

# Index

## A

**About This Form Help menu command, 118**

**Access, Outlook data in, 264–269**

- ADO, 265–266
- DAO, 267–269
- DAO DBEngine, 268–269
- overview of, 264–265

**Access Expression Service, 36**

**Account object**

- checking for existing contact, 100
- ItemSend event using, 70
- methods/properties/events, 378
- overview of, 28–29

**AccountRuleCondition, 378**

**Accounts collection**

- checking for existing contact, 100–102
- methods/properties/events, 378
- overview of, 28–29

**AccountType property, 29**

**Activate event**

- CommandBar user interface, 210
- DoStartup in Explorer, 203–205
- ExplWrap class in C#, 343–344
- ExplWrap class in VB.NET, 340–341
- Inspector menus and toolbars, 198
- InspWrap class in C#, 356–358
- InspWrap class in VB.NET, 351–353
- workaround for, 284

**ActiveExplorer object**

- checking for existing contact, 100–101
- in Explorer.Close event, 155–156, 161–162, 316–317, 328
- overview of, 10–11
- retrieving, 49–50

**ActiveInspector object**

- as currently active displayed tem, 49–50
- Inspector.Close event, 156, 164, 317–318, 329–331
- overview of, 10–11
- simulating form environment in VBA, 124

**ActiveWindow, 49–50**

**ActiveX Data Objects (ADO), 265–266**

**Add Child Task control, 306**

**Add method**

- custom views, 248–249
- CustomTaskPanels collection, VSTO, 213

Documents collection, Word, 254

Items collection, 50

Pages collection, 185–187

**AddChildTask method**

- Connect class, VB.NET, 312–314
- InspWrap, C#, 359–361
- InspWrap, VB.NET, 349–351

**AddExpl method**

- Explorer wrappers in C#, 167–168
- Explorer wrappers in VB.NET, 165
- OutExp in VB.NET, 336–337
- OutExpl in C#, 337–338

**AddInsp method**

- defined, 165
- Inspector wrappers in C#, 174–176
- Inspector wrappers in VB.NET, 171–172
- OutInsp in C#, 345–346
- OutInsp in VB.NET, 343–345

**AddressBook method, CDO 1.21, 27**

**AddressEntry object**

- checking for existing contact, 100–102
- new properties/methods, 410
- using RDOAddressEntry object, 290–291, 294
- using restricted properties/methods with, 35

**AddressList object**

- checking for existing contact, 101
- new properties/methods, 410
- picking names, 55–57

**AddressRuleCondition, 378–379**

**ADO (ActiveX Data Objects), 265–266**

**AdvancedSearch object**

- creating search folders from, 106
- DASL syntax for, 37
- defined, 14–15
- JET syntax not supported in, 37
- SQL syntax used with, 21
- working with, 74–77

**Affronti, Michael, 429**

**AllContactsToExcel method, 258–259**

**Alt+F11, 2, 4**

**AND operator, AdvancedSearch method, 75–76**

**APIs (application programming interfaces), alternate**

- CDO, 288–289
- Redemption, 289–303

**AppDomain namespace, 280–281**

## Application events

### Application events, 69–77

AdvancedSearch method, 74–77  
 ItemLoad, 71–72  
 ItemSend, 70–71  
 NewMailEx, 72–74  
 Startup, 69–70

### Application object, 39–53

BeforeFolderSharingDialog event, 48  
 context menus, 45–48  
 CreateItem and CreateItemFromTemplate methods, 50  
 DefaultProfileName property, 45  
 defined, 7  
 Explorers collection, 49–50  
 GetObjectReference method, 44–45  
 initializing globally trusted, 40–43  
 IsTrusted property, 45  
 ItemLoad event, 48–49  
 ItemSend event, 52–53  
 MAPILogonComplete event, 52  
 new properties/methods, 410  
 NewMail and NewMailEx events, 50–51  
 overview of, 39–40  
 Quit event, 52  
 Reminder event and Reminders collection, 51  
 Startup event, 52  
 Version property, 53

### AppointmentItem object, 410, 415–416

#### architecture, Outlook, 5

#### AssignToCategoryRuleAction, 379

#### Attachment object

adding or removing from emails, 71  
 context menus, 45  
 new properties/methods, 411  
 using code window, 121–122  
 working with custom form regions, 245

#### AttachmentContextMenuDisplay event, 45

#### AttachmentSelection, 379

#### Auto\_Open macro, Excel, 257

#### AutoDiscoverXML property, Exchange, 54

#### AutoFormatRule, 379

#### AutoFormatRules collection, 380

#### AxHost class

getImage callback using, 232–233  
 overview of, 215  
 using Connect class in VB.NET, 318–319  
 using in C#, 216–217  
 using in VB.NET, 216  
 VB.NET vs. C# and, 319

## B

#### BeforeFolderMove event, 77, 79

#### BeforeFolderSharingDialog event, 48

#### BeforeItemMove event, 77–79

#### behavior, forms, 138–139

#### binary properties, PropertyAccessor, 282–283

#### BinaryToString method, PropertyAccessor, 24, 34

#### blogs, Outlook and Office 2007, 428–430

#### Body property

PropertyAccessor limitations, 17, 283  
 security and, 34  
 setting StorageItem, 26

#### Bookmark fields, Word, 254

#### Browser Control Form, 269–272

#### BusinessCardView, 248, 380

#### button controls. *see* CommandBarButton control;

#### CommandButton control; toggleButton controls

#### Byrne, Ryan and Randy, 428

## C

### C# code

AxHost in, 216–217  
 COM addins in this book using, 147  
 communicating with COM addin in, 190–191  
 communicating with COM addin in VSTO, 194–196  
 custom form regions in, 243–244, 246–247  
 custom task panes in, 237–238, 239–240  
 displaying property pages in, 186–187  
 Explorer wrappers in, 167–171  
 IDExtensibility2 in, 150–151  
 initializing trusted Application/NameSpace objects  
 in, 41–43  
 Inspector wrappers in, 174–179  
 interface handlers in VSTO, 214–215  
 IRibbonExtensibility in, 222–223  
 ItemSend in, 52–53  
 missing parameter values in, 48  
 OnDisconnection Catch-22 bug in, 158–164  
 OnDisconnection Catch-22 bug in VSTO, 164  
 Ribbon callbacks in, 230–234  
 utility code in, 182–185

### C# code, Task Management

Connect class, 319–335  
 ExplWrap class, 341–344  
 InspWrap, 353–362  
 MyAxHost class, 335–336  
 OutExpl, 338–339  
 OutInsp, 345–347  
 Ribbon XML, 373  
 TaskPane, 366–371

### cache, forms, 115, 136

#### CalendarModule, 381

#### CalendarSharing, 381

#### CalendarView, 248, 381–382

#### callbacks. *see* Ribbon callbacks

#### Cancel argument, 79

#### Caption property, property pages, 186

#### CardView, 248, 383

#### CAS security, managed code addins, 280–281

**Categories collection**

- creating context menu rule, 98
- methods/properties/events, 383–384
- overview of, 54–55

**Category object**

- methods/properties/events, 384
- overview of, 54–55

**CategoryRuleCondition, 384****CDO**

- as alternate API, 288–289
- avoiding PropertyAccessor limitations with, 17
- Outlook 2007 workarounds using 1:21, 284
- Redemption replacing, 289

