

Index

■ Numbers & Symbols

-= syntax, 202

3D cube creation process, 49

■ A

Abort method, 279–280

action behaviors, 67

Active Record (AR) pattern, 115–117

Add New Test dialog, 215

Add Service Reference window, 302

AddNew Test wizard, 216

AddObject method, 143

Address entity, 130–131

address, of endpoint, 292

Aggregate function, 186

Agile development approach, 209

animations, 51–53, 111

App class, 78

App.xaml class, 62

App.xaml file, 77, 79, 83

App.xaml Resource element, 90

App_Start method, 84

Application Files button, 256

application layer, 31

application manifest files, 255

Application Updates dialog, 255

Application.Activated method, 79

Application.Deactivated method, 79

Application.Run method, 78

Application.ShutDown method, 79

ApplicationCommands class, 192–195

applications

- configuration, 35–41

- deployment, 251–260

 - publishing options, 254–256

 - requirements for, 251–252

 - security, 252–254

 - TimeTracker application, 257–260

- developing

 - database schema, 34

 - requirements, 25–26

 - using third-party resources, 38–41

- layered, 27–30, 219–224

- loosely coupled, 209

- n-tier, 28–30
- updates, 254–255
- appointment calendar control, 108–109
- Appointment entity, 134–135
- Apress TimeTracker. *See* TimeTracker application
- AR objects, 115–116
- architecture
 - WCF, 291–292
 - WPF, 13–17
- assemblies, 13–17
 - technologies, 14
 - data templates, 15–16
 - DataBinding, 17
 - styles, 14–15
- Assembly properties window, 257
- Assert class, 140, 214
- Assert.Inconclusive method, 220
- Asset panel, 49, 64
- association object, 129
- associations, between entities, 129–134
- asynchronous code
 - BackgroundWorker component, 285–287
 - Thread class, 279–280
 - Thread object, 276–279
 - ThreadPool component, 283
- asynchronous operations, 280–282

- asynchronous pattern, 281–283
- attached properties, 8
- attribute syntax, 4
- authentication, 309–311
- Authenticode, 252
- authorization, 309–311
- AutoGenerateColumns property, 106
- autogenerated code, 114, 123–127
- AutoMapper, 26

■ B

- background processes, 280
- Background property, 4
- BackgroundWorker component, 285–287
- BaseDetailsViewModel class, 73–74, 76, 184
- BaseListViewModel class, 74–76, 184
- BaseViewModel class, 72–73, 76
- BeginInvoke method, 281
- behaviors, 44, 51, 67–69
- bidirectional data binding, 159, 162
- binding. *See* data binding
- Binding class, 160
- binding source, 159–160
- binding target, 159–160
- bindings, 309
- bitmaps, 48
- BizTalk, 290

blendability feature, 57
 Border XAML element, 64
 box model technique, 11
 BrushConverter, 4
 business layer, 28, 30, 35, 117, 220
 business logic, 30, 113, 117–121, 261
 Button control, 4, 12
 ButtonClick event, 282, 286

■ C

C# code, 276
 CancelAsync method, 287
 CanExecute event, 193
 CanExecute method, 191–196, 200
 CanExecuteChanged event, 191, 197
 CanExecuteRoutedEventArgs, 194
 Canvas, 10
 Cascading Style Sheets (CSS), 14, 91
 Certificate Authority server, 253
 certificates, 252–253, 258
 ChannelService class, 292
 charts, 107–108
 CheckBoxColumn, 106
 child controls, 9
 child ViewModels, 185–186
 class library, 62
 classes
 binding, to data template, 166–168

 splitting definition of, 127
 Cleanup method, 150
 Click event, 68
 ClickOnce deployment, 251–260
 TimeTracker application, 257–260
 publishing options, 254–256
 file dependencies, 255–256
 updates, 254–255
 security, 252–254
 certificates, 252–253
 Code Access Security, 253–254
 client tier, 30
 closing bracket (>), 3
 CLR, 6–7, 13–14
 code
 C#, 276
 asynchronous, 276–287
 coloring, 219
 loosely coupled, 28
 sequential execution of, 276
 Code Access Security, 253–254
 Code Coverage tool, 210–212, 216–219
 Code First development, 114, 117
 code-behind file, 6–7, 20, 43
 coded UI test, 216
 CodePlex, 9
 code-signing certificates, 252
 collection property, 70

