

# Index

## A

**AbstractDecoratedTextEditor class, 359**

**AbstractTextEditor class, 357–359**

**accessibility options (SWT), 202–203**

**actions (JFace), 226–227, 367**

defining

attributes, 368–370

delegates, 371–372

enabling and disabling, 370–371

in the manifest, 368

selection events, 370

global, 367

implementing manually, 372

local, 367

**active help systems, 382, 474–475**

**adapters, SWT events, 142**

**Add Java Exception Breakpoint function, 117**

**agile programming. See extreme programming**

**annotations (JFace), 223–224**

**Ant, 388**

configuring, 388–389

editing scripts, 389–390

**ANTLR, 547**

**applications, exporting (Duke Speaks project), 109–110**

**architecture, Eclipse platform**

core runtime, 315

extension points, 314

help system, 316

OSGi, 314

overview, 314, 316–317

RCP vs. IDE, 315

resource management, 315

team support, 316–317

user interfaces, 316

**AspectJ, 547**

**associations, 54–55**

**Attrezzo per Xindice, 545**

**attributes, 342**

**audio players (FreeTTS), 75**

**automated testing, 538**

**automatic code completion**

Code Assistant, 28–30, 77–78

comment out code, 31

delegate methods, 32

encapsulating fields, 32

generating constructors, 33

importing types, 31

inheriting constructors, 32

Javadoc comments, 31

overriding methods, 32

**AWT components with SWT, 191**

embedded contents, 192

events, 192–196

**Azzurri Clay, 547**

## B

**Back function, 37, 53**

**backing up workspaces, 2**

**beans, 48**

generic, 48

properties, 48–49

**Beck, Kent, 540**

**bookmarks**

bookmark manager, 366

creating, 22

**breakpoints**

managing, 117–118

setting, 116

**Breakpoints View, 117–118**

**browser widget (SWT), 177**

**build.properties files (spell checker project), 493–495**

**BundleActivators (OSGi), 314**

**BusyIndicator class (SWT), 175**

**buttons**

Duke Speaks project GUI, 94–95

SWT, 156

## C

**Call Hierarchy View, 27–28**

**CDT, 547**

**cell editors**

Jukebox project, 285–286, 299–300

overview, 217–218

**central code management**

comparison functions, 135

conflict resolution, 133–134

CVS (Concurrent Versions System), 129–130

Export function, 135

repositories, 130–131

sharing projects, 132

Team group preferences, 135

version management, 133

## **Change Method Signature function, 39**

### **Cheat Sheets**

- internationalization, 404
- plug-ins, 383–384

### **CheckedTreeSelectionDialog class, 373**

### **classes, creating, 7–9, 76**

### **classfiles, compiler preferences, 15**

### **clipboard (SWT), 198–199**

### **closing windows, 65**

### **code, formatting, 15–16**

### **Code Assistant**

- Duke Speaks project, 77–78
- overview, 28–30

### **code folding (Java Editor), 35–36**

### **code generation templates. See templates**

### **code organization**

- associations, 54–55
- Find/Replace function, 62–63
- importing files, 67–69
- Java Browsing Perspective, 71
- navigation tools, 53
- Outline View, 57–58
  - context functions, 59–60
  - representation, 58–59
- packages, 55–57
- perspectives
  - configuring, 66–67
  - defining, 65–66
- project properties, 69–71
- resources, 52
  - storing, 52–53
  - synchronizing, 53
  - types, 52
- Search function, 60–62
- windows
  - closing, 65
  - docking, 63
  - FastView, 64
  - maximizing, 65
  - minimizing, 65
  - opening, 65
  - stacking, 64
- workbench, 51–52

### **collaborative projects, 129–130**

- comparison functions, 135
- conflict resolution, 133–134
- CVS (Concurrent Versions System), 129–130
- Export function, 135
- plug-ins, 316–317
- repositories, 130–131
- sharing projects, 132
- Team group preferences, 135
- version management, 133

### **collapsible GUI elements, 346–348**

### **colors**

- JFace, 206
- SWT, 186–187

### **column headers (Jukebox project), 285–286**

### **ColumnLayout layout manager, 346**

### **ComboContentAssistSubjectAdapter class (JFace), 226**

### **combos (SWT), 165–166**

### **command line options, 4–5**

### **comment out code, 31**

### **Compare Editor, 133–134**

### **Compare With function, 60**

### **comparing resources, 43**

### **compilation units**

- moving, 39
- renaming, 39

### **compilers, preferences, 14–15**

- classfiles, 15
- JDK compliance, 15
- warnings and errors, 14–15

### **component events, processing, 353**

### **components**

- VE (Visual Editor for Java), 47
- workbench, 351–352

### **composites (SWT), 155**

### **Concurrent Versions System (CVS), 129–130**

### **configuring**

- Ant, 388–389
- debugger, 113–114
- perspectives, 66–67
- plug-ins
  - manifest, 329–332
  - Plug-in Development perspective, 327–329
  - Schema Editor, 341–343
  - SDK extension points, 332–341
- text processing (JFace), 223

### **conflicts, resolving, 133–134**

### **constants, refactoring code, 42**

### **constructors**

- generating, 33
- inheriting, 32

### **ContainerGenerator class, 374**

### **containers, 322**

### **ContainerSelectionDialog class, 374**

### **content assistants**

- Jukebox project, 304–307
- overview, 224–225

### **content pane (Duke Speaks project), 96**

### **ContentViewer class (JFace), 216**

### **context associations (help systems), 380–381**

### **context identifications (help systems), 381**

### **context-sensitive help, 26, 473–474**

### **Control class (SWT), 146**

### **ControlEditor class (SWT), 175**

### **Convert Anonymous Type to Nested Type function, 41**

**Convert Local Variable to Field function, 42**  
**Convert Nested Type to Top Level function, 41**  
**CoolBar class (SWT), 170**  
**Copy function, 60**  
**Correction Assistant**  
 Quick Assist, 34–35  
 QuickFix, 33–34  
**Countour Generator (FreeTTS), 74**  
**Create a New Class Wizard, 7–8**  
**cursors (SWT), 189–190**  
**custom widgets (SWT), 174–176**  
**Cut function, 60**  
**CVS (Concurrent Versions System), 129–130**