**certificates**

- creating code-signing, 3–4
- deploying unmanaged code addins using, 280
- security levels and, 61

**Change Large Icon button, 127****Change Small Icon button, 127****CheckBox control, 125, 141–144****ChildTag string, C# Connect class, 319–321****ChildTasks property, 306****CI search syntax, 36–38****CI\_PHRASEMATCH operator, 38****CI\_STARTSWITH operator, 38****classes**

- simulating form environment in VBA, 123–124
- user, 62–63, 69
- UserForms as, 64–68
- wrapper. *see* wrapper classes

**Click event handlers, WordMail, 285–286****ClickOnce, VSTO addins, 281****Close event. *see also* Explorer.Close event**

- Explorer wrappers in C#, 168–171
- Explorer wrappers in VB.NET, 166–167
- ExplWrap class in C#, 343–344
- Inspector and Explorer, 152
- Inspector wrappers in C#, 176, 178–179
- Inspector wrappers in VB.NET, 172–173
- InspWrap class in VB.NET, 356–358
- OnDisconnection Catch-22 bug, 156–157, 162–164
- Outlook 2003 WordMail workarounds, 284
- VB.NET ExplWrap class, 340–341

**clsCategorizeRule class, 62–64, 94****clsContactWrap class, 63, 71–72, 85–88****clsEmailWrap class, 63, 82–85****clsMailWrap class, 80–85****clsSearch class, 76–77****code modules, inserting into VBA project, 63–64****code window, forms, 120–122****code-signing certificates**

- managed code addins, 281
- in Outlook VBA, 3–4
- unmanaged code addins, 280

**colContactWrapper collection, 63, 72, 86****colMailWrapper collection, 63, 82****ColumnFormat, 385****Columns**

- adding and removing from tables, 31
- checking today's to-do-list, 102–103
- methods/properties/events, 384–385
- using Microsoft Excel for Outlook data, 260–261

**Columns collection**

- checking today's to-do-list, 104
- methods/properties/events, 385
- not adding computed property values to, 34

**COM addins. *see also* OnDisconnection Catch-22 bug**

- C# utility code, 182–185
- displaying Outlook property pages, 185–187
- distributing VBA projects using, 109–110
- Explorer and Inspector event handlers, 152–155
- IDTExtensibility2 and trusted, 148–151
- managed vs.unmanaged code, 147–148
- overview of, 147
- programming Ribbon with, 97
- VB.NET utility code, 180–182

**COM addins, and Outlook user interface, 197–251**

- AxHost, 215–217
- custom task panes, 234–240
- custom views, 248–251
- form regions, 240–247
- menus and toolbars in C#, 206–212
- menus and toolbars in Explorers, 198–199
- menus and toolbars in VB.NET, 199–206
- VSTO interface handlers, 213–215
- working with Ribbon. *see* Ribbon

**COM addins, communicating with, 188–196**

- in C#, 190–191
- in C# with VSTO, 194–196
- overview of, 188
- testing, 189
- in VB.NET, 188–189
- in VB.NET with VSTO, 191–193

**COM addins, OnDisconnection Catch-22 bug, 155–179**

- C# with VSTO, 164
- Explorer wrappers and, 164
- Explorer wrappers in C#, 167–171
- Explorer wrappers in VB.NET, 165–167
- Inspector wrappers and, 164
- Inspector wrappers in C#, 174–179
- Inspector wrappers in VB.NET, 171–174
- overview of, 155–157
- VB.NET with VSTO, 157–164

**COM addins, real-world Outlook programming**

- for compatibility, 276–277
- deploying managed code, 280–282
- deploying unmanaged code, 279–280
- Outlook 2007 limitations, 282–283
- possible Ribbon workaround problems, 277–278
- supporting multiple versions of Outlook, 275–276

## CommandBar objects

### CommandBar objects

- creating context menu, 47
- creating context menu rule, 94–97
- creating in C#, 206–212
- creating in VB.NET, 199–206
- menus and toolbars in Explorers, 198
- Outlook 2003 WordMail workarounds, 285–287

### CommandBarButton control

- creating context menu rule, 94–95
- creating context menus, 46–47
- creating new, 201–202, 208–209
- creating Tag properties for, 204
- declaring class-level instances of, 204, 210
- icons for, 205
- Outlook 2003 WordMail limitations, 283
- PasteFace method, 205
- usage in user interface, 198–199

### CommandBarComboBox control

- creating, 202–203
- creating new, 209–210
- usage in user interface, 198–199

### CommandBarControl

- creating context menu rule, 96–98
- defined, 97
- unique Tag values for, 201

### CommandBarPopup control

- adding to menu, 200–201
- creating new, 207–208
- declaring class-level instances of, 204, 210
- defined, 198
- Tag properties for, 204
- usage in user interface, 198–199

### CommandBars collection

- adding user interface to Inspector, 284–285
- data display, 10
- locating CommandBar object, 47
- menus and toolbars as members of, 97, 198
- in VB.NET, 199

### CommandButton control

- adding to Browser Control Form, 270
- adding to NoSubjectForm, 67
- adding to ToDoForm, 68

### <commands> tag, Ribbon, 219

### CompareEntryIDs method, Namespace, 53–54

### comparison operators, JET query syntax, 37

### Connect class

- communicating with COM addin in C#, 190–191
- communicating with COM addin in VB.NET, 189
- tasks management. *see* Task Management System,
  - Connect class in C#; Task Management System,
  - Connect class in VB.NET

### constants, VBScript vs. VBA form code, 124

### ContactItem object

- ItemLoad event, 72
- new properties/methods, 411, 416–417

- returning from AddressEntry, 101
- Unload event, 86

### contacts wrapper class, 85–88

### ContactsModule, 385–386

### ContactToExcel method, 258

### context menus

- creating rule, 94–99
- overview of, 45–48

### ContextMenuClose event, 46

### Control Toolbox

- adding WebBrowser control to, 269
- creating form region, 141–144
- working with, 119–120

### controls, form

- Active X. *see* OCXs (ActiveX controls)
- adding from Control Toolbox, 119–120
- adding to custom forms, 114–115
- using Field Chooser to bind, 118–119
- using form regions, 139–144

### controls, Ribbon customization rules, 219–220

### Create method, Rules collection, 92

### CreateItem method, Application object, 50

### CreateItemFromTemplate method, Application object, 50

### CS\_Action callback, Ribbon callbacks, 331–335

### CTPFactoryAvailable event, Connect class, 310, 335

### CType method, VB TaskPane, 365–366

### CurrentProfileName property, Namespace, 45, 54

### customization dialog, 76

### CustomizationContext, 284–285

### CustomPropertyChange event handler, 134–135

### CustomTaskPane object, Office, 213

- InspWrap in C#, 353, 360

- InspWrap in VB.NET, 347, 353

- OutInsp in C#, 346

- OutInsp in VB.NET, 343

- VSTO CustomTaskPanes collection vs., 213

- working with custom task panes, 239

### CustomTaskPanes collection, VSTO, 213–214, 235

## D

### DailyTaskListView, 248

### DAO, Access, 267–269

### DAO DBEngine, 268–269

### DASL (DAV Searching and Locating) query syntax, 35–38

- AdvancedSearch method, 74–77

- CI syntax, 37–38

- defined, 14

- for filters, 76, 249–250

- JET query syntax, 36–37

- namespaces, 36

- overview of, 35–36

- syntax, 37

- working with, 16

**data**

- accessing with tables, 9–10
- display, 10–11
- storage, 8–10

**DataReady event, 84****date properties, PropertyAccessor, 24****DAV Searching and Locating. see DASL (DAV Searching and Locating) query syntax****debugging, form code using VBA, 124****DefaultProfileName property, Application object, 45****Deleted Items folder, previous versions of Outlook, 77****DeleteProperties method, PropertyAccessor, 22–23****DeliveryReceipt property, workarounds for, 287****Designer class, 279****Developer tab, Ribbon**