- ColumnDefinition property, 102
- ComboBox, 12
- ComboBoxColumn, 106
- Command pattern, 190–191
- Command property, 204
- CommandBinding, 192, 195, 201
- CommandManager, 200–201
- CommandManager.InvalidateRequerySuggested method, 200
- commands
 - CRUD, 115, 120
 - dynamically re-evaluating execution of, 201–204
 - ICommand interface, 195–198
 - initializing, 205
 - interactive, 198–201
 - introduction to, 190–195
 - listening for changes, 200
 - RoutedCommand class, 191–195
 - TimeTracker, 204–207
- complex types, 128
- ComplexProperty property, 128
- components, redistributing, 289
- composite key, 34
- concurrency, 115, 283
- ConfigureContainer method, 84
- Contact entity, 132
- Container property, 100
- Content property, 5–6, 95
- ContentPresenter WPF control, 243
- ContinueWith method, 285
- ContractDescription class, 292
- controls
 - adding to views, 87–112
 - creating styles for, 55–57
 - customizing, 15, 55–57, 110–111
 - customizing layout of, 166–168
 - data binding to, 159–187
 - default, 12
 - defining data template for, 166–168
 - document, 12
 - styles, 91–97
- Controls folder, 95
- ControlTemplate, 12, 168
- Convert method, 163–164
- ConvertBack method, 163–164
- core code, 13
- COUNT function, 236
- Create Dataset dialog, 238
- Create, Read, Update, and Delete (CRUD) commands, 115, 120
- CreateEntityKey method, 148
- CreateObjectSet<T> method, 145–146
- credentials, 310
- Crystal Reports (CR), 227

custom controls, creating, 15, 55–57, 110–111

custom data source, creating, 57–58

custom effects, 110–111

custom fonts, 51

custom references, 3

Customer entity, 133–134

■ D

dashboard

layout, 109–110

user controls for, 107–108

Data Access Layer (DAL), 114

data binding, 159–187

bidirectional, 159, 162

binding expressions, 162–165

DataTemplate, 166–168

direction, 162

endpoints, 292

INotifyPropertyChanged interface, 160–161

introduction to, 159

IValueConverter interface, 162–165

of Model to ViewModel, 271

OneTime, 162

OneWay, 162

OneWayToSource, 162

source, 159–160

syntax, 170–171

target, 159–160

UI validation, 168–169

unidirectional, 159

UpdateSourceTrigger property, 162

ViewModels, 171–187

in WPF, 159–171

[DataContract] attribute, 297

data contracts, 290

Data Definition Language (DDL) scripts, 114

data integrity, 309

data layer, 28, 30

creating, 135–158

Unit of Work implementation, 135–150

repository pattern, 151–158

testing, 221–223

transaction management, 140–143

Data Manipulation Language (DML), 114

[DataMember] attribute, 297

data service object, 305

data templates, 166–168

data validation, 103–104, 168–169

DataAnnotations, 172, 177

database

creating new, 133

independence, 114

logic, 119

- objects, 34
- unit test, 216
- Database First development, 114
- database schema, 34
 - generating, 133
 - O/RM and, 114
- database tier, 29
- DataBinding, 17
- DataContext, 57, 70, 102, 119
- DataContracts, 296–297
- DataErrorValidationRule component, 169
- DataGrid control, 67–69, 104, 106
- dataset, creating, for reports, 233–236
- datasource
 - creating, for reports, 232–233
 - custom, 57
- DataTemplates, 12, 15–16
- DateConv, 165
- debugging, 209
- default controls, 12
- default properties, 6
- delegate methods, 194
- DeleteCommand, 207
- deleting
 - objects, 145–146
 - roles, 153–154
- dependency injection (DI), 79–81. *See also* inversion of control

- Dependency Injection pattern, 152
- dependency properties, 14, 96
- deployment, 251–260
 - publishing options, 254–256
 - requirements for, 251–252
 - security, 252–254
 - TimeTracker application, 257–260
- description, of endpoint, 292
- design patterns, 261–273
 - Active Record (AR), 115–117
 - advantages of using, 261
 - applying to graphical layer, 261
 - asynchronous, 281–283
 - Command, 190–191
 - Dependency Injection, 152
 - Model View Controller (MVC), 262
 - Model View Presenter (MVP), 191, 262–268
 - Model View ViewModel (MVVM), 22, 26, 171, 191, 262, 268–273
 - Passive View MVP, 263–266
 - Presentation Model (PM), 268
 - Repository, 117–119, 121, 151–158
 - Supervising Controller, 266–268
 - for UI, 261–273
 - Unit of Work (UoW), 119–121
- design-time data, 57–58
- detached entity, updating, 149–150

Details view, 69–70, 73–74, 100–104
 DialogService, 82
 dictionaries

- accessing, 91
- creating, 88
- embedding images in, 89–91

 Dispatcher class, 279
 Dispatcher object, 279
 DisplayMemberPath, 15
 Dispose method, 136
 dockable panels, 20
 DockPanel control, 8, 10, 12
 document controls, 12
 DocumentPageView, 50
 domain, 113
 Domain Driven Design (DDD), 33, 113

