SqlCommandBuilder, 627–628
- SqlDataAdapter class
  - events of, 654
  - loading data sets from database, 616–618
  - updating database with data sets, 626
- SqlDataReader, 654–658
- SqlDataSource control
  - DetailsView control and, 514
  - master-details data binding in, 515–518
  - overview of, 500–505
- SqlParameter class, 636–637
- SqlTransaction class, 643–644
- STAThread attribute, Main method
  - overview of, 572–573
  - Windows Forms applications using, 550
  - WinFX using, 528
- StatusCells, 270–273
- StatusColumns, 273–277
- StatusImage, 276
- StatusStrip control, 598–600
- Stored procedures
  - basing table adapters on, 67–69
  - configuring table adapter to use, 74–77
  - loading data sets from database, 616
  - retrieving data with, 623–624
- Strings
  - automatic formatting and parsing, 147–148
  - filter, 175–176
  - setting textual data-binding behavior of custom objects, 413–414
- Styles
  - DataGridView, 281–283
  - DataGridViewCell, 245
  - WinFx layered, 523–524
- SupportsAdvancedSorting property, IBindingListView, 323–324, 444





- SupportsChangeNotification property,
    - IBindingList, 311, 313
  - SupportsFiltering property,
    - IBindingListView, 323, 444
  - SupportsSearching property, IBindingList, 311, 322
  - SupportsSearchingCore method, 430–431
  - SupportsSorting property, IBindingList, 317
  - Synchronization
    - of data between controls, 96–97
    - data paths within data sources, 94–96
    - ICollection interface, 297
    - of many-to-many relationships, 172–174
    - updating to parent data-bound controls, 168–172
  - System namespace, 550
  - System.Collections namespace, 19–20
  - System.ComponentModel namespace, 20
  - System.Data namespace, 605
  - System.Transactions, 647–649
  - System.Windows namespace, WinFx, 525, 527
  - System.Windows.Controls namespace,
    - WinFx, 526
  - System.Windows.Windows.Window objects,
    - WinFx, 525
- T**
- Tab order, controls, 596–598
  - TabIndex property, 597
  - Table adapters, 52–77
    - adding helper data access methods, 66–67
    - adding queries to, 69–72
    - adding transaction support to, 62–66
    - basing on stored procedures or views, 67–69
    - configuring to use stored procedures, 74–77
    - connection management, 58–62
    - defined, 35, 41
    - designer-generated code, 55–56
    - filling and updating typed data sets, 56–58
    - generating data-bound controls, 11
    - overview of, 52–56
    - using Query Builder for SQL statements, 72–73
  - Table-oriented complex data binding, 199
  - TableAdapter Query Configuration wizard, 67–72
  - TableLayoutPanel control, 595–597
  - TableMappings property, 618
  - Tables
    - adding to data sets, 48–49
    - creating Windows Forms application, 553
    - loading data sets from database, 616–619
    - presenting tabular data. *See* DataGridView control
  - Tag property, DataGridViewCell, 245
  - Templates
    - binding collection to grid with, 541–543
    - CellTemplate property, 246, 252, 274–275
    - FormView, 515
  - Test container, 352–353
  - Testing, custom controls, 350
  - Text property
    - ASP.NET data-binding expressions, 497
    - Button control, 580
    - CheckBox control, 580–581
    - Label controls, 580
    - RichTextBox control, 582
    - TextBox control, 581
    - TreeView control, 585
  - TextBlock control, WinFx, 526, 530–531
  - TextBox control
    - AutoComplete, 160–162, 371–375
    - binding DateTime column to, 144–145
    - binding numeric column to, 145–147
    - DataGridViewTextBoxColumn in, 249–250
    - as default control type, 190
    - icon, 191
    - overview of, 581
    - setting data bindings, 93–96
    - using data-binding expressions in, 497
    - in WinFx applications, 530–531, 535
    - in XAML, 540
  - TextChanged event, 101
  - TextReader, 664–665
  - Textual controls, 413–414
  - Third-party components
    - adding control to Visual Studio Toolbox, 199
    - adding outside data source, 202
    - designing own data-bound controls from, 21
    - working with in Toolbox, 555–557
  - This property, IDataErrorInfo, 331
  - This property, IList, 299
  - ThreeState property, CheckBox controls, 580–581





ThreeState property, DataGridViewCheckBoxColumn, 251

Tiers, layers vs., 23–26

Toolbox, Visual Studio

- adding custom controls to, 350–352
- creating Windows Forms, 555–557
- customizing, 199–200

ToolStrip control

- adding with designer, 212–213
- adding with Search Criteria Builder dialog, 212
- BindingNavigator control as version of, 103
- overview of, 598–600