- adding controls to form, 125
- enabling in Office applications, 114
- forms development environment, 116–118
- using code window, 120–122

**development environment**

- forms, 116–122
- VBA form code for, 124

**digital certificates. see code-signing certificates****digital signatures**

- creating code-signing certificate, 3–4
- deploying unmanaged code addins, 280
- disabling startup macro warnings, 61

**disclaimers, email, 52–53****display, data, 10–11****DistListItem, 411–412****DockPositionChange event, InspWrap, 353, 361–362****DocumentItem, Office supporting, 22****DoStartup procedure, 203–205, 210–211****downloads, Microsoft, 430–432****DueDate property, to-do-list, 103****E****early binding, 276–277****email wrapper class, 82–85****Encrypted property, workarounds for, 287****EntryID property, 10, 53–54****error arrays, accessing, 23****error handling, VBScript vs. VBA form code, 124****Excel, using Outlook data in, 257–264****Exchange**

- checking for existing contact, 102
- CompareEntryIDs method, 53–54
- connecting Outlook to, 54
- mailboxes as stores, 24
- properties, 54
- Public Folders Store, 292
- publishing forms in, 113, 129
- SDK, 435
- StorageItems in public folders and, 25–26

**ExchangeConnectionMode property, Namespace, 54****ExchangeDistributionList, 386****ExchangeMailboxServerName property, 54****ExchangeMailboxServerVersion property, 54****ExchangeUser, 386–388****Explorer object**

- Connect class in VB.NET, 316
- defined, 49
- Explorer.Close event, 327, 328
- NavigationPane, 27
- new properties/methods, 412
- overview of, 10–11
- setting up Explorer event handlers, 153–154

**Explorer.Close event**

- Connect class handling, 307
- Connect class in C#, 327–328
- Connect class in VB.NET, 316–317
- ExplWrap class in VB.NET, 340–341

**Explorers collection**

- Application object, 49–50
- custom task panes, 234–240
- overview of, 10–11
- setting up Explorer event handler, 152–155
- Tag properties, 47
- wrappers, overview of, 164–165
- wrappers in C#, 167–171
- wrappers in VB.NET, 165–167

**ExplWrap class**

- OutExpl in C#, 338–339
- OutExpl in VB.NET, 337–338
- task management in C#, 341–344
- task management in VB.NET, 340–341

**Export function, macros, 110****Extended MAPI, 34**

- difficulties of using, 34
- ItemLoad event and, 48
- MAPI mailing list for, 436
- MAPI viewer for, 434–435
- MAPItable filters, 295–298
- PropertyAccessor limitations and, 17
- setting folder properties for forms in early versions of Outlook, 137
- software development kits for, 433
- untrusted code using, 35
- using CDO vs., 288
- working with standard forms, 112

**F****F2, 4****F5, 109****Family folder, 91, 93****Field Chooser, 118–119, 125****Fields designer, viewing XML in, 250–251****filter definition window, 20–21**

## filters

### filters

- checking today's to-do-list, 103–104
- custom views, 249–250
- MAPITable, 295–298
- searches, 74–77

### FindControl method, 200

### FindExchangePublicFoldersStore method, 292

### FindFirst method, MAPITable filter, 297–298

### FindNext method, MAPITable filter, 297–298

### FlagStatus property, 73–74

### folder events, 77–79

### Folder objects

- checking today's to-do-list, 104
- creating context menus, 46
- creating custom views, 248–249
- data display, 10
- data storage, 8–9
- events, 69, 77–79
- Folder collections of, 9
- MAPIFolder objects and, 6
- new properties/methods, 412
- overview of, 6–7
- picking folders, 55
- PropertyAccessor syntax 1, 16
- retrieving StoreID, 24
- setting target folder for Family rule, 93
- workarounds for hidden items, 282

### Folder property pages, 186

### FolderContextMenuDisplay event, 46

### FolderGetTable method, 31

### Folders collection

- accessing data stores using Namespace, 9
- of Folder objects, 9
- overview of, 6
- setting target folder for Family rule, 91

### For loop, 199

### form coding, 39

### form regions

- creating, 141–144
- implementing with COM addins, 240–247
- interface support with VB.NET in VSTO, 213–214
- location, 140–141
- overview of, 139
- properties, 140–141
- registry settings, 140–141

### formatting

- custom views, 249–250
- data for printing, 254–257
- images for CommandBarButton, 211–212

### FormDesign property, 114

### FormNameRuleCondition, 388

### FormRegion, 388–389

### FormRegionStartup, 389

### forms, 111–145

- behavior of, 138–139
- custom, 124–128

development environment, 116–122

managing, 136–137

OFT files and, 112

prototyping code in VBA, 122–124

publishing, 128–132

standard, 111–115

testing code for, 132–135

using form regions, 139–144

### Forms Manager, 136

### forums, Outlook, 438

### Framework, 280

### Free/Busy data, 294

### FromRssFeedRuleCondition, 389

### functions, as VBA procedure, 60

## G

### GAC (Global Assembly Cache), 276

### garbage collector

- Explorer/Inspector wrapper classes and, 319
- GetObjectReference method and, 44
- keeping references alive and safe from, 81, 152, 158, 316, 327
- New Explorer/Explorer\_Close event handlers, 325
- programming Outlook on .NET for shutdown, 43
- speeding up call to shutdown, 181
- Teardown method, 313, 324

### GetArray method, tables, 31

### GetContact method, 101

### GetCustomUi callback, 278

### GetDefaultFolder ( ) method, 7

### GetFreeBusy method, RDOAddressEntry object, 294

### GetIdsFromNames method, Redemption, 298–301

### getImage callback

- in C#, 232
- in C# MyAxHost class, 335–336
- for controls displaying image, 227

### getLabel callback, 228

### GetObjectReference method, 44–45

### getPressed callback, 227, 232

### GetProperties method, PropertyAccessor, 22–23

### GetRow method, MAPITable filter, 295–297

### GetSelectNamesDialog, 55–57

### GetStorage method, 25–26, 284

### getSuperTip callback, 228

### GetTodaysToDoList code, 102–105

### GetValues method, rows, 31

### getVisible callback, 228, 233

### Global Assembly Cache (GAC), 276

### Globally Unique Identifiers (GUIDs), 10, 18–21

### Globals code module, VB, 337

### Gregg, Ryan, 428

### groups, 218, 220

### GUIDs (Globally Unique Identifiers), 10, 18–21

**H**

**Harris, Jensen, 429**  
**Help file, 14**  
**hexidecimal notation, 17**  
**hidden items, workarounds, 282, 284**  
**High Importance search folder, 107**  
**Hill, David, 430**  
**HKEY\_CURRENT\_USER, 279–280**  
**HKEY\_LOCAL\_MACHINE, 279**  
**HTML files, 269–272**  
**HTML toggleButton, 225**