- Code First development and, 117
- Repository pattern, 117–119
- Unit of Work (UoW) pattern, 119–121

 domain entities, 113

- Address entity, 130–131
- Appointment entity, 134–135
- Contact entity, 132
- creating new, 122–123
- Customer entity, 133–134
- customizing, 127
- Employee domain entity, 127–128
- marking status of, 143

- relationships between, 128–134
- updating detached, 149–150

 domain layer, 30
 domain model, 32–34

- behaviors and relationships in, 128–130
- creating new entities in, 122–123
- creating, with Entity Framework, 122–135
- mapping, 122
- serializing, 304

 domain-first development, 34
 domain-specific language (DSL), 113
 dot syntax, 4
 DoubleAnimation element, 111
 DoWork event, 286–287
 drawing, in Expression Blend, 48–51
 DropShadow effect, 64
 dummy data, 65–66, 70
 dynamic resources, 17, 88

■ E

EditCommand, 206
 EF Designer tool, 123, 127
 EF Framework, 127
 EF proxy classes, 143
 embedded styles, 14
 Employee domain entity, 127–130
 EmployeeDetailsReport, 237–240

- EmployeeDetailsView, 246–248
- EmployeeDetailsViewModel, 175–180
- EmployeeListReport, 231–233
- EmployeeListView, 204–207, 244–246
- EmployeeListViewModel, 180–184, 205, 244
- EmployeeRepository class, 182
- EndInvoke method, 281–283
- endpoints, 289–292, 309
 - address, 292
 - binding, 292
 - description, 292
 - testing, 302–303
- Enterprise Library 5.0, 26
- entities. *See* domain entities
- Entity Framework, 113–58
 - Active Record pattern, 115–16
 - adding objects, 143–145
 - Code First development, 117
 - creating domain model with, 122–135
 - data layer, 35, 135–158
 - deleting objects, 145–146
 - Designer file, 122
 - extension methods, 147–148
 - modifying objects, 146–150
 - object relational mappers, 113–115
 - Repository pattern, 117–119, 151–158
 - TimeTracker domain, 121–135
 - unit of work, 119–121, 135–150
- EntityFrameworkExtensions class, 147
- EntityKey class, 157
- EntityObject class, 127, 147
- EntitySet property, 125
- EntityState property, 143
- error templates, custom, 169
- event handlers, 189–190, 202
- event listeners
 - attaching to ICommand, 201–204
 - controlling, 202
- event model, 202
- event triggers, 93, 110–111
- events
 - in XAML, 7
 - predefined, 189
 - routing, 189–190
 - triggering behaviors with, 68
- Events tab, Properties panel, 189–190
- .exe file, 9
- Execute method, 191–193, 196
- ExecuteCommand method, 191
- Executed event, 193
- ExecutedRoutedEventArgs, 194
- Expression Blend. *See* Microsoft Expression Blend
- Expression Gallery web site, 67