## D

**data model separation (forms), 350**  
**data transfer**  
 JFace, drag and drop, 218  
 SWT  
 clipboard, 198–199  
 drag and drop, 199  
**debugger. See also troubleshooting**  
 breakpoints  
 managing, 117–118  
 setting, 116  
 configuring, 113–114  
 controlling program execution, 115–116  
 Debug Perspective, 114–115  
 HotSwap, 117  
 testing expressions, 117  
 variables, 116–117  
**debugging, 118–119. See also debugger**  
**Debug Perspective, 114–115**  
**Declarations function, 60**  
**declaring applications, 514**  
**defining perspectives, 65–66**  
**delegate methods, 32**  
**Delete function, 60**  
**deployment**  
 customizing products, 394–396  
 Eclipse runtime environment, 394  
 example files, 396–398  
 features, 391  
 creating, 391–392  
 deploying, 393  
 fragments, 390  
 Hex Game project, 534  
 installation formats, 388, 394  
 internationalization, 401–402  
 Cheat Sheets, 404  
 help systems, 404  
 national language resource bundles, 405  
 text constants in manifest files, 403  
 text constants in programs, 402–403  
 patches, 405

RCP (Rich Client Platform) applications, 507–508  
 spell checker project, 490  
 configuring Ant scripts, 492–495  
 feature, defining, 490–492  
 installation, 498  
 language feature, defining, 495–497  
 update site, defining, 497–498  
 update sites, 398–399  
 adding, 400  
 install handlers, 401  
 installing features, 400  
 managing configuration, 400–401  
 updating features, 400  
 zip files, 394  
**description editor (Jukebox project), 300–301**  
**description editor dialog (Jukebox project), 302–303**  
 code scanner, 303  
 content assistant, 304–307  
 SourceViewer, 307–311  
**description window (Jukebox project), 275–278**  
**Deselect Working Set function, 53**  
**development teams, 129–130**  
 comparison functions, 135  
 conflict resolution, 133–134  
 CVS (Concurrent Versions System), 129–130  
 Export function, 135  
 plug-ins, 316–317  
 repositories, 130–131  
 sharing projects, 132  
 Team group preferences, 135  
 version management, 133  
**dialogs**  
 JFace, 206–207  
 implementing, 210–213  
 InputDialog class, 207–208  
 making persistent, 213–214  
 MessageDialog class, 208–209  
 TitleAreaDialog class, 209–210  
 Jukebox project  
 description editor, 302–311  
 file-selection, 292–293  
 plug-ins, 372–373  
 CheckedTreeSelectionDialog class, 373  
 ContainerGenerator class, 374  
 ContainerSelectionDialog class, 374  
 ElementListSelectionDialog class, 374  
 ElementTreeSelectionDialog class, 373  
 ListSelectionDialog class, 374  
 NewFolderDialog class, 374  
 ResourceListSelectionDialog class, 374  
 SaveAsDialog class, 374  
 SWT, 152–153  
 message boxes, 154–155  
 predefined, 153–154  
**Disable function, 117**  
**Disconnect function, 115**