**I****icons**

CommandBarButton controls, 205, 211–212  
 creating for forms, 127  
 supplying for controls with getImage callback, 227–228

**IconView, 248, 389–390****ICustomTaskPaneConsumer interface**

Connect class implementing, 307  
 creating custom task panes in C#, 237–238  
 CTPFactoryAvailable event in C#, 335  
 CTPFactoryAvailable event in VB.NET, 310

**IDTEExtensibility2 interface, 148–151**

addin templates, 151  
 Connect class implementing, 307  
 Connect class in C#, 321–325  
 Connect class in VB.NET, 308–310  
 deploying unmanaged code addins, 279  
 implementing in C#, 150–151  
 implementing in VB.NET, 149–150  
 overview of, 148–149  
 version differences, 149

**Import function, 110****Importance items, High, 31****Importance search folder, 107****ImportanceRuleCondition, 93, 390****Incomplete Tasks search folder, 105–106****Init method, TaskPane, 362–365, 366–370****InitHandler method, Connect class, 314–316, 325–327****InitHandler procedure, utility code, 180, 183****Inspector object**

Close events, 153, 156  
 new properties/methods, 412

**Inspector.Close event, Connect class, 307, 317–318, 329–331****Inspectors collection**

activating menus and toolbars in, 198  
 CommandBar objects and controls, 97  
 custom task panes, 236–240  
 displaying Outlook items, 10–11, 49  
 Inspector wrappers, 164, 171–179  
 IRibbonExtensibility in VB.NET, 220–222  
 Ribbon callbacks, 224–229, 312–314, 331–335

setting up Inspector event handler, 152–155  
 Tag properties, 47  
 using NewInspector event to instantiate wrapper class, 81–82  
 WordMail workarounds, 283–287

**InspWrap class**

Inspector wrappers in C#, 174–179  
 Inspector wrappers in VB.NET, 173–174  
 task management in C#, 353–362  
 task management in VB.NET, 347–353

**InstallShield, 279****interface, VBA editor options, 4****interface handlers, VSTO addins, 213–215****Internet, and custom forms, 115****Internet Explorer, Browser Control Form, 272****Inter-Personal Message (IPM), forms, 112****Invalidate method, 220, 234****InvalidateControl method, 220, 225–226, 234****IPictureDisp objects**

AxHost workaround for, 215–217  
 for button icons, 205, 211  
 designing form regions, 241  
 getImage callback and, 227–228, 232–233  
 using C# MyAxHost class, 335–336  
 WordMail limitations, 283

**IPM (Inter-Personal Message), forms, 112****IRibbonExtensibility interface**

Connect class implementing, 307  
 implementing in C# code, 222–223  
 implementing in VB.NET, 220–222  
 supplying Ribbon XML, 220  
 supporting multiple versions of Outlook, 277–278

**IRibbonUI property**

Connect class in C#, 331–335  
 Connect class in VB.NET, 311–314  
 OutInsp in C#, 346–347  
 OutInsp in VB.NET, 344–345

**IsInstantSearchEnabled property, Store object, 36–38****IsMarkedAsTask property, contacts wrapper class, 87–88****IsTrusted property, Application object, 34, 45****IsWordMail property, 284****Item argument, BeforeItemMove event, 79****Item event handlers, 124****Item object, 39, 412****ItemAdd events, limitations of, 78, 282****ItemContextMenuDisplay event, 46, 97–98****ItemLoad event, Application object**

overview of, 48–49  
 using contacts wrapper class, 85–86  
 working with, 71–72

**items**

Body property, 17  
 creating, 50  
 in folders, 6–7  
 Inspectors collection displaying, 10–11

## items (continued)

### items (continued)

monitoring with NewMailEx, 50–51  
 overview of, 8  
 Reminder, 51  
 setting up Inspector event handler, 152–155  
 StorageItems, 25–26

**ItemSend event, Application object, 52–53, 70–71**

## J

**Jausovec, Peter, 429–430**

**JET query syntax, 36–37**

**JournalModule, 390**

## K

**Key property, 204, 337–338**

**keywords, CI search, 38**

### KillExpl method

Explorer wrappers in C#, 171  
 Explorer wrappers in VB.NET, 167  
 ExplWrap in C#, 343–344  
 OutExpl in C#, 338–339  
 OutExpl in VB.NET, 337–338

### KillInsp method

Inspector wrappers in C#, 179  
 InspWrap in C#, 356–358  
 OutInsp in C#, 346–347  
 OutInsp in VB.NET, 344–345

## L

**Label control, NoSubjectForm, 66**

**late-binding, 276–277**

**licensing, Active X controls, 120**

**ListBox control, 68**

**LocalTimeToUTC method, PropertyAccessor, 24**

### location

of form region, 140–141  
 of Inspector menus and toolbars, 197–198  
 of installed COM addins, 279  
 of published forms, 113, 129

**Location property, Appointments, 345**

**Location property, Contacts, 421**

**Logoff method, RDOSession, 289–290**

**Logon method, RDOSession, 290**

## M

**Macbeth, Melissa, 429**

**macro Subs, 60, 63–64**

**macros, 2, 4**

**Macros dialog, 108–109**

**MailItem events, 83**

**MailItem object, 413**

**MailModule, 390–391**

### managed code

avoiding using CDO for, 288  
 deploying addins, 280–282  
 developing addins for, 276  
 reflection for, 276–277  
 vs. unmanaged, 147–148

**management, forms, 136–137**

**MAPI (Message Application Programming Interface)**

creating X-header for email in Redemption, 293  
 mailing list, 438  
 Outlook using for data access, 6  
 properties, 14  
 viewer tools, 434–435

**MAPIFolder object, 6**

**MAPILogonComplete event, 52**

**MAPIOBJECT property, Redemption, 289–290**

**MAPITable filters, Redemption, 295–303**

**MAPIUtils object, Redemption, 291**

**MarkAsTaskRuleAction, 391**

**mdb databases, and Access DAO, 267–268**

**MDBView, 434**

**memory, 48–49**

**Menu Bar line, running macros from, 109**

**menus, in Explorers, 198–199**

**menus, in VB.NET, 199–206**

**Message Application Programming Interface. see MAPI (Message Application Programming Interface)**

**MessageClass property, forms**

defined, 112  
 form regions, 140  
 managing forms, 136–137  
 publishing forms with code, 129–132

**MessageClass property, items, 111**

**messages, Mail, 415–416**

**MFCMapi, 434**

**Microsoft Access, Outlook data in, 264–269**

ADO, 265–266  
 DAO, 267–269  
 DAO DBE, 268–269  
 overview of, 264–265

**Microsoft Development Network (MSDN), 427–428**

**Microsoft Excel, using Outlook data in, 257–264**

**Microsoft Exchange Server MAPI Editor, 435**

**Microsoft Internet Controls, 272**

**Microsoft Word, Outlook data in, 254–257**

**MoveOrCopyRuleAction, 391**

**MoveTo argument, 79**

**MoveToFolder action, Family rule, 93**

**MoveToStart method, tables, 30**

**MSDN (Microsoft Development Network), 427–428**

**multiple properties, accessing, 22–23**

**MyAxHost class, 335–336**

**N****Name property, contacts wrapper class, 87****named properties (UserProperties)**

- overview of, 8
- PropertyAccessor syntax, 14–15, 21–22
- publishing forms using, 129, 131
- testing code for forms, 134–135