ToolTipText property, DataGridViewCell, 245

ToolTipText property, DataGridViewColumn, 248

Tostring method, ListBox control, 582–583

Tostring method, overriding, 413–414

Transacted object editing, 416–420, 439–441

Transactions

- adding support to table adapters, 62–66
- client-side, 650–651
- overriding to add sorting to collection, 432–433
- scoping with System.Transactions, 647–649
- working with in ADO.NET, 643–647

TransactionScope class, 62, 648–649

TreeView control, 518, 585

try-catch block, 645–646

try-finally block, 620, 646–648

TryParse method, 103

Two-way data binding, 82–83

Type conversions

- automatic formatting and parsing, 147–148
- binding events vs., 148–154
- overview of, 138–139

Type-safe indexer, 37–38

Type safety, 34–37

Typed data sets, 31–52

- creating, 41–42
- creating with command line tools, 77–78
- creating with Data Set Designer, 42–49
- designer-generated code files for, 203
- displaying in Data Source window, 189
- filling and updating with table adapter, 56–58
- generated code, 49–52
- generating table adapters. *See* Table adapters
- internals of, 37–41

- review of, 31–34
- safety of, 34–37
- using in code, 78–79

TypeInfo, data source files, 203–204

TypeName property, ObjectDataSource control, 506

TypeUISetting, data source files, 204

## U

Unbound columns, 226–232

- adding to grids, 264–265
- defined, 218, 227
- generating computed columns, 232
- overview of, 226–227
- using CellFormatting event, 227–231
- using RowsAdded event, 231–232

Unchanged method, client-side transactions, 650–651

Update method, data sets, 33

Update method, table adapters, 54

UpdateCoordinates method, bar chart controls, 384

UpdateDataBinding method, data-bound custom controls, 387–389, 396

Updates

- ADO.NET, handling concurrency during, 628–631
- ADO.NET, stored procedures, 632–637
- ADO.NET, with data sets, 624–628
- coordinates on bar chart controls, 384
- data bindings on data-bound custom controls, 387–389, 396
- of data sources, 106–108
- to parent data-bound controls, 168–172
- typed data sets with table adapters, 56–58

Use Windows Authentication, 8–9

UserControl base class, 586–588

UserControl class, 569

Users, validation, 457

using block, 620, 646–648

using statement, 78–79

## V

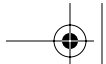
Val property, virtual mode, 237–238

Validate method, 463

ValidateChildren method

- using AutoValidate property, 472
- validation down control hierarchy, 473
- validation up control hierarchy, 463



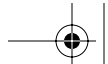


## 694 ■ INDEX

- Validated events
    - DataGridView, 462–463
    - defined, 459
    - overview of, 459–461
    - setting error provider message in response to, 465
  - Validation, 457–485
    - calling EndEdit and, 158
    - capturing data errors on data sets, 475–479
    - data concurrency resolution, 483–484
    - DataGridView DataError event, 468–471
    - DataGridView error displays, 467–468
    - DataGridView events, 462–463
    - down control hierarchy, 472–473
    - ErrorProvider control and, 464–467
    - for extended controls, 474–475
    - handling events, 459–462
    - IDataErrorInfo and, 479–482
    - overview of, 479–482
    - up control hierarchy, 463
    - updating data sources, 107–108
    - using AutoValidate property, 471–472
    - Windows Forms, 458–459
  - ValidationConstraints enumeration values, 473
  - Value property, ConvertEventArgs, 149–150, 156
  - Value property, DataGridViewCell, 245
  - ValueChanged event, DateTimePicker, 582
  - ValueMember property, ComboBox control, 85–86, 88–90
  - ValueMember property, DataGridViewComboBoxColumn, 254
  - ValueType property, DataGridViewColumn, 248
  - Vector graphics, WinFx, 523
  - View Code, 526
  - Views
    - basing table adapters on, 67–69
    - binding to data table really binding to data, 113
    - typed datasets defining, 33
  - ViewState, 495–496
  - Virtual mode, DataGridView, 233–241
    - choosing data sets or custom collections, 237–238
    - defined, 218
    - initializing grid, 238–239
    - overview of, 233–234
    - setting up, 234–237
    - summary, 240–241
    - understanding, 239–240
  - Visible property, DataGridViewCell, 245
  - Visible property, DataGridViewColumn, 248
  - Visual Studio 2005, 4–8
  - Visual Studio Designer
    - defining grids, 263–266
    - design-time declarative programming, 378
    - designer-generated code, 563–566
    - initialization, 334–337, 359–360
  - Visual Studio Designer, bound controls, 177–216
    - adding data sources, 179–180
    - adding database data source, 181–185
    - adding object data source, 186–189
    - adding Web service data source, 185–186
    - choosing data source type, 180
    - code, 202–205
    - customizing, 196–199
    - from data sources, 189–196
    - data sources files, 202–205
    - with Data Sources window, 177–179
    - generating data-binding code for existing controls, 199–201
    - generating other data-binding code, 205–206
    - master-details bound controls, 214–216
    - Properties window settings, 206–210
    - selecting type, 196
    - summary, 216
    - using smart tags, 210–214
  - Visual Studio Options dialog, 197–198
  - Visual Studio, Windows Forms applications, 554–563
    - adding members to form, 557–561
    - creating empty project, 554
    - hooking up event handler and data binding, 561–563
    - working with Toolbox, 555–557
- ## W
- Web reference folder, 202–203
  - Web reference icon, 191
  - Web servers, validation controls, 483–484
  - Web service data sources, 185–186, 202–203
  - Weinhardt, Michael, 475
  - WHERE clause, SqlDataSource control, 515
  - Width property, DataGridViewColumn, 248