# Display class (SWT)

---

## **Display class (SWT), 147–148**

**docking windows, 63**

**documentation, 125–127, 343**

**drag and drop**

JFace, 218

Jukebox project, 286–288

SWT, 199

**Duke Speaks project**

animation events, 75–77

animator, 77–80

Animator.java class, 78–80

Code Assistant, 77–78

creating classes, 77

embedding into FreeTTS, 81–83

connecting to the Java audio system, 83–84

exporting, 109–110

FreeTTS speech synthesizer, 74–75

Player class source code, 106–109

setup, 73–74

third-party software, 555

user interface, 84

animated face, 84–86

buttons, 94–95

content pane, 96

control panel, 87

domain model, 87–91

events, 94

integrating, 95

labels, 94

layouts, 93

PlayerPanel class source code, 97–106

presentation, 91–92

sliders, 93

text, 94

VE (Visual Editor for Java), 92

**dynamic help systems, 382**

## **E**

**easysql, 545**

**Eclipse**

core classes, 318–320

installing, 1–5

migrating to new versions, 551–553

platform architecture

core runtime, 315

extension points, 314

help system, 316

OSGi, 314

overview, 314, 316–317

RCP vs. IDE, 315

resource management, 315

team support, 316–317

user interfaces, 316

requirements, 1

Welcome screen, 3

**Eclipse 3, migrating to, 552–553**

**Eclipse Modeling Framework. See EMF**

**Eclipse UI, 344**

actions, 367

defining in the manifest, 368–372

global, 367

implementing manually, 372

local, 367

dialogs, 372–373

CheckedTreeSelectionDialog class, 373

ContainerGenerator class, 374

ContainerSelectionDialog class, 374

ElementListSelectionDialog class, 374

ElementTreeSelectionDialog class, 373

ListSelectionDialog class, 374

NewFolderDialog class, 374

ResourceListSelectionDialog class, 374

SaveAsDialog class, 374

forms, 344–346

collapsible elements, 346–348

data model separation, 350

layouts, 346

Master-Details-Block, 350

resource management, 349

text markup, 348–349

help system, 379

active help, 382

Cheat Sheets, 383–384

context associations, 380–381

context identifications, 381

deploying, 381–382

dynamic help, 382

table of contents, 379–380

perspectives, defining, 377–379

preferences, 377

property pages, 377

wizards, 374

IWorkbenchWizard interface, 376

newWizard extension point, 375–376

WizardNewFileCreationPage class, 376–377

workbench, 350–351

components, 351–352

editors, 355–362

event processing, 352–355

views, 362–366

workbench page, 351

workbench window, 351

**Eclipse Visual Editor for Java. See VE**

**Edit Active Working Set function, 53**

**EditorPart class, 355**

**editors, workbench, 355–356**

AbstractTextEditor class, 357–359

FormEditor class, 360–361

keyboard shortcuts, 357

markers, 361–362

- menus, 356
  - MultiEditor class, 360
  - saving files, 357
  - status line, 357
  - StatusTextEditor class, 359
  - TextEditor class, 360
  - toolbars, 356
  - Eiffel for Eclipse, 547**
  - ElementListSelectionDialog class, 374**
  - ElementTreeSelectionDialog class, 373**
  - embedded Ant. See Ant**
  - embedding external tools, 135–136**
  - EMF (Eclipse Modeling Framework), 45, 546**
  - Enable function, 117**
  - Encapsulate function, 41–42**
  - encapsulating fields, 32**
  - Error Log, 26**
  - errors, 14–15. See also Problems view**
  - event processing, 49–50**
  - events**
    - Duke Speaks project GUI, 94
    - SWT events package, 141
      - adapters, 142
      - events, 142
      - listeners, 141–142
      - overview, 143–145
    - SWT with Swing/AWT components, 192–196
    - workbench, 352–353
      - component events, 353
      - initialization, 354–355
      - selection events, 354
      - sequence, 355
      - window events, 353
  - executable prototypes, 538**
  - ExpandableComposite class, 346–347**
  - exporting**
    - applications (Duke Speaks project), 109–110
    - preferences, 10
  - expressions, testing, 117**
  - Expressions view (scrapbooks), 24**
  - extension points, 314, 514**
  - external tools, embedding, 135–136**
  - Extract/Inline Constant function, 42**
  - Extract Interface function, 40**
  - Extract Local Variable function, 41**
  - Extract Method function, 39**
  - extreme programming, 537**
    - automated testing, 538
    - executable prototypes, 538
    - restructuring code, 540–541
- F**
- factories, refactoring code, 40**
  - FastView, 64, 378**
  - Feature Patch, 405**
  - features, 25, 391**
    - creating, 391–392
    - deploying, 393
    - disabling, 400
    - installing, 400
    - spell checker project
      - about.html file, 492
      - about.ini file, 492
      - manifest, 491
      - project, creating, 490–491
      - updating, 400
  - field editors (JFace), 232–233**
  - fields, encapsulating, 32**
  - file associations, 54–55**
  - files, 323–324**
    - importing, 67–69
    - saving (workbench editor), 357
  - file-selection dialogs (Jukebox project), 292–293**
  - FillLayout class (SWT), 178–179**
  - Filters function, 53**
  - Find/Replace function, 62–63**
  - folders, 323**
  - FontRegistry class (JFace), 205–206**
  - fonts (SWT), 187–188**
  - formatting code, 15–16**
  - FormEditor class, 360–361**
  - FormLayout class (SWT), 182–184**
  - forms, 344–346**
    - collapsible elements, 346–348
    - data model separation, 350
    - layouts, 346
    - Master-Details-Block, 350
    - resource management, 349
    - text markup, 348–349
  - FormText class, 348–349**
  - Forward function, 37, 53**
  - fragments, 390**
    - spell checker project
      - help files, 489
      - Java-Properties plug-in, 490
      - manifest texts, 488
      - program texts, 488
      - project, creating, 487–488
  - FreeTTS speech synthesizer, 67. See also Duke Speaks project**
    - downloading, 67–68
    - importing, 68–69
    - overview, 74–75
    - Programmer's Guide, 75
- G**
- GEF, 545**
  - Generalize Type function, 40**
  - generating constructors, 33**
  - generic beans, 48**

**global actions, 367**

**global welcome screen (RCP), 508–509**

**Go Into function, 53**

**Go to Line... function, 37**

**GotoMarker, 362**

**Graphics Context class (SWT), 185–186**

**graphics package (SWT), 185**

colors, 186–187

cursors, 189–190

fonts, 187–188

Graphics Context class, 185–186

images, 188–189

**GridLayout class (SWT), 180–182**

**groups (SWT), 156**

**GUI extension points, 334–338**

**GUIs. See also JFace; SWT**

composing (with VE), 46–48

Duke Speaks project

animated face, 84–86

buttons, 94–95

content pane, 96

control panel, 87

domain model, 87–91

events, 94

integrating, 95

labels, 94

layouts, 93

PlayerPanel class source code, 97–106

presentation, 91–92

sliders, 93

text, 94

VE (Visual Editor for Java), 92

Eclipse platform architecture, 316

Eclipse UI, 344

actions, 367–372

Cheat Sheets, 383–384

dialogs, 372–374

forms, 344–350

help system, 379–382

perspectives, defining, 377–379

preferences, 377

property pages, 377

wizards, 374–377

workbench, 350–366

Jukebox project

description window, 275–278

Player module, 247–249

playlist viewer, 278–295

## H

**Hello World application**

creating new classes, 7–9

launching, 9–10

perspectives, opening, 5–7

project, creating, 7

talking Hello World example, 67–71

**help systems, 26–27, 316**

Hex Game project, 516

internationalization, 404

plug-ins, 379

active help, 382

context associations, 380–381

context identifications, 381

deploying, 381–382

dynamic help, 382

table of contents, 379–380

spell checker project, 473

active help, 474–475

context-sensitive help, 473–474

Java properties plug-in, 483–484

running the help action, 476–477

table of contents, 473

**Hex Game project**

customizing, 515

declaring the application, 514

deployment, 534

game engine, 527–531

help system, 516

HexView class, 521–527

IGame interface, 520–521

IStatusListener interface, 520

manifest, 512–513, 516–517

overview, 511–512

perspective, defining, 515

plug-ins required, 514

RcpApplication class, 517–518

RcpPerspective class, 519–520

RcpWorkbenchAdvisor class, 518–519

setting up, 512

texting, 534

view, defining, 515

welcome screen, 515, 531–533

**hierarchy**

packages, 56–57

viewers (JFace), 215–216

**HotSwap, 117**

**hover infos, 27, 222**

## I

**IAction interface (JFace), 226–227**

**IContainer interface, 322**

**IDE (Integrated Development Environment), RCP comparison, 315**

**IEditorInput interface, 355**

**IFile interface, 323–324**

**IFolder interface, 323**

**ILabelProvider class (JFace), 216**

**ImageRegistry class (JFace), 206**

**images (SWT), 188–189**

**I****IMarker interface, 324–325****importing**

- files, 67–69
- preferences, 10
- types, 31

**Improve, 547****incremental backups, 2****inheriting constructors, 32****INI files (Jukebox project), 261–268****Inline method or local variable function, 41****input dialogs (JFace), 207–208****installation formats, 388, 394****installed JREs preferences, 12–13****install handlers, 401****installing**

- Eclipse, 1–5
- VE (Visual Editor for Java), 45

**interfaces, creating, 76–77****internationalization, 401–402**

- Cheat Sheets, 404
- help systems, 404
- national language resource bundles, 405
- spell checker project, 484
  - language fragment, creating, 487–490
  - text constants in Java code, 484–486
  - text constants in manifest files, 487
- text constants in manifest files, 403
- text constants in programs, 402–403

**Introduce Parameter function, 39****Intro View, 26****IPath interface, 319****IPlatformRunnable interface (RCP), 503****IProgressMonitor interface, 320****IResourceChangeListener interface, 325–326****IResource interface, 320–322****IWorkbench object, 351****IWorkbenchWizard interface, 376****IWorkspaceRoot interface, 322****J****J2SE 5.0, 541–542****JAR files (Duke Speaks project), 109–110****Java 1.5, 541–542****Java Beans. See beans****Java Beans view, 48****Java Browsing Perspective, 71****Java Build Path, editing, 74****JavaCC, 547****Java classes, SDK extension points, 334****Java console, debugging, 118–119****Java Debug Wire Protocols (JDWP), 119****Javadoc**