**NameSpace object**

- accessing data stores, 9
- accessing Outlook item from data store, 7
- Categories collection, 54–55
- defined, 5
- Exchange properties, 54
- initializing globally trusted, 40–43
- new properties/methods, 413
- overview of, 53–54
- picking folders, 55
- picking names, 55–57
- retrieving CurrentProfileName, 45

**namespaces, DASL, 36****naming conventions, Outlook VBA project, 3–4****Navigate method, WebBrowser control, 270–271****NavigationFolder, 27, 392****NavigationFolders collection, 392****NavigationGroup, 27, 392****NavigationGroups collection, 393****NavigationModule, 393****NavigationModules collection, 393****NavigationPane, 27, 393–394****NewBar function, 199****NewComboBox function, 202–203****NewExplorer event**

- Connect class handling, 307
- Connect class in C#, 327–328
- Connect class in VB.NET, 316–317
- OnDisconnection Catch-22 bug and, 155–156
- opening folder in new window and, 152
- setting up Explorer event handler, 154–155

**NewInspector event**

- Connect class handling, 307
- Connect class in C#, 329–331
- Connect class in VB.NET, 317–318
- instantiating wrapper classes using, 81–82
- OnDisconnection Catch-22 bug, 156
- opening up Outlook items and, 152
- WordMail workarounds, 284

**NewItemAlertRuleAction, 394****NewMail event, 50–51, 282****NewMailEx event**

- Outlook 2007 limitations, 282
- overview of, 50–51
- working with, 72–74

**newsgroups, Outlook, 435–438****Nissen, Mads, 429****NoSubjectForm, VBA, 64–67, 84****NOT operator, AdvancedSearch method, 75–76****notation, hexadecimal, 17****NotesModule, 394****O****o1DefaultSelectNamesDisplayMode enumeration, 55–56****Object Browser, 4****Object declarations, and unmanaged code, 276****OCXs (ActiveX controls)**

- deploying unmanaged code addins, 279–280
- and form behavior, 138–139
- using in custom forms, 114, 119–120
- using in VBA UserForms, 64
- using with form regions, 245

**ODBC (Open DataBase Connectivity) drivers, 8****Office Developer Center Outlook, 428****Office development information, 428****Office document syntax, 15, 22****<officeMenu> tag, Ribbon, 219****Offline Storage (OST) files, 8****OFT (Outlook form template) files, 112, 129–132****OISStorageIdentifier enumeration, 25–26****On Error Resume Next statement, Family rule, 8****onAction callback, Ribbon, 224–225, 229–231****OnAction property, 201****OnAddInsUpdate event, 148****OnBeginShutdown event, 148****OnConnection event**

- communicating with COM addin, 39, 188
- Connect class in C#, 321–325
- Connect class in VB.NET, 308–310
- defined, 148

**OnDisconnection Catch-22 bug, 155–179**

- C# with VSTO, 164
- Explorer wrappers and, 164
- Explorer wrappers in C#, 167–171
- Explorer wrappers in VB.NET, 165–167
- Inspector wrappers and, 164
- Inspector wrappers in C#, 174–179
- Inspector wrappers in VB.NET, 171–174
- overview of, 155–157
- VB.NET with VSTO, 157–164

**OnDisconnection event**

- Catch 22 bug, 155
- Connect class in C#, 321–325
- Connect class in VB.NET, 308–310
- defined, 148

**one-offed forms**

- controlling Active X behavior, 138–139
- disadvantages of custom forms, 115
- trusted forms vs., 138

**online resources**

- CDO, 288–289
- DASL namespaces, 36

## online resources (*continued*)

### online resources (*continued*)

- deploying VSTO 2005 SE addins, 281
- downloading Ribbon tlb, 278
- finding user group, 436
- forums, 438
- MAPI mailing list, 438
- Microsoft, 427–432
- Microsoft Exchange Server MAPI Editor, 435
- Microsoft shim wizard, 282
- newsgroups, 435–438
- non-Microsoft, 432–433
- OnUpdate event example, 45
- Outlook object model guard, 34
- software development kits, 435
- support catalogs, 438
- task management, 305
- virtual machines, 436
- Windows registry, 137

**OnStartupComplete event, 148–149, 235–238**

**OnUpdate event, 45**

**OOM. see Outlook Object Model**

**Open DataBase Connectivity (ODBC) drivers, 8**

**OpenReport method, 265**

**operators, AdvancedSearch method, 75–76**

**Option property page, 186**

**OptionButton controls, NoSubjectForm, 65–66**

**OptionsPagesAdd event, 186–187**

**OR operator, AdvancedSearch method, 75–76**

**OrderField object, 394–395**

**OrderFields collection, 395**

**Organizational Forms Library, 129**

**OST (Offline Storage) files, 8**

**Out of Office state, Property tag, 17–18**

**OutExpl method, task management, 337–339**

**OutInsp method, task management, 344–347**

**Outlook, real-world programming, 275–303**

- addin deployment, 279–282
- coding for compatibility, 276–277
- Outlook 2003 and earlier, 283–287
- Outlook 2007, limitations of, 282–283, 287
- Ribbon considerations and workarounds, 277–278
- supporting multiple versions of Outlook, 275–276
- using CDO, 288–289
- using Redemption, 289–303

**Outlook 2003 and earlier, 283–287**

**Outlook 2007, improvements to**

- DASL, 35–38
- performance, 30–34
- security, 34–35
- unified object model. *see* Outlook Object Model

**Outlook 2007, overview of, 1–11**

- data display, 10–11
- data storage, 8–10
- folders, 6–7
- items, 8
- profiles and data stores, 6

- sessions and data stores, 6
- setting up Outlook VBA, 1–4
- understanding, 5

### Outlook data, using with other applications

- Microsoft Access, 264–269
- Microsoft Excel, 257–264
- Microsoft Word, 254–257
- Web Browser, 269–272

### Outlook form template (OFT) files, 112, 129–132

### Outlook Object Model, 377–425

- Account object, 378
- AccountRuleCondition, 378
- accounts, 28–29
- Accounts collection, 378
- AddressEntry object, 410
- AddressList object, 410
- AddressRuleCondition, 378–379
- Application object, 410
- AppointmentItem object, 410
- AssignToCategoryRuleAction, 379
- Attachment object, 411
- AttachmentSelection collection, 379
- AutoFormatRule, 379
- AutoFormatRules collection, 380
- BusinessCardView, 380
- CalendarModule, 381
- CalendarSharing, 381
- CalendarView, 381–382
- CardView, 383
- Categories collection, 383–384
- Category object, 384
- CategoryRuleCondition, 384
- Column object, 384–385
- ColumnFormat, 385
- Columns collection, 385
- ContactItem object, 411
- ContactsModule, 385–386
- DistListItem object, 411–412
- ExchangeDistributionList, 386
- ExchangeUser, 386–388
- Explorer object, 412
- Folder object, 412
- FormNameRuleCondition, 388
- FormRegion, 388–389
- FormRegionStartup, 389
- FromRssFeedRuleCondition, 389
- IconView, 389–390
- ImportanceRuleCondition, 390
- Inspector object, 412
- Item object, 412
- JournalModule, 390
- MailItem object, 413
- MailModule, 390–391
- MarkAsTaskRuleAction, 391
- MoveOrCopyRuleAction, 391
- NameSpace object, 413