Extensible Application Markup Language.
See XAML

extension methods, 147–148

Extreme Programming (XP), 209

■ F

file dependencies, 255–256

files, importing, 44

Fixtures, 220

fonts, custom, 51

foreign keys, 129

format text, 51

FormatCommand property, 198

framework element, 10

full behaviors, 67

■ G

generic test, 216

GenericWeakEventManager, 205

GetAll method, 155–156

GetById method, 156–157

graphical layer, applying design pattern to,
 261

Grid container, 100, 102

Grid control, 11, 107

GridView, 12

Group By function, 236, 240–241

■ H

Header property, 100

Help tab, Expression Blend, 45

HorizontalAlignment property, 11

HyperlinkColumn, 106

■ I

IAsyncResult interface, 281

ICommand interface, 175, 177, 191–200,
 272–273

attaching event listeners, 201–204

interaction with
 INotifyPropertyChanged interface,
 201

ICommand objects, 180, 204–207

icons, 26

Id property, 123, 127, 149

IDataErrorInfo interface, 103, 169, 172–173

IDialogService interface, 80–81, 182

IDisposable interface, 136–137

image button, 94–96

ImageButton controls, 104

images, embedding images in, 89–91

INavigationService interface, 80, 100, 182

in-browser applications, 9

Inconclusive method, 214

Initialize method, 150

InitializeCommands method, 205, 245

InitializeService method, 305

Inlines property, 6

INotifyPropertyChanged interface, 70, 72, 160–161, 171, 201, 203, 269–272

input controls, 12

Insert<T> method, 152

Int32 property, 127

IntelliSense feature, 44

interactive commands, 198–201

internal scheduler, 276

Internet Information Services (IIS), 259–260, 290

inversion of control (IoC), 79–84, 103, 110

Invoke method, 279

Invoker, 191

IPersonService.cs interface file, 297

IRepository interface, 151–152

IsAttachedTo method, 148

IsEnabled property, 93

IsInTransaction property, 143

IsReference attribute, 297

ItemsControl, 12, 15

ItemTemplate, 168

IUnitOfWork interface, 135–136, 140

IUnityContainer interface, 180, 182

IValueConverter interface, 162–165

IWeakEventListener interface, 202

■ J

Java Forms, 14

■ K

KAXAML, 23

key, 34

keyframes, 51

■ L

Label control, 50

lambda expression, 284

Language-Integrated Query (LINQ), 146

layered applications, 27–30, 219–224

layers, 35–38

layout containers, 10

layout controls, 12

libraries, third-party, 26

line-of-business (LOB) applications, 17

list controls, 12, 15

List view model, 74–75

list views, 66–67, 104–106

Listbox control, 1, 12, 15–16

- customizing layout of, 166
- defining data template for, 166–168

ListViewModel, 184–185

load test, 216

logical code, separation of, in tiers, 28

Login view, 98–100

loosely coupled applications, 209

loosely coupled code, 28

LostFocus mechanism, 162

■ **M**

- MageUI.exe utility, 255
- main screen, prototyping, 64–65
- MainView, 109–110
- Makecert.exe utility, 252
- Managed Extensibility Framework (MEF), 45
- manifest files, 255
- many-to-many relationships, 128–132
- Margin property, 11
- MarkDeleted method, 120, 145–146, 153
- MarkDirty method, 120, 146–150
- MarkNew method, 120, 143–145
- MaxHeight, 100
- media integration layer (milcore), 13
- messages, 289–290
- metadata, 290
- MethodNameCompleted event, 287
- Microsoft Expression Blend, 1, 8, 21–22, 43–60
 - animations, 51–53
 - behaviors, 67–69
 - creating mock-up and prototyping, 62–70
 - design-time data, 57–58
 - drawing, 48–51
 - shapes and objects, 49–50
 - text and text effects, 50–51
 - dummy data in, 65
 - hardware requirements, 43–44
 - mockups, 59–60
 - new features, 44
 - New Project window, 46
 - overview, 43–44
 - SketchFlow, 59–60
 - startup page, 45
 - styling and template controls, 53–57
 - workspace, 45–48
 - panels, 46–48
 - resizeable regions, 47
- Microsoft Expression Studio, Ultimate version, 43
- Microsoft Reporting Services. *See* SQL Server Reporting Services; reports
- Microsoft Unity, 80
- middle tier, 28
- milcore, 13
- mockups, 21, 59–60, 62–70
- Model
 - binding to ViewModel, 271
 - in MVVM pattern, 269–271
- Model Browser panel, 128
- Model View Controller (MVC) pattern, 262
- Model View Presenter (MVP) pattern, 191, 262
 - Passive View, 263–268
 - Supervising Controller, 266–267
 - when to use, 268