- comments, creating, 31
- documentation, 125–127

**Javadoc View, 27–28****Java Editor**

- code folding, 35–36
- syntax coloring, 36
- typing aids, 35

**Java properties plug-in**

- internationalization, 490
- spell checker project, 477
  - help system, 483–484
  - JavaPropertiesPlugin class, 480–481
  - JavaPropertiesPreferencePage class, 482–483
  - Java-Properties tokenizer, 483–484
- manifest, 478, 479–480
- preferences, 481–482
- setup, 477
- tokenizer extension, 478–479

**Java Speech API (JSAPI), 67. See also Duke Speaks project****JavaZoom sound modules, 238****JDK compliance, 15****JDWP (Java Debug Wire Protocols), 119****JellySWT, 546****JFace. See also Jukebox project**

- dialogs, 206–207
  - implementing, 210–213
  - InputDialog class, 207–208
  - making persistent, 213–214
  - MessageDialog class, 208–209
  - TitleAreaDialog class, 209–210
- Action interface, 226–227
- managers, 227
  - MenuManager, 227–228
  - StatusLineManager, 228
- preferences, 230–231
  - field editors, 232–233
  - PreferenceConverter class, 231
  - PreferencePage class, 232
  - preference page trees, 233–234
  - PreferenceStore class, 231
- resource management
  - FontRegistry class, 205–206
  - ImageRegistry class, 206
  - JFaceColors class, 206
  - JFaceResources class, 206
- text processing, 218
  - annotations, 223–224
  - ComboContentAssistSubjectAdapter class, 226
  - configuring, 223
  - content assistants, 224–225
  - document model, 218–219
  - ProjectionViewer class, 226
  - scripts, 219–220
  - TextContentAssistSubjectAdapter class, 226
  - text formatters, 224
  - text presentation, 225–226
  - TextViewer class, 220–223

- viewers, 214
  - cell editors, 217–218
  - ContentViewer class, 216
  - data transfer, 218
  - event model, 215
  - hierarchy, 215–216
  - ILabelProvider class, 216
  - StructuredViewer class, 216–217
  - TreeView class, 217
- windows, 206–207
- wizards, 228
  - Wizard class, 228–229
  - WizardDialog class, 230
  - WizardPage class, 229–230
  - WizardSelectionPage class, 230

**JFaceColors class, 206**

**JFaceDbc, 545**

**JFaceResources class, 206**

**Jigloo, 546**

**JREs**

- adding, 12–13
- version 1.1, 13

**JSAPI (Java Speech API), 67. See also Duke Speaks project**

**Jukebox project**

- deploying, 311
- description editor, 300–301
- DescriptionEditorDialog class, 302–303
  - code scanner, 303
  - content assistant, 304–307
  - SourceViewer, 307–311
- description window, 275–278
- design goals, 237–238
- FileCellEditor class, 299–300
- JARs needed, 240
- Java Build Path, 240
- JavaZoom sound modules, 238
- jlGui 2.2, 238
- Player class, 247–249
- Player module, 241
  - BasicPlayerListener interface, 260
  - graphics operations, 249–250
  - instrumentation, 251–258
  - layout, 241–242
  - Player class source code, 243–247
  - threads, 242–243
  - window management, 258–260
- playlist domain model, 261
  - accessing features, 270–271
  - interface, 261–268
  - IPlaylist implementation, 268–270
  - ISelectionProvider interface, 274–275
  - IStructuredContentProvider interface, 273
  - managing entries, 271–273
  - switching playlists, 273–274

- PlaylistLabelProvider class, 295–296
  - cell text, 297–299
  - returning warning icons, 296–297
- playlists, storing, 238
- playlist viewer, 275–278
  - file-selection dialogs, 292–293
  - menus, 293–295
  - nested grid layout, 289–290
  - PlaylistViewer class, 281–288
  - PlaylistWindow class, 278–281
  - toolbars, 290–292
- project installation, 238, 240–241
- third-party software, 555
- UML class diagram, 239

### **JUnit, 120**

- setup, 120–121
- test suites, 122–124

## **K**

**keyboard shortcuts, defining, 11–12, 357**

**KLEEN, 546**

## **L**

**Label class (SWT), 161**

**labels (Duke Speaks project), 94**

**labels (SWT), 161**

**Last Edit Location function, 37**

**launching applications, 9–10**

**layout package (SWT), 177–178**

- FillLayout class, 178–179
- FormLayout class, 182–184
- GridLayout class, 180–182
- RowLayout class, 179–180
- StackLayout class, 184–185

**layouts, 49**

- Duke Speaks project, 93
- Eclipse UI, 346
- perspectives, 378

**Link with Editor function, 53**

**listeners, SWT events, 141–142**

**lists (SWT), 164**

**ListSelectionDialog class, 374**

**local actions, 367**

**local debugging. See debugger**

**Local History functions, 43–44**

**long-running processes, managing, 326**

## **M**

**MagicDraw, 546**

**managers (JFace), 227**

- MenuManager, 227–228
- spell checker project, 423–424
- StatusLineManager, 228

**managing long-running processes, 326**

**markers, 324–325**

- declaring, 361
- GotoMarker, 362
- inheritance, 361–362
- lifecycle, 362

**marking name occurrences, 63****Master-Details-Block design pattern, 350****maximizing views, 65****maximizing windows, 65****MenuManager (JFace), 227–228****menus**

- creating manually, 356
- Jukebox project, 293–295
- overview, 170–174

**message boxes**

- overview, 208–209
- SWT, 154–155

**methods**

- call hierarchies, viewing, 27–28
- delegate, 32
- overriding, 32
- refactoring code, 40–41

**migrating to new Eclipse versions, 551–553****minimizing views, 65****minimizing windows, 65****modifying types, 38–39****Monitor class (SWT), 152****monitoring long-running processes, 320****movable tool groups (SWT), 170****Move function, 39****MultiEditor class, 360****MyEclipse, 548****N****name occurrences, marking, 63****national language resource bundles, deploying, 405****navigation tools, 36–37, 53**

- Outline View, 57–58
  - context functions, 59–60
  - representation, 58–59
- Package Explorer, 56

**nested grid layout (Jukebox project), 289–290****NewFolderDialog class, 374****newWizard extension point, 375–376****Next Annotation function, 37****NullProgressMonitor class, 320****O****Occurrences in File function, 60****OLE, SWT support, 201****Omondo, 546****Open Call Hierarchy function, 37****Open Call Hierarchy function, 59****Open Declaration function, 36–37****opening windows, 65****Open in New Window function, 53****Open Service Gateway Initiative (OSGi), 314****Open Super Implementation function, 37, 59****Open Type Hierarchy function, 37, 59****org.eclipse.core.runtime package, 318–320****org.eclipse.swt.accessibility package, 138****org.eclipse.swt.awt package, 138****org.eclipse.swt.browser package, 138****org.eclipse.swt.custom package, 138****org.eclipse.swt.dnd package, 139****org.eclipse.swt.events package, 139, 141**