- NavigationFolder, 392
  - NavigationFolders collection, 392
  - NavigationGroup, 392
  - NavigationGroups collection, 393
  - NavigationModule, 393
  - NavigationModules collection, 393
  - NavigationPane, 393–394
  - NewItemAlertRuleAction, 394
  - NotesModule, 394
  - OrderField, 394–395
  - OrderFields collection, 395
  - PlaySoundRuleAction, 395
  - PostItem object, 413–414
  - PropertyAccessor, 395–396. *see* PropertyAccessor
  - Recipient object, 414
  - Row, 396
  - Rule, 396
  - RuleAction, 396–397
  - RuleActions collection, 397
  - RuleCondition object, 398
  - RuleConditions collection, 398–399
  - Rules collection, 399
  - Search object, 414
  - SelectNamesDialog, 399–400
  - SenderInAddressListRuleCondition, 400
  - SharingItem, 400–401
  - StorageItem, 401
  - Store object, 401–402
  - Stores collection, 402
  - Table object, 402
  - TableView, 402–404
  - TaskItem object, 414
  - TasksModule, 404
  - TextRuleCondition, 404
  - TimelineView, 405
  - TimeZone, 406
  - TimeZones collection, 406
  - ToOrFromRuleCondition, 406–407
  - user interface objects, 26–28
  - UserDefinedProperties collection, 407
  - UserDefinedProperty, 407
  - View object, 414
  - ViewField, 407
  - ViewFields collection, 408
  - ViewFont, 408
  - working with StorageItems, 25–26
  - working with stores, 24–25
  - Outlook Spy, 434**
  - Outlook VBA, 59–110**
    - accessing items from folders, 7
    - class modules, 61–63
    - code modules, 63–64
    - code-signing certificates, 3–4
    - coding for compatibility, 276–277
    - distributing individual macros, 110
    - distributing projects, 109–110
    - distributing VBA project, 109–110
    - interface and options, 4
    - macro security, 2, 60–61
    - macros and VBA procedures, 60
    - Outlook Application events, 69–77
    - Outlook Folder events, 77–79
    - Outlook user events, 80
    - overview of, 1
    - procedures, 60
    - project file, 59
    - prototyping forms code in, 116–118, 122–124
    - running macros, 108–109
    - using code window, 121–122
    - working with ThisOutlookSession class, 60
    - working with UserForms, 64–68
    - wrapper classes and collections, 81–88
  - Outlook VBA, macro projects**
    - check for existing contact using Accounts collection, 100–102
    - check today's to-do list, 102–105
    - create context menu rule, 94–99
    - create custom rules, 89
    - create Family rule, 89–94
    - create importance search folder, 107
    - create incomplete tasks search folder, 105–106
    - create overdue reminders search folder, 107–108
    - create overdue tasks search folder, 106–107
    - create UserForms, 64–68
  - OutOfOffice state, workarounds for, 287**
  - OutOfOfficeText property, workarounds for, 287**
  - Overdue reminders search folder, 107–108**
  - Overdue Tasks search folder, 106–107**
- ## P
- parent property, AddChildTask method, 349–351**
  - ParentTag string, 319–321**
  - ParentTask property, 306**
  - passwords, forms, 127, 136**
  - PasteFace method, 205, 241**
  - performance, improvements in, 30–34**
  - Personal Forms Library**
    - opening previously published forms in, 113
    - publishing and running form in, 128
    - publishing forms to, 129, 130
    - and Redemption, 301–303
  - Personal Storage (PST) files, 8, 24**
  - PickFolder dialog, 55**
  - Plain Text toggleButton, 225–226**
  - PlaySoundRuleAction, 395**
  - pop-up controls. *see* CommandBarPopup control**
  - Post form icons, changing, 127**
  - PostItem object, 413–414**
  - Private property, Redemption, 289**
  - profiles, Outlook, 5–6**

## programming Outlook

**programming Outlook.** *see* **Outlook, real-world programming**

### properties

- accessing. *see* **PropertyAccessor**
- accessing multiple, 22–23
- binding controls to, 118–119
- Browser Control Form, 269
- creating form region, 141–144
- DASL definitions of, 20–21
- form control, 126–127
- form region, 140–141
- item, 8
- JET syntax using standard names of, 37
- Outlook and named MAPI, 14
- overview of, 14
- PropertyAccessor** limitations, 283
- store, 24–25

**Property ID syntax, PropertyAccessor, 15, 18–21**

**property pages, displaying, 185–187**

### Property tags

- checking today's to-do-list, 103
- creating with Property ID syntax, 18–21
- Importance search folder, 107
- Incomplete Tasks search folder, 105
- Overdue reminders search folder, 108
- Overdue Tasks search folder, 106
- in Redemption code, 298–301
- syntax, 16–18

### PropertyAccessor, 14–24

- accessing multiple properties, 22–23
- binary/string methods, 24
- date/time methods, 24
- limitation of, 17
- methods/properties/events, 395–396
- Outlook 2007 limitations, 287–288
- overview of, 14–15
- Property ID syntax, 18–21
- Property tag syntax, 16–18
- syntax 3, 21–22
- syntax 4, 22
- and untrusted code, 35

**PropertyChange event handler, 86–88, 134–135**

**PST (Personal Storage) files, 8, 24**

**Public folders, running script in, 139**

### publishing forms

- choosing location, 129
- with code, 129–132
- customized, 115, 128
- standard, 113
- and trusted forms, 138

## Q

**Quit event, 52**

## R

**RDO (Redemption Data Objects) class, 289**

**RDOAddressEntry object, 290–291, 294**

**RDOItems collection, 289**

**RDOSession object, 290**

**RDOStores collection, 292, 294–295**

**ReadReceipt property, workarounds for, 287**

**real-world programming.** *see* **Outlook, real-world programming**

**receive rule, 89–94, 98–99**

**Recipient object, 101, 290**

**Recipients collection, 27, 101**

### Redemption

- as alternate API, 289–303
- avoiding **PropertyAccessor** limitations with, 17
- Outlook 2007 workarounds using, 284

**Redemption Data Objects (RDO) class, 289**

**reflection, 276–277**

### registry

- deploying unmanaged code addins using, 279
- form region settings, 140–141, 144
- implementing form regions with COM addins, 240
- managing forms in, 137

**Reminder event, 51**

**Reminders collection, 51**

**RequestService, override, 213–215**

**Restrict filter, 255**

**Restrict method, MAPITable filter, 297**

**Results collection, AdvancedSearch, 74, 76–77**

**Ribbon, 217–234.** *see also* **Developer tab, Ribbon**

- blog, 429
- CommandBar vs., 217
- customization rules, 219–220
- for multiple versions of Outlook, 277–278
- programming with COM addin, 97
- shims supporting, 282
- task management user interface, 305–306
- VB.NET code for, 220–223
- VSTO code for, 234
- VSTO interface handlers using VB.NET, 213–214
- XML schema, 218–220