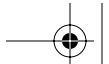






- WindowLoaded event handler, XAML, 540–541
  - Windows class, WinFx, 527, 539–540
  - Windows Communication Foundation (Indigo), 521
  - Windows Forms applications, building
    - data-bound
      - adding new data sources and data connection, 6–8
      - creating project, 4–5
      - customizing data source control mappings, 10–11
    - data binding, 2–3
    - data-bound controls, 20–21
    - data objects and collections, 16–18
    - data sets, 18–20
    - data sources, 15–16
    - generating data-bound controls, 11–12
    - layered application architecture, 21–27
    - overview of, 3–4
    - running, 13–14
    - selecting data objects, 8–10
    - smart clients, 27–28
  - Windows Forms applications, programming, 547–600
    - architecture, 567–570
    - binding data reader to, 607
    - commanding and controlling, 598–600
    - containing forms with parent forms, 577–578
    - creating custom user control, 586–588
    - data application in, 548–554
    - data binding improvements in, 3
    - data sources in, 16
    - designer-generated code, 563–566
    - displaying other forms, 576–577
    - handling control events, 574–576
    - layout of, 589–596
    - main method, 570–574
    - overview of, 547–548
    - tab order settings, 597–598
  - Windows Forms, data display controls, 578–586
    - Button, 580
    - CheckBox, 580–581
    - ComboBox, 584
    - DataGrid, 585–586
    - DataGridView, 586
    - DateTimePicker, 582
    - label, 579–580
    - ListBox, 582–584
    - ListView, 584
    - overview of, 578–579
    - PictureBox, 585
    - RadioButton, 581
    - RichTextBox, 582
    - TextBox, 581
    - TreeView, 585
  - Windows Forms Programming in C# (Sells)*, 378
  - Windows Forms validation, 458–459
  - Windows Forms, with Visual Studio, 554–563
    - adding members to form, 557–561
    - creating empty project, 554
    - hooking up event handler and data binding, 561–563
    - working with Toolbox, 555–557
  - Windows Presentation Foundation (WPF), 521
  - WinFx applications, 521–545
    - binding collection to grid with templates, 541–543
    - control styling in, 543–545
    - data binding, 532–535
    - data contexts and data sources, 536
    - overview of, 521–522
    - UP programming and capabilities, 522–525
    - writing, 525–531
    - XAML and, 537–541
  - WithEvents keyword, 576
  - WPF (Windows Presentation Foundation), 521
  - WriteXML method, 609–611
  - .wsc file, 203
- ## X
- XAML (Extensible Application Markup Language)
    - binding collection to grid with templates, 541–543
    - control styling in WinFx, 543–545
    - overview of, 537–541
    - programming WinFx applications, 524–525
  - XDR (XML Data Reduced) schemas, 78
  - XML data access, ADO.NET, 659–663
    - loading data into XPathDocument, 664–665
    - navigating XML document, 667–669
    - overview of, 658–659
    - querying XML data, 665–667
  - XmlDataDocument class, 659–663
  - XPathDocument class, 663–664





696  INDEX

- XML Data Reduced (XDR) schemas, 78
- XML, data source files, 202–205
- XML Schema Support utility (xsd.exe), 77–78
- XmlDataDocument class
  - defined, 658
  - obtaining XPathNavigator for, 664
  - searching data sets with, 638
  - working with, 659–663
- XMLDataSource control, 507–508, 518
- XmlDocument class
  - defined, 658
  - navigating, 667–669
  - obtaining XPathNavigator for, 664
  - XmlDataDocument deriving from, 659
- XmlReader class, 659, 664–665
- XmlWriter class, 659
- XPathDocument class, 663–665
- XPathNavigator
  - navigating XML documents, 667–669
  - obtaining for XPathDocument, 664
  - querying XML data, 665–667
- XPathNodeIterator class, 666–667
- .xsc file, 204–205
- XSD (XML Schema Definition file)
  - configuring data source, 184
  - converting XDR schemas to, 78
  - creating typed data sets with, 42
  - typed data set-generated code and, 49–50