- adapters, 142
- events, 142
- listeners, 141–142
- overview, 143–145

**org.eclipse.swt.graphics package, 139, 185**

- colors, 186–187
- cursors, 189–190
- fonts, 187–188
- Graphics Context class, 185–186
- images, 188–189

**org.eclipse.swt.internal package, 139****org.eclipse.swt.layout package, 139, 177–178**

- FillLayout class, 178–179
- FormLayout class, 182–184
- GridLayout class, 180–182
- RowLayout class, 179–180
- StackLayout class, 184–185

**org.eclipse.swt.ole.win32 package, 139****org.eclipse.swt.printing package, 139****org.eclipse.swt.program package, 139****org.eclipse.swt.widgets package, 139, 145**

- browser widget, 177
- buttons, 156–157
- combos, 165–166
- composites, 155
- Control class, 146
- custom widgets, 174–176
- dialogs, 152–153
  - message boxes, 154–155
  - predefined, 153–154
- Display class, 147–148
- groups, 156
- labels, 161
- lists, 164
- menus, 170–174
- Monitor class, 152
- movable tool groups, 170
- overview, 146
- ProgressBar class, 159
- sashes, 167–168
- scales, 158–159
- Scrollable class, 159
- Shell class, 148–150
- sliders, 158–159
- tabbed folders, 168–169

- tables, 161–164
- text fields, 159–160
- toolbars, 169–170
- trees, 166–167
- Widget class, 146

### **org.eclipse.swt package, 138**

#### **organizing code**

- associations, 54–55
- Find/Replace function, 62–63
- importing files, 67–69
- Java Browsing Perspective, 71
- navigation tools, 53
- Outline View, 57–58
  - context functions, 59–60
  - representation, 58–59
- packages, 55–57
- perspectives
  - configuring, 66–67
  - defining, 65–66
- project properties, 69–71
- resources, 52
  - storing, 52–53
  - synchronizing, 53
  - types, 52
- Search function, 60–62
- windows
  - closing, 65
  - docking, 63
  - FastView, 64
  - maximizing, 65
  - minimizing, 65
  - opening, 65
  - stacking, 64
- workbench, 51–52

### **OSGi (Open Service Gateway Initiative), 314**

#### **Outline View, 57–58, 365–366**

- context functions, 59–60
- representation, 58–59

#### **outputting to printers (SWT), 196–198**

#### **overriding methods, 32**

## **P**

#### **Package Explorer, 55–57**

#### **packages**

- hierarchy, 56–57
- organizing, 55–57

#### **PageBookView class, 364–365**

#### **parsing (Jukebox project), 261–268**

#### **Paste function, 60**

#### **patches, 405**

#### **Path class, 319**

#### **Pause Generator (FreeTTS), 74**

#### **PDA platforms (SWT), 202**

#### **persistency, workbench views, 363**

#### **persistent dialogs (JFace), 213–214**

#### **perspectives**

- configuring, 66–67
- defining, 65–66, 377–378
  - Action Sets, 378–379
  - FastViews, 378
  - layout, 378
- Hex Game project, 515
- opening, 5–7
- workbench components, 352

#### **Phraser (FreeTTS), 74**

#### **PitchMark Generator (FreeTTS), 75**

#### **Platform class, 318**

#### **Player module (Jukebox project), 241**

- BasicPlayerListener interface, 260
- graphics operations, 249–250
- GUI, creating, 247–249
- instrumentation, 251–258
- layout, 241–242
- Player class source code, 243–247
- threads, 242–243
- window management, 258–260

#### **playlist domain model (Jukebox project), 261**

- accessing features, 270–271
- interface, 261–268
- IPlaylist implementation, 268–270
- ISelectionProvider interface, 274–275
- IStructuredContentProvider interface, 273
- managing entries, 271–273
- switching playlists, 273–274

#### **playlist viewer (Jukebox project), 278**

- file-selection dialogs, 292–293
- menus, 293–295
- nested grid layout, 289–290
- PlaylistViewer class, 281–288
- PlaylistWindow class, 278–281
- toolbars, 290–292

#### **Plugin class, 318–319**

#### **Plug-in Development perspective, 327–329**

#### **Plug-in Registry, 25**

#### **plug-ins. See also Hex Game project; spell checker**

##### **project**

- actions, 367
- defining in the manifest, 368–372
- global, 367
- implementing manually, 372
- local, 367
- ANTLR, 547
- AspectJ, 547
- Attrezzo per Xindice, 545
- Azzurri Clay, 547
- CDT, 547
- Cheat Sheets, 383–384
- dialogs, 372–373
  - CheckedTreeSelectionDialog class, 373
  - ContainerGenerator class, 374

- ContainerSelectionDialog class, 374
- ElementListSelectionDialog class, 374
- ElementTreeSelectionDialog class, 373
- ListSelectionDialog class, 374
- NewFolderDialog class, 374
- ResourceListSelectionDialog class, 374
- SaveAsDialog class, 374
- easysql, 545
- Eclipse platform architecture
  - core runtime, 315
  - extension points, 314
  - help system, 316
  - OSGi, 314
  - overview, 314, 316–317
  - RCP vs. IDE, 315
  - resource management, 315
  - team support, 316–317
  - user interfaces, 316
- Eclipse UI, 344
- Eiffel for Eclipse, 547
- EMF (Eclipse Modeling Framework), 546
- features, 391
  - creating, 391–392
  - deploying, 393
  - disabling, 400
  - installing, 400
  - updating, 400
- forms, 344–346
  - collapsible elements, 346–348
  - data model separation, 350
  - layouts, 346
  - Master-Details-Block, 350
  - resource management, 349
  - text markup, 348–349
- fragments, 390
- GEF, 545
- help system, 379
  - active help, 382
  - context associations, 380–381
  - context identifications, 381
  - deploying, 381–382
  - dynamic help, 382
  - table of contents, 379–380
- Improve, 547
- installation formats, 388
- JavaCC, 547
- JellySWT, 546
- JFaceDbc, 545
- Jigloo, 546
- KLEEN, 546
- MagicDraw, 546
- manifest, 329–330
  - Build page, 332
  - Dependencies page, 331
  - Extension Points page, 332
  - Extensions page, 331–332
  - Overview page, 331
  - Runtime page, 331
  - workbench components, 352
- migrating to new Eclipse versions, 552
- MyEclipse, 548
- Omondo, 546
- perspectives, defining, 377–379
- Plug-in Development perspective, 327–329
- preferences, 377
- Profiler, 548
- property pages, 377
- Rational, 548
- RCP (Rich Client Platform), 502–503
- Schema Editor, 341–343
- SDK extension points, 332
  - GUI extension points, 334–338
  - Java classes, 334
  - schema-based, 333
  - templates, 338–341
- Slime UML, 547
- SpellChecker for Eclipse, 549
- SWT-Designer, 546
- Sysdeo Eclipse Tomcat Launcher, 548
- Systinet WASP Server for Java, 548
- Team Repository Providers, 547
- TimeStorm 2.0, 549
- Together Edition for Eclipse, 548
- Transcoder, 547
- W4Eclipse, 546
- wizards, 374
  - IWorkbenchWizard interface, 376
  - newWizard extension point, 375–376
  - WizardNewFileCreationPage class, 376–377
- workbench, 350–351
  - components, 351–352
  - editors, 355–362
  - event processing, 352–355
  - views, 362–366
  - workbench page, 351
  - workbench window, 351
- X-Men, 548
- XML Buddy, 548
- xored WebStudio, 547
- Pocket PC platforms (SWT), 202**
- PopupList class (SWT), 175**
- PreferenceConverter class (JFace), 231**
- PreferenceDialog class (JFace), 234**
- PreferenceManager class (JFace), 234**
- PreferenceNode class, 234**
- PreferencePage class (JFace), 232**
- preference page trees (JFace), 233–234**
  - PreferenceDialog class, 234
  - PreferenceManager class, 234
  - PreferenceNode class, 234