### Ribbon callbacks

- Connect class implementing, 307
- Connect class in C#, 331–335
- Connect class in VB.NET, 311–314
- for multiple versions of Outlook, 278
- overview of, 224–229
- using in C#, 230–234

### Ribbon XML

- Connect class in VB.NET, 311
- task management in C#, 373
- task management in VB.NET, 372–373

**Ribbon\_OnLoad callback handler, 229**

**Ribbon\_OnLoad event**

- defined, 220
- firing of, 228
- Ribbon callbacks in C#, 234, 332–335
- Ribbon callbacks in VB.NET, 312–313

**Rich Text, custom forms, 115****Rich Text toggleButton, 225****Row, 396****Rule, 396****RuleAction, 396–397****RuleActions collection, 397****RuleConditions, 92–93, 398****RuleConditions collection, 398–399****Rules collection, 89–99**

- context menu rules, 94–99
- creating custom rules, 89
- executing rules on demand, 93
- Family rule, 89–94
- methods/properties/events, 399
- overview of, 91–92

**Rules Wizard, 89, 92–94****Run button, macros, 108–109****Run icon, macros, 109****RunAutoMacros method, 257****S****schema, Property ID syntax, 18–21****SchemaName syntax, 15****scripts, form, 139****SDKs (software development kits), 435****search folders**

- creating from completed AdvancedSearch, 106
- Importance, 107
- Incomplete Tasks, 105–106
- Overdue reminders, 107–108
- Overdue Tasks, 106–107

**Search object, 414. see also AdvancedSearch object****security**

- CDO vs. Outlook object model, 288
- certificates. see code-signing certificates
- improvements in, 34–35
- macro, 2
- for managed code addins, 280–281
- using Redemption for Outlook, 289

**Selection collection, 98, 101****SelectionChange event**

- creating CommandBar user interface, 210
- DoStartup procedure in Explorer, 203–205
- ExplWrap class in C#, 343–344
- ExplWrap class in VB.NET, 340–341

**SelectNamesDialog, 399–400****SelectNamesDialog, 27–28****Selfcert.exe, 3–4****self-help, Outlook, 438–439****Send event**

- email wrapper class, 84
- Inspector wrapper classes and, 165
- Outlook 2007 limitations, 282

**send rule, creating custom rule as, 89****SendAndReceive method, Namespace, 54****SenderAddress property, context menu rule, 99****SenderInAddressListRuleCondition, 400****SendUsingAccount**

- ItemSend event, 70–71
- Outlook 2007 limitations, 282
- syntax for object properties, 29

**Sensitivity property, contacts wrapper class, 86–88****Session object, 7, 289–290****Set keyword, 29, 93****SetDefaultDisplayMode method,****GetSelectNamesDialog, 55–56****SetProperties method, PropertyAccessor, 22–23****SetupButtonsForThisInspector method, 285–286****shared folders, 139****shared managed code addins, 281–282****SharingItem, 400–401****shims, 282****ShortcutContextMenuDisplay event, 46****Show Related Tasks control, 305–306****ShutdownAdmin procedure, VSTO, 43****Signed property, workarounds for, 287****SMTPAddress property, RDOAddressEntry, 290–291****software development kits (SDKs), 435****SortedList objects**

- Connect class in C#, 319–321
- OutExpl in C#, 338–339
- Teardown method, 325

**spreadsheets, Outlook data in, 257–264****SQL (Structured Query Language), Outlook data in, 264–269****standard forms**

- customization, 113–115
- integrating custom forms with, 114
- libraries, 112
- MessageClass property, 111–112
- publishing, 113

**Standard Forms Library, 112****Startup event handler**

- defined, 52
- Outlook user events, 80–88
- working with, 69–70

**Startup method, VSTO, 44****StorageItem, 25–26, 401****Store object**

- accessing using Namespace, 9
- AdvancedSearch method for, 74–77
- defined, 5
- methods/properties/events, 401–402
- Outlook data organized into, 8

## Store object *(continued)*

### Store object *(continued)*

Outlook profiles/sessions and, 6  
 Rules collections for, 91  
 table limitations and, 33  
 at top level of Outlook storage, 24  
 working with, 24–25

### StoreContextMenuDisplay event, 46

### StoreKind property, RDOSTores collection, 294–295

### Stores collection, 402

### string properties, PropertyAccessor, 282–283

### StringToBinary method, PropertyAccessor, 24

### strong references, 44

### Sub procedures, 60

### subfolders, 6

### Super Tip, viewing text of, 228

### support, development

forums, 438  
 MAPI mailing list, 438  
 newsgroups, 435–438  
 overview of, 435  
 self-help, 438–439  
 support catalogs, 438

### support catalogs, Outlook, 438

### Switch block, 331

### SyncObjects collection, 54

## T

### Table object

accessing data with, 9–10  
 checking today's to-do-list, 104  
 default columns, 33  
 filtering, 31–32  
 limitations of, 33–34  
 methods/properties/events, 402  
 performance improvement using, 30–31, 287  
 StorageItems and, 25–26  
 unavailable in untrusted code, 35  
 using Outlook data in Word, 254–257

### TableView, 248, 402–404

### tabs

creating with Ribbon XML, 218  
 Ribbon customization rules, 220

### Tag properties

AdvancedSearch using, 74, 76–77  
 creating for  
     CommandBarPopup/CommandBarButton, 204  
 handling collections with, 47

### target folder, Family rule, 91

### Task Management System, 305–373

C# ExplWrap class, 341–344  
 C# InspWrap, 353–362  
 C# MyAxHost class, 335–336  
 C# OutExpl, 338–339  
 C# OutInsp, 345–347  
 C# Ribbon XML, 373

C# TaskPane, 366–371

features, 305–306

VB ExplWrap class, 340–341

VB Globals, 337

VB InspWrap, 347–353

VB OutExpl, 337–338

VB OutInsp, 344–345

VB Ribbon XML, 372–373

VB TaskPane, 362–366

### Task Management System, Connect class in C#, 319–335

CTPFactoryAvailable event, 335

declarations, objects and variables, 319–321

IDTExtensibility2 events, 321–325

InitHandler method, 325–327

NewExplorer/Explorer.Close events, 327–328

NewInspector/Inspector.Close events, 329–331

Ribbon callbacks, 331–335

Teardown method, 325–327

### Task Management System, Connect class in VB.NET, 307–319

converting image file with AxHost, 318–319

CTPFactoryAvailable event, 310

declaring GUID and ProgID, 307–308

IDTExtensibility2 events, 308–310

InitHandler method, 314–316

NewExplorer/Explorer.Close events, 316–317

NewInspector/Inspector.Close events, 317–318

Ribbon callbacks, 311–314

Ribbon XML, 311

Teardown method, 314–316

### task panes, custom, 235–236

Connect class implementing, 307

creating in C#, 237–238

creating in VB.NET, 235–237

overview of, 234–235

working with, 239–240

### TaskItem object, 414

### TaskItem property, Inspector wrapper class, 344–345

### TaskPane class, 362–371

### Tasks folder, 105–107

### TasksModule, 404

### Teardown method

in C#, 183–185

Connect class in C#, 325–327

Connect class in VB.NET, 314–316

OnDisconnection Catch-22 bug, 156

in VB.NET, 180–182

### testing

addin communications, 189

code for forms, 132–135

code in Outlook VBA, 1–4

### TextBox control, 66, 125–127

### TextRuleCondition, 93, 404

### themed controls, form regions, 139

### themes, custom form pages, 128

### third-party libraries, untrusted code, 34–35

**ThisOutlookSession class, 60–62, 68–69. *see also*  
Outlook VBA**

**time**

- converting in Redemption, 291
- Outlook 2007 limitations, 282
- PropertyAccessor properties, 24
- table limitations and, 34