Model View ViewModel (MVVM) pattern,
 22, 26, 171, 191, 195, 261–262, 268–
 273

 exposing the commands, 272–273

 INotifyPropertyChanged interface, 269–
 271

modifying objects, 146–150

MouseDownElementBehavior, 67

MSDN Ultimate subscription, 43

MSTest, 209, 212–216

multi-core processors (CPUs), 275

multithreading, 275–288

 asynchronous operations, 280–282

 code, in .NET framework, 276–287

 BackgroundWorker component,
 285–287

 Task object, 283–285

 Thread class, 279–280

 Thread object, 276–279

 ThreadPool object, 282–283

 definition of, 275

 when to use, 276

MVVM toolkits, 26

MvvmCommand, 196–200

 enabling listening, 203

 implementaion with , 205

 refactored, 203

■ N

Name property, 149

namespaces, 2–3

naming conventions, 38

navigation, between views, 77–79

NavigationService class, 80, 180, 244

.NET Framework 4, 8

 installation and distribution, 251

 multithreading code in, 276–287

 parallel processes in, 275

NHibernate, 114

n-tier architecture, 28–30

NullReferenceException error, 161

■ O

object data sources, 57

object persistence, 113–115

object relational mappers (O/RM), 113–
 116, 123

ObjectContext class, 123, 136–137, 140

objects

 adding new, 143–145

 AR, 115–116

 creating data template for, 168

 deleting, 145–146

 drawing, 49–50

 in XAML, 3

 modifying, 54–55, 146–150

- serialization of, 297
 - validating, 168–169
- ObjectSet<T> method, 155
- ObservableCollection<T> method, 76
- ObservableObject class, 70
- ObservableObject<T> method, 198
- oData protocol, 304–307
 - oData queries, 306–307
 - oData service, 306
 - oData syntax, 306–307
- OnCanExecuteChanged method, 197
- OneTime data binding, 162
- one-to-many relationships, 129
- one-to-one relationships, 129
- OneWay data binding, 162
- OneWayToSource data binding, 162
- OnLoad event, 111
- OnPropertyChanged event, 146, 161
- OOP principles, 27
- opening bracket (<), 3
- [OperationContract] attribute, 298
- ordered test, 216
- OriginalText property, 201

■ P

- Padding property, 11
- Page, 2
- parallel processes, 275–276, 281. *See also* multithreading
- parameters
 - array of, 247
 - creating reports with, 238–240
- partial keyword, 127
- Passive View MVP pattern, 263–266
- Passive View pattern, 268
- persistence layer, 117
- predefined events, 189
- presentation layer, 35
- presentation logic, 1, 261
- Presentation Model (PM) pattern, 262, 268
- presentation patterns, 261
- presentation tier, 28
- PresentationCore, 13
- PresentationFramework, 13
- PresentationModel, 268
- Presenter, 262–263, 265
- PreviewCanExecute event, 193
- PreviewExecuted event, 193
- primary keys, 34, 129
- principle of last knowledge, 27
- PrintCommand, 247
- process identity, 311
- production projects, 21
- ProgressChanged event, 287
- ProgressChangedEventArgs argument, 287

Project tab, Expression Blend, 45

properties

dependencies of, 14

modifying, 54–55

Properties panel, Expression Blend, 54–55

property element syntax, 5–6

property triggers, 93

PropertyChanged event, 160–161, 201,
269–270

PropertyChangedEventArgs argument, 161

PropertyChangedEventHandler class, 203

PropertyChangedEventManager class, 203

prototyping, 62–70

public attributes, 297

Publish command, 260

publish.htm page, 260

■ Q

queries, 114, 238–239

Query Designer window, 234–235

■ R

raster images, 48

.rdlc file extension, 227

.rdlc reports, 228

ReceiveWeakEvent method, 202

Rectangle control, 64

Rectangle-Sketch control, 64

red-green-refactor, 210, *See also* Test
Driven Development

references, 36–37, 39–41

region groups, 47

relationships, between entities, 128–134

RelayCommand, 196–198

ReportParameter array, 247

ReportProgress method, 287

reports, 227–249

creating dataset, 233–236

creating datasource, 232–233

designing, 236–237

detailed view, 237–240

displaying, in WPF view, 242–244

footer, editing, 236

header, editing, 236

introduction to, 227

loading

detailed view, 246–248

list view, 244–245

with parameters, 238–240

SQL Server Reporting Services (SSRS),
228–48

configuring, 228–231

creating dataset, 233–236

creating datasource, 232–233

creating reports, 231–232

designing report, 236–237

- detailed view, 237–241
 - hosting, in a WPF control, 241–246
 - TimeTracker, 231–241
- ReportViewer control, 227–228, 241–246
- ReportViewModel, 246
- Repository class, 117, 151
- Repository object, 304
- Repository pattern, 117–119, 121, 151–158
 - adding role, 152–153
 - additional methods, 157–158
 - deleting role, 153–154
 - GetAll roles, 155–156
 - GetById method, 156–157
 - IRepository interface, 151–152
 - updating role, 154–155
- RepositoryBase class, 151
- RepositoryBase object, 152
- representational state transfer (REST), 304
- RequerySuggested event, 200
- resource dictionary, 2, 88–91, 167–168
- Resource property, 14, 88
- ResourceDictionary.xaml file, 97
- resources, 87–88
- REST support, in WCF, 290
- Resume method, 280
- RichTextBox, 50
- Role domain entity, 122–127

- Role entity, 128–130, 152
- RoleRepository class, 152, 157
- RoleRepositoryFixture test class, 152
- RoleRepository delete method, 222
- roles
 - adding, 152–153
 - deleting, 153–154
 - GetAll method, 155–156
 - GetById method, 156–157
 - updating, 154–155
- root elements, 2–3
- RoutedCommand class, 177, 191–195, 204
- RoutedCommand object, 272
- RowDefinition property, 102
- RunWorkerAsync method, 286
- RunWorkerCompleted event, 287
- RunWorkerCompletedEventArgs event, 287