## preferences

- compiler, 14–15
- formatting code, 15–16
- importing and exporting, 10
- installed JREs, 12–13
- JFace, 230–231
  - field editors, 232–233
  - PreferenceConverter class, 231
  - PreferencePage class, 232
  - preference page trees, 233–234
  - PreferenceStore class, 231
- plug-ins, 377
- templates, 16–18
- VE (Visual Editor for Java), 46
- workbench, 11–12

**Preferences class, 319**

**Preferences dialog, 10–11**

**PreferenceStore class (JFace), 231**

**Previous Annotation function, 37**

**printer output (SWT), 196–198**

**Problems Filter, 20–21**

**Problems view, 19–21**

**processes, long-running**

- managing, 326
- monitoring, 320

**processing events, 352–353**

- component events, 353
- initialization, 354–355
- selection events, 354
- sequence, 355
- window events, 353

**productivity tools**

- automatic code completion
  - Code Assistant, 28–30, 77–78
  - comment out code, 31
  - delegate methods, 32
  - encapsulating fields, 32
  - generating constructors, 33
  - importing types, 31
  - inheriting constructors, 32
  - Javadoc comments, 31
  - overriding methods, 32
- Correction Assistant
  - Quick Assist, 34–35
  - QuickFix, 33–34
- help system, 26–27
- Javadoc View, 27–28
- Java Editor
  - code folding, 35–36
  - syntax coloring, 36
  - typing aids, 35
- Local History functions, 43–44
- Redo function, 42–43

- refactoring code, 38
  - constants, 42
  - factories, 40
  - methods, 39–40
  - modifying types, 38–39
  - types and classes, 40–41
  - variables, 41
- source code navigation, 36–37
- system information file, 25–26
- Undo function, 42–43

## products

- customizing, 394–396, 508
- deployment
  - Eclipse runtime environment, 394
  - example files, 396–398
  - features, 391, 391–392, 393
  - fragments, 390
  - installation formats, 388, 394
  - update sites, 398–401
  - zip files, 394
- internationalization, 401–402
  - Cheat Sheets, 404
  - help systems, 404
  - national language resource bundles, 405
  - text constants in manifest files, 403
  - text constants in programs, 402–403
- patches, 405

## Profiler, 548

## programming style

- agile programming, 537
- automated testing, 538
- executable prototypes, 538
- extreme programming, 537
- Java 1.5, 541–542
- making refinements, 539–540
- restructuring code, 540–541
- reusing components, 541

## ProgressBar class (SWT), 159

## ProgressMonitorWrapper class, 320

## ProjectionViewer class (JFace), 226

## projects, 322–323

- creating, 7
- natures, 322
- properties, 69–71

## properties

- beans, 48–49
- projects, 69–71
- Swing components, 49

## property pages, 377

## Property View, 366

## Pull up function, 39

**Q****Quick Assist, 34–35****QuickFix, 33–34****R****Rational, 548****RCP (Rich Client Platform), 501–502. See also Hex****Game project**

- deploying applications, 507–508
- global welcome screen, 508–509
- IDE comparison, 315
- IPlatformRunnable interface, 503
- plug-ins, 502–503
- product customization, 508
- testing applications, 507
- WorkbenchAdvisor class, 503–504
  - application hooks, 504–505
  - event loop hooks, 506
  - information providers, 506–507
  - welcome screen hook, 506
  - window hooks, 505–506

**Read Access function, 60****Redo function, 42–43****Refactor function, 60****refactoring code, 38, 540–541**

- constants, 42
- factories, 40
- methods, 39–40
- modifying types, 38–39
- types and classes, 40–41
- variables, 41

**References function, 60****remote debugging, 119****Remove All Terminated Launches function, 115****Rename function, 39****Replace With function, 60****repositories, 130–131****ResourceListSelectionDialog class, 374****resource management**

- Eclipse platform architecture, 315
- Eclipse UI forms, 349
- JFace
  - FontRegistry class, 205–206
  - ImageRegistry class, 206
  - JFaceColors class, 206
  - JFaceResources class, 206
- SWT, 200–201

**Resource Navigator, 51–52****ResourceNavigator class, 363–364****resources, 52, 320–322**

- comparing, 43
- containers, 322
- files, 323–324
- folders, 323

projects, 322–323

reacting to changes in, 325–326

replacing with older versions, 43

restoring, 43

storing, 52–53

synchronizing, 53

types, 52

workspace root, 322

**Restore from Local History function, 60****restoring resources, 43****Resume function, 115****Rich Client Platform. See RCP****rich clients, 501****RowLayout class (SWT), 179–180****Run Configurations, 9****Run icon, 9****S****sashes (SWT), 167–168****SashForm class (SWT), 175****SaveAsDialog class, 374****saving files (workbench editor), 357****scales (SWT), 156****schema-based extension points, 333****Schema Editor, 341–343****scrapbooks, 22–24****Scrollable class (SWT), 159****SDK extension points, 332**