**TimelineView, 248–250, 405**

**TimeTravelFormRegionWrapper class, 245**

**TimeZone, 406**

**TimeZones collection, 406**

**tlb (type library), Ribbon, 278**

**TNEF (Transport Neutral Encapsulation Format), 115**

**Today's ToDo List macro, 67–68**

**ToDoForm, 67–68, 102–105**

**toggleButton controls**

- getPressed callback for, 227
- onAction callback for, 225
- overview of, 218–219
- Ribbon callbacks in C# for, 229–234

**toolbars**

- in Explorer, 198–199
- running macros from Outlook, 109
- in VB.NET, 199–206

**tools**

- MAPI viewing, 434–435
- software development kits, 435
- virtual machines, 435–436

**ToOrFromRuleCondition, 406–407**

**Transport Neutral Encapsulation Format (TNEF), 115**

**treeview control, in task pane, 362**

**troubleshooting and support resources**

- forums, 438
- MAPI mailing list, 438
- MAPI viewing tools, 434–435
- Microsoft resources, 427–432
- newsgroups, 436–438
- non-Microsoft Web sites, 432–433
- self-help, 438
- software development kits, 435
- support catalogs, 438
- virtual machines, 435–436

**trust, and code-signing certificate, 4**

**trusted code, 34, 35**

**trusted COM addins, 148–151**

**trusted forms, 138**

**Trusted Publishers list, 61**

## U

**unified object model. *see* Outlook Object Model**

**Unload event**

- contacts wrapper class, 86
- email wrapper class, 84
- overview of, 48–49
- using code window, 122

**unmanaged code**

- coding for compatibility, 276–277
- deploying addins, 279–280
- managed vs., 147–148

**untrusted code, 34, 35**

**user classes, 62–63, 69**

**user events, 80–88**

**user interface**

- COM addins and. *see* COM addins, and Outlook user interface
- task management, 305–306
- WordMail, 284–285

**UserDefinedProperties collection, 407**

**UserDefinedProperty, 407**

**UserForms, VBA, 64–68**

- NoSubjectForm, 64–67
- overview of, 64
- ToDoForm, 67–68

**UserProperties (named properties)**

- overview of, 8
- PropertyAccessor syntax, 14–15
- publishing forms using, 129, 131
- testing code for forms, 134–135

**UTC values**

- converting time in Redemption, 291
- Outlook 2007 limitations, 282
- PropertyAccessor date/time methods, 24

**UTCToLocalTime method, PropertyAccessor**

- defined, 24
- table limitations and, 34

## V

**variables**

- VBA editor options, 4
- VBScript vs. VBA form code, 124

**VB\_Action callback, 311–314**

**VB projects. *see* Outlook VBA**

**VB.NET, 198**

- Application object and, 40–41
- COM addin examples in this book using, 147
- communicating with COM addin, 188–189
- communicating with COM addin in VSTO, 191–193
- custom form regions, creating, 242–243
- custom form regions, working with, 245–246
- custom task panes, 235–237
- displaying property pages, 185–186
- Explorer wrappers, 165–167
- implementing IExtensibility2, 149–150
- implementing IRibbonExtensibility, 220–222
- initializing trusted Application/NameSpace objects, 40–43
- Inspector wrappers, 171–174
- OnDisconnection Catch-22 bug in VSTO with, 157–164
- setting up Inspector/Explorer event handlers, 152–155
- unmanaged code in, 148

## VB.NET (continued)

### VB.NET (continued)

using AxHost, 216  
 using ItemSend, 52  
 utility code, 180–182  
 VSTO interface handlers, 213–214

### VB.NET, Task Management

Connect class, 307–319  
 ExpWrap class, 340–341  
 Globals, 337  
 InspWrap, 347–353  
 OutExpl, 337–338  
 OutInsp, 344–345  
 Ribbon XML, 372–373  
 TaskPane, 362–366

### VBScript

storing in code window, 120–122  
 using VBA form code as, 124  
 writing form code only in, 114

### Version property

form region settings, 143  
 form settings, 127  
 overview of, 53  
 publishing custom forms, 115

### view designer, view XML in, 250–251

### View object

creating custom timeline for Inbox, 249  
 new properties/methods, 414  
 types of custom views, 248  
 viewing XML, 250

### ViewContextMenuDisplay event, 46

### ViewField, 250, 407

### ViewFields collection, 250, 408

### ViewFont, 408

### views

creating custom, 248–249  
 filtering and formatting custom, 249–250  
 improvements to, 26–28  
 XML, 250–251

### Views collection, 248–249

### Virtual PCs, 436

### VisibleStateChange event, InspWrap, 353, 361–362

### VMWare, 436

### VSTO template code

2205 vs. 2005SE versions, 44  
 Application shutdown code, 43–44  
 Application startup code, 41–43  
 blogs for, 429  
 COM addin examples in this book using, 147  
 communicating with COM addin in C#, 194–196  
 communicating with COM addin in VB.NET, 191–193  
 creating custom form regions, 243–244  
 deploying 2005 SE addins, 281  
 Explorer wrappers in C#, 167–171  
 Explorer wrappers in VB.NET, 165–167  
 IDExtensibility2 in C#, 150–151  
 IDExtensibility2 in VB.NET, 150

InitHandler in C#, 183

initHandler in VB.NET, 180

Inspector wrappers in C#, 174–179

Inspector wrappers in VB.NET, 171–174

interface handlers, 213–215

limitations of, 148

OnDisconnection Catch-22 bug in VB.NET, 157–164

startup event, 148–149

version used in this book, 39

working with Ribbon in, 234

## W

### weak references, 44

### Web Browser, Outlook data in, 269–272

### Web sites

online resources. *see* online resources

troubleshooting and support. *see* troubleshooting and support resources

### WebBrowser control, 269–271

### WEBDav (Web Distributed Authoring and Versioning), 35

### Whitechapel, Andrew, 430

### WindowActivate event handler, 285

### Windows Platform SDK, 435

### Wise installer, 279

### WithEvents, Connect class, 307

### Word, Outlook data in, 254–257

### WordMail, Outlook 2003 limitations, 283–287

### Workbooks collection, using Outlook data in, 259–264

### Worksheets, using Outlook data in, 259–264

### wrapper classes, 81–88

Close events, 152

contacts, 85–88

email, 82–85

Explorer, in C#, 167–171

Explorer, in VB.NET, 165–167

Explorer and Inspector, 164–165

importance of, 164–165

overview of, 81

using NewInspector event to instantiate, 81–82

### wrapper collection

Connect class in VB.NET, 317

email wrapper, 82–85

Ribbon callbacks, 224

wrapper classes, 81

## X

### X-header, creating for email item, 293

### XL-Dennis, 278

### XML

for custom views, 248, 250–251

implementing form regions with COM addins, 241

Ribbon, overview of, 218–220

Ribbon in C#, 373

Ribbon in VB.NET, 372–373