■ S

- sample data, 44, 57
- Samples tab, Expression Blend, 45
- SAP Crystal Reports, 227
- SaveChanges() method, 140, 143
- schedule controls, 108–109
- SearchCommand, 205–206
- SearchMethod, 205

- security
 - bindings, 309
 - certificates, 252–253, 258
 - ClickOnce, 252–253
 - Code Access Security, 253–254
 - credentials, 310
 - TimeTracker application, 259
 - WCF, 308–311
- security principal, 311
- SELECT statement, 240
- self-closed elements, 3
- self-reference relationships, 129
- self-signed certificates, 252
- separation of concerns concept, 27
- sequential execution, of code, 276
- [ServiceContract] attribute, 298
- ServiceDescription class, 292
- service-oriented applications, WCF and, 290
- Service-Oriented Architecture (SOA), 28, 289
- services, 289
 - adding methods to, 297–299
 - adding service references, 301–302
 - creating, in WCF, 295
 - customizing behavior of, 299–300
 - in WCF, 292, 299–302
 - testing, 300–301
- ServiceSecurityContext, 311
- setter property, 92
- shapes, drawing, 49–50
- Shared Dataset Properties dialog, 233
- ShowReportDialog method, 244
- ShutdownMode property, 79
- Silverlight, 1
- Simple Object Access Protocol (SOAP), 289–290
- single responsibility concept, 27
- single-thread design, 13
- Singleton pattern, 299
- Sketchable controls toolkit, 67
- SketchFlow controls, 64–65
- SketchFlow Map panel, 63–66
- SketchFlow plug-in, 21, 43–44, 59–60, 62, 67–69
 - behaviors, 67–69
 - creating mock-up and prototyping with, 62–70
- SketchRectangleUC, 64
- Sleep method, 278
- SOA applications, WCF and, 289
- SOAP communication protocol, 289–290
- solution file (.sln), 19
- solution folders, 35
- Solution Items folder, 217
- SQL Server 2008 R2 Express Edition, 26, 34, 133

- SQL Server Management Studio (SSMS), 34, 133
- SQL Server Reporting Services (SSRS), 227–228, 248
 - configuring, 228–231
 - creating reports, 231–232
 - creating dataset, 233–236
 - creating datasource, 232–233
 - designing report, 236–237
 - detailed view, 237–241
 - deployment of VS solution in, 232
 - hosting reports, 241–46
 - portal page, 230
- StackPanel CommandBinding collection, 194
- StackPanel control, 1, 11–12
- stand-alone applications, 9
- Start method, 279
- Startup method, 78
- startup process, custom, 78–79
- StartupUri attribute, 77
- static resources, 88
- stored procedures, 34, 114
- storyboards, 51–52
- styles, 14–15, 87–88
 - basic concepts, 91–93
 - control, 91–97
 - creating, 55–57
 - custom, 110–111
 - defining, 92
 - in Expression Blend, 44
 - for image button, 94–96
 - themes, 97
 - triggers, 93–94
- StyleTemplate, 12
- styling controls, 53–57
- Supervising Controller pattern, 266–268
- Suspend method, 280
- svcutil.exe, 292
- Syndication Service Library, 293
- syntax
 - binding, 170–171
 - XAML, 2–8
- System.ComponentModel namespace, 285
- System.ComponentModel.DataAnnotations assembly, 168, 171
- System.EventArgs class, 202
- System.Threading namespace, 279
- System.Threading.DispatcherObject class, 13
- System.Threading.Tasks namespace, 283
- System.Windows.Controls class, 14
- System.Windows.DependencyObject class, 14
- System.Windows.Forms namespace, 241
- System.Windows.FrameworkElement class, 14
- System.Windows.Media.Visual class, 14

System.Windows.SetterBase class, 92
 System.Windows.UIElement class, 14

■ T

tabbed panels, 47
 TabControl, 100–101
 table object, 34
 tables, grouping, 240
 TargetType attribute, 92
 Task object, 283–285
 Task Parallel Library, 285
 technologies, 14

- data templates, 15–16
- DataBinding, 17
- styles, 14–15

 template controls, 53–57
 TemplateColumn, 106
 templates, 26

- data, 15–16, 166–168
- in Expression Blend, 44

 temporary certificates, 258
 test client utility, WCF, 294–295
 Test Driven Development (TDD), 209–225

- advantages of, 210
- approach, 138–140
- code coverage, 210–212, 216–219
- TimeTracker application and, 219–224
- with MSTest, 212–216

test first principle, 209
 .testsettings file, 217
 test windows, 93
 testing, 138–140, 209–225