GUI extension points, 334–338

Java classes, 334

schema-based, 333

templates, 338–341

**Search function, 60–62****Search View, 61–62****Section class, 347–348****Segmenter (FreeTTS), 74****selection events, processing, 354****Select Working Set function, 53****serializing (Jukebox project), 261–268****Shell class (SWT), 148–150****shortcuts, defining, 11–12****Show in Package Explorer function, 37****sliders**

Duke Speaks project, 93

SWT, 156

**Slime UML, 547****Sort function, 53****source code navigation, 36–37****SourceViewer class (JFace), 223****speech synthesizers. See also Duke Speaks project**

FreeTTS, 67

downloading, 67–68

importing, 68–69

Programmer's Guide, 75

FreeTTS speech synthesizer, 74–75

Java Speech API (JSAPI), 67

JSAPI (Java Speech API), 67

Via Voice, 73

### **SpellChecker for Eclipse, 549**

#### **spell checker project**

analyzing documents, 463

Check Spelling action, 424

CheckSpellingActionDelegate class, 431–438

disposal, 430–431

document management, 428–429

getInstance() method, 426–427

selection provider, 427–428

SpellCheckingTarget class, 425–426

text replacement, 429–430

configuring, 463

DefaultSpellCheckerPreferencePage class, 469

domain model, 464–465

GUI, 466

preferences, 463–464

reading from PreferenceStore, 471–472

ShortIntegerFieldEditor class, 470–471

SpellCheckerPreferencePage class, 466–469

core classes, 409

correction window

actions (JFace), 449–450

managing images, 450–452

SepellCorrectionView class, 439–449

deploying, 490

configuring Ant scripts, 492–495

feature, defining, 490–492

installation, 498

language feature, defining, 495–497

update site, defining, 497–498

engine, 408

engine/interface interaction, 452

creating engines, 459–460

managers (JFace), 453–454

managing engines, 458–459

operations, 462–463

plug-in selection, 454–457

processing spelling errors, 461–462

running the engine, 457–458

help system, 473

active help, 474–475

context-sensitive help, 473–474

running the help action, 476–477

table of contents, 473

internationalization, 484

language fragment, creating, 487–490

text constants in Java code, 484–486

text constants in manifest files, 487

Java properties plug-in, 477

help system, 483–484

JavaPropertiesPlugin class, 480–481

JavaPropertiesPreferencePage class, 482–483

Java-Properties tokenizer, 483–484

manifest, 478, 479–480

preferences, 481–482

setup, 477–478

tokenizer extension, 478–479

plug-in configuration, 412

imported files, 413–419

manifest, 413–419

setup, 410–411

SpellCheckerPlugin class, 419–421

dictionary URL, 421–422

initializing preferences, 422–423

manager, 423–424

third-party software, 555

#### **stacking windows, 64**

#### **StackLayout class (SWT), 184–185**

#### **Standard Widget Toolkit. See SWT**

#### **status line, accessing, 357**

#### **StatusLineManager (JFace), 228**

#### **StatusTextEditor class, 359**

#### **Step Into function, 115**

#### **Step Over function, 116**

#### **Step Return function, 116**

#### **Step with Filters function, 116**

#### **StructuredViewer class (JFace), 216–217**

#### **StyledText class (SWT), 175**

#### **SubProgressMonitor class, 320**

#### **Suspend function, 115**

#### **Swing components**

properties, 49

with SWT, 191

embedded contents, 192

events, 192–196

#### **SWT. See also Jukebox project**

accessibility options, 202–203

advantages, 140

data transfer, 198

clipboard, 198–199

drag and drop, 199

disadvantages, 140

events package, 141

adapters, 142

events, 142

listeners, 141–142

overview, 143–145

example program, 150–151

graphics package, 185

colors, 186–187

cursors, 189–190

fonts, 187–188

Graphics Context class, 185–186

images, 188–189

layouts package, 177–178

FillLayout class, 178–179

FormLayout class, 182–184

GridLayout class, 180–182

- RowLayout class, 179–180
- StackLayout class, 184–185
- package overview, 138–139
- Pocket PC platforms, 202
- printer output, 196–198
- resource management, 200–201
- run configuration, 151–152
- with Swing/AWT components, 191
  - embedded contents, 192
  - events, 192–196
- SWT package, 141
- widgets package, 145
  - browser widget, 177
  - buttons, 156–157
  - combos, 165–166
  - composites, 155
  - Control class, 146
  - custom widgets, 174–176
  - dialogs, 152–155
  - Display class, 147–148
  - groups, 156
  - labels, 161
  - lists, 164
  - menus, 170–174
  - Monitor class, 152
  - movable tool groups, 170
  - overview, 146
  - ProgressBar class, 159
  - sashes, 167–168
  - scales, 158–159
  - Scrollable class, 159
  - Shell class, 148–150
  - sliders, 158–159
  - tabbed folders, 168–169
  - tables, 161–164
  - text fields, 159–160
  - toolbars, 169–170
  - trees, 166–167
  - Widget class, 146
- Windows32 support (OLE), 201

**SWT-Designer, 546**

**Synchronize function, 133**

**synchronizing resources, 53**

**syntax coloring (Java Editor), 36**

**Sysdeo Eclipse Tomcat Launcher, 548**

**system information file, 25–26**

**Systinet WASP Server for Java, 548**

**T**

**tabbed folders (SWT), 168–169**

**TableEditor class (SWT), 176**

**tables (SWT), 161–164**

**TableTree class (SWT), 176**

**TableTreeEditor class (SWT), 176**

**TableWrapLayout layout manager, 346**

**talking Hello World example, 67–71**

**tasks, creating, 21–22**

**Tasks View, 366**

**team development, 129–130**

- comparison functions, 135
- conflict resolution, 133–134
- CVS (Concurrent Versions System), 129–130
- Export function, 135
- plug-ins, 316–317
- repositories, 130–131
- sharing projects, 132
- Team group preferences, 135
- version management, 133

**Team Repository Providers, 547**

**Team Synchronizing Perspective, 133**

**templates**

- automatic code completion, 29–30
- preferences, 16–18
- SDK extension points, 338–341

**Terminate function, 115**

**testing expressions, 117**

**test suites (JUnit)**

- creating, 122–124
- running, 124

**text, Duke Speaks project GUI, 94**

**Text class (SWT), 159–160**

**text constants, internationalizing, 402–403**

**TextContentAssistSubjectAdapter class (JFace), 226**

**TextEditor class, 360**

**text fields (SWT), 159–160**

**text formatters (JFace), 224**

**text markup, 348–349**

**text presentation (JFace), 225–226**

**text processing (JFace), 218**

- annotations, 223–224
- ComboContentAssistSubjectAdapter class, 226
- configuring, 223
- content assistants, 224–225
- document model, 218–219
- ProjectionViewer class, 226
- scrips, 219–220
- TextContentAssistSubjectAdapter class, 226
- text formatters, 224
- text presentation, 225–226
- TextViewer class, 220–223
  - hover infos, 222
  - operations, 222–223
  - selection, 221
  - SourceViewer class, 223
  - text presentation, 223
  - viewport, 221
  - visible text region, 221

**TextViewer class (JFace), 220–223**

- hover infos, 222
- operations, 222–223
- selection, 221

## TextViewer class (JFace) (continued)

---

- SourceViewer class, 223
- text presentation, 223
- viewport, 221
- visible text region, 221
- thin clients, 501**
- TimeStorm 2.0, 549**
- Together Edition for Eclipse, 548**
- Toggle Method Breakpoint function, 60**
- Toggle Watchpoint function, 60**
- Tokenizer (FreeTTS), 74**
- toolbars**
  - creating manually, 356
  - Jukebox project, 290–292
  - overview, 169–170
  - workbench views, 363
- Transcoder, 547**
- TreeEditor class (SWT), 176**
- trees (SWT), 166–167**
- tree structures, organizing elements into, 342–343**
- TreeViewer class (JFace), 217**
- troubleshooting. See also debugging**
  - Correction Assistant
    - Quick Assist, 34–35
    - QuickFix, 33–34
  - help system, 26–27
- Type Hierarchy Browser, 56–57**
- types**
  - modifying types, 38–39
  - moving, 38
  - refactoring code, 40–41
  - renaming, 39
- typing aids (Java Editor), 35**

## U

- Undo function, 42–43**
- Update Manager, 398–399**
- Update Manager Log, 26**
- update sites**
  - adding, 400
  - install handlers, 401
  - installing features, 400
  - managing configuration, 400–401
  - spell checker project, 497–498
  - Update Manager, 398–399
  - updating features, 400
- Up To function, 53**
- Use Supertype function, 41**
- Utterance Processor (FreeTTS), 75**

## V

- variables**
  - debugger, 116–117
  - refactoring code, 41
- Variables View, 116**
- VE (Visual Editor for Java), 45**

- beans, 48
  - generic, 48
  - properties, 48–49
- composing GUIs with, 46–48
- Duke Speaks project, 92
- event processing, 49–50
- installing, 45
- invoking, 46
- layouts, 49
- maximizing views, 46
- preferences, 46
- version 1.1 JREs, 13**
- version management, 133**
- Via Voice, 73**
- viewers (JFace), 214**
  - cell editors, 217–218
  - ContentViewer class, 216
  - data transfer, 218
  - event model, 215
  - hierarchy, 215–216
  - ILabelProvider class, 216
  - StructuredViewer class, 216–217
  - TreeViewer class, 217
- views**
  - Hex Game project, 515
  - maximizing, 65
  - minimizing, 65
  - workbench, 362–363
    - bookmark manager, 366
    - Outline View, 365–366
    - PageBookView class, 364–365
    - persistency, 363
    - Property View, 366
    - ResourceNavigator class, 363–364
    - Tasks View, 366
    - toolbars, 363
- Visual Editor for Java. See VE**

## W

- W4Eclipse, 546**
- warnings, 14–15. See also Problems view**
- Widget class (SWT), 146**
- widgets package (SWT), 145**
  - browser widget, 177
  - buttons, 156–157
  - combos, 165–166
  - composites, 155
  - Control class, 146
  - custom widgets, 174–176
  - dialogs, 152–153
    - message boxes, 154–155
    - predefined, 153–154
  - Display class, 147–148
  - groups, 156
  - labels, 161
  - lists, 164

- menus, 170–174
  - Monitor class, 152
  - movable tool groups, 170
  - overview, 146
  - ProgressBar class, 159
  - sashes, 167–168
  - scales, 158–159
  - Scrollable class, 159
  - Shell class, 148–150
  - sliders, 158–159
  - tabbed folders, 168–169
  - tables, 161–164
  - text fields, 159–160
  - toolbars, 169–170
  - trees, 166–167
  - Widget class, 146
  - window events, processing, 353**
  - windows**
    - closing, 65
    - docking, 63
    - FastView, 64
    - JFace, 206–207
    - managing (Jukebox project), 258–260
    - maximizing, 65
    - minimizing, 65
    - opening, 65
    - stacking, 64
    - workbench, 351
  - Windows32, SWT support, 201**
  - Wizard class (JFace), 228–229**
  - WizardDialog class (JFace), 230**
  - WizardNewFileCreationPage class, 376–377**
  - WizardPage class (JFace), 229–230**
  - wizards**
    - JFace, 228
      - Wizard class, 228–229
      - WizardDialog class, 230
      - WizardPage class, 229–230
      - WizardSelectionPage class, 230
    - plug-ins, 374
      - IWorkbenchWizard interface, 376
      - newWizard extension point, 375–376
      - WizardNewFileCreationPage class, 376–377
  - WizardSelectionPage class (JFace), 230**
  - workbench, 51–52, 350–351**
    - components, 351–352
    - editors, 355–356
      - AbstractTextEditor class, 357–359
      - FormEditor class, 360–361
      - keyboard shortcuts, 357
      - markers, 361–362
      - menus, 356
      - MultiEditor class, 360
      - saving files, 357
      - status line, 357
      - StatusTextEditor class, 359
      - TextEditor class, 360
      - toolbars, 356
    - event processing, 352–353
      - component events, 353
      - initialization, 354–355
      - selection events, 354
      - sequence, 355
      - window events, 353
    - views, 362–363
      - bookmark manager, 366
      - Outline View, 365–366
      - PageBookView class, 364–365
      - persistency, 363
      - Property View, 366
      - ResourceNavigator class, 363–364
      - Tasks View, 366
      - toolbars, 363
    - wizards, 374
      - IWorkbenchWizard interface, 376
      - newWizard extension point, 375–376
      - WizardNewFileCreationPage class, 376–377
    - workbench page, 351
    - workbench window, 351
  - WorkbenchAdvisor class, 503–504**
    - application hooks, 503–504
    - event loop hooks, 506
    - information providers, 506–507
    - welcome screen hook, 506
    - window hooks, 503–504
  - workbench preferences, 11–12**
  - workspace, 320**
    - backing up, 2
    - long-running processes, managing, 326
    - markers, 324–325
    - resources, 320–322
      - containers, 322
      - files, 323–324
      - folders, 323
      - projects, 322–323
      - reacting to changes in, 325–326
      - workspace root, 322
      - selecting location, 2
  - Write Access function, 60**
- ## X
- X-Men, 548**
  - XML Buddy, 548**
  - xored WebStudio, 547**