- calculator example, 213–215
- CleanUp method, 150
- for updated objects, 148–150
- Initialize method, 150
- RoleRepository class, 152
- transaction testing, 141–143
- unit, , 138–140, 209, 216

 text

- formatting, 51
- in Expression Blend, 50–51

 text controls, 50
 TextBlock, 12, 50
 TextBox, 12, 14, 50
 TextBoxColumn, 106
 themes, 97
 third-party DLLs, 38–41
 third-party resources, 38–41
 third-party tools, 26
 thread affinity, 13
 Thread class, 278–280
 Thread object, 276–280
 ThreadPool object, 282–283
 threads

- in background, 280

- pausing, 280
- resuming, 280
- starting, 279
- stopping, 279–280
- synchronizing, 280
- three-tier application architecture, 28–30
- TimeTracker application (example)
 - adding features to, 110–111
 - architecture, 27–31
 - commands, 204–207
 - configuration, 35–41
 - creating prototype, 62–65
 - custom startup process, 78–79
 - customizing UI, 91–97
 - data layer, 135–158
 - IUnitOfWork interface, 135–136
 - repository pattern, 151–158
 - testing code, 138–140
 - transaction management, 140–143
 - Unit of Work implementation, 137–150
 - database schema, 34, 133
 - deployment, 257–260
 - Details view, 69–70
 - dictionaries, 88
 - domain entities in, 121
 - address entity, 121
 - Address entity, 130–131
 - appointment entity, 121
 - Appointment entity, 134–135
 - Contact entity, 132
 - customer entity, 121
 - Customer entity, 133–134
 - Employee domain entity, 127–128
 - employee entity, 121
 - relationships between, 128–132
 - Role domain entity, 122–127
 - role entity, 121
 - domain model, 32–34, 122
 - Entity Framework, 122
 - external resources in, 38–41
 - integrating WCF into, 304–308
 - layer diagram, 30
 - layers, 35–38
 - list views, 66–67
 - ObjectContext, 123–125
 - overview, 27
 - publishing, 259–260
 - reports, creating, 231–241
 - requirements, 25–26
 - security settings, 259
 - Test Driven Development and, 219–224
 - user controls for dashboard, 107–108
 - user interface, 61
 - user stories, 31–32
 - user tasks, 62

- view models, 70–76
- views, adding, 97–109
- TimeTracker commands, 204–207
- TimeTracker domain, 121–135
- TimeTrackerModel.Designer.cs file, 123
- TimeTrackerModelContainer class, 125
- TimeTrackerModelContainer object, 137
- token authentication, 311
- tools, WPF 4, 17–23
 - additional, 22–23
 - express editions, 17
 - Expression Blend, 21–22
 - Visual Studio 2010, 18–21
- .ToString method, 15, 166
- transaction management, 140–143
- transitions, 44
- TransparentStop property, 111
- triggers, 93–94, 110–111
- Triggers property element, 110
- T-SQL code, 34
- two-way binding, 162
- TypeConverterAttribute, 4–5

■ U

- UI. *See* user interface (UI)
- UISpy.exe, 23
- unidirectional data binding, 159
- Unit of Work, 137–138
- IDisposable interface, 136
- IUnitOfWork interface, 135–136
- objects
 - adding new, 143–145
 - deleting, 145–146
 - modifying, 146–150
 - transaction management, 140–143
 - unit testing, 138–140
- Unit of Work (UoW) pattern, 119–121
- Unit Test wizard, 216
- unit testing, 139–140, 209, 216
- UnitOfWork class, 136–138
- Unity, 81–82
- Update method, 118, 154–155
- updates, 154–155, 254–255
- UpdateSourceTrigger property, 162
- use case diagrams, 31
- user cases, 31–32
- user controls, for dashboard, 107–108
- user input, validation of, 103–104
- user interface (UI)
 - controls, 159
 - creating, 1
 - customizing, 94, 166–168
 - adding features, 110–111
 - applying styles, 91–97
 - data binding and, 159
 - design patterns for, 261–273

- layers, 35
- mockups of, 59–60
- testing, 223–224
- user interface (UI) controls, 9
 - common controls, 11–12
 - controls containers, 10–11
 - creating with XAML or C#, 12
 - positioning, 10–11
- user interface (UI) design process, 61
 - creating mock-up and prototyping, 62–70
 - overview, 61–62
- user requests, reacting to, 189
- user stories, 31–32
- UserControl control, 2, 9, 102–108
- UTF-8, 1

■ V

- ValidatesOnDataError, 169
- validation
 - adding to ViewModels, 171–173
 - of bound data, 168–169
 - data, 103–104
- validation errors, 169
- validation rules, 168–172
- Validation.HasError property, 104, 168
- ValidationRule class, 168
- value objects, 128
- ValueConversion attribute, 164
- vector graphic images, 48
- VeriSign, 252
- VerticalAlignment property, 11
- ViewModel class, 204–207
- view models, 70–76
 - additional, 184–185
 - base implementation, 70
 - BaseDetailsViewModel, 184
 - BaseViewModel, 72–73, 76
 - binding model to, 271
 - child, 185–186
 - data binding, 171–187
 - Details, 73–74
 - Employee, 173–174
 - EmployeeDetailsViewModel, 175–180
 - EmployeeListViewModel, 180–184
 - exposing ICommand in, 272
 - INotifyPropertyChanged interface, 70–72, 171
 - List, 74–75, 184
 - in MVVM pattern, 269–271
 - testing, 223
 - validators, 171–173
- views, 10, 34
 - adding controls to, 87–112
 - bound to ViewModels, 174
 - creating, 61–85, 97–109

- Details view, 100–104
- List view, 104–106
- Login view, 98–100
- mock-up and prototyping, 62–70
- overview, 61–62
- Detail view, 69–70, 73–74, 100–104
- list, 66–67, 104–106
- in MVVM pattern, 269
- navigation between, 77–79
- opening, 110
- in Presentation Model pattern, 268
- for report viewer, 243
- in Supervisor Controller pattern, 266–267
- testing, 224
- Visual Studio, 1
- Visual Studio 2010, 8, 18–21, 26
 - code coverage with, 216–219
 - creating application in, 35
 - reporting in, 227–246
 - solution folders, 35
 - test settings, 218
 - testing in, 209

■ W

- WCF. *See* Windows Communication Foundation (WCF)
- WCF Data Service, 304–308
- WCF Service Application, 293

- WCF Service Library, 293
- WCF Workflow Service Application, 293
- weak event pattern, 202
- WeakEvent, 202
- WeakEventListener, 202–203
- WeakEventManager object, 202–203
- Web performances test, 216
- web services, 289
- WHERE SQL statement, 146
- Window object, 2, 286
- Window root element, 2
- Windows Azure AppFabric, 290
- Windows Communication Foundation (WCF), 28, 289–311
 - adding methods to services, 297–299
 - adding service references, 301, 302
 - architecture, 291–292
 - creating service, 295
 - customizing services, 295–300
 - DataContracts, 296–297
 - Data Service, 304–308
 - endpoints, 291–292
 - features, 290
 - hosting domain for, 304
 - integrating into TimeTracker application, 304–308
 - interacting with, 289–312
 - introduction to, 289–292

- project types, 293
- sample project, 292–304
- security, 308–311
- service and clients, 292
- test client utility, 294–295
- testing endpoints, 302–303
- testing services, 300–301
- uses of, 290
- web settings for, 294
- Windows Forms control, 14, 241–246
- Windows Forms ReportViewer control, 228
- Windows Presentation Foundation (WPF), 1
 - architecture, 13–17
 - assemblies and CLR, 13–14
 - milcore, 13
 - PresentationCore, 13
 - PresentationFramework, 13
 - technologies, 14–17
 - integrating WCF Data Service in, 307–308
 - introduction to, 8–23
 - themes, 97
 - tools, 17–23
 - additional, 22–23
 - express editions, 17
 - Expression Blend, 21–22
 - Visual Studio 2010, 18–21
 - UI controls, 9–12
 - use of external resources by, 39–41
- Windows Workflow Foundation (WF), 1, 290
- WindowsFormsHost control, 241–243
- WindowsFormsIntegration.dll assembly, 241
- WorkerDoWork method, 286
- WPF. *See* Windows Presentation Foundation (WPF)
- WPF Application Structure, 36
- WPF applications
 - in Visual Studio 2010, 20
 - in-browser, 9
 - multithreading and, 275–288
 - stand-alone, 9
 - WPF Databound Application, 22
- WPF Binding class, 160
- WPF Databound Application, 22
- WPF dispatcher, 279
- WPF Schedule Control, 108
- WPF SketchFlow Application, 62
- WPF Toolkit, 26, 106–108
- WrapPanel, 11

■ **X, Y, Z**

XAML (Extensible Application Markup Language), 1

code, 1

events in, 7

syntax, 2–8

attached properties, 8

attributes, 4

code behind, 6–7

content property, 6

namespaces, 2–3

objects and common elements, 3–6

property elements, 5–6

root elements, 2–3

TypeConverterAttribute, 4–5

.xaml file extension, 1

XAML files, parts of, 20

XAML markup, 20, 43, 110

XAML Power Toys, 23

XAML PAD, 23

.xbap extension, 9

x:Key attribute, 15, 87–88, 92, 164

XML data sources, 57

xmlns attribute, 3

xmlns:x attribute, 3