

Index

Symbols

/**, 73
 @ symbol, 73
 >> symbol, 21

A

Add JARs option, 89
 addItem() method, 65
 agile development, 14
 team ownership, 225-226
 Agile Manifesto, 130-131
 agile software development, 129, 133
 Agile Manifesto, 130-131
 coding standards, 137
 collective ownership, 136
 continuous integration, 136
 continuous testing, 135, 155
 iterative development, 132
 refactoring, 135
 self-adaptive process, 134
 Ant, 200
 automating Checkstyle, 276-277
 build file elements, 200
 init target, 217
 build properties, 201
 build tasks, 203-205
 building and testing complete projects,
 217-221
 building the UBL project, 206
 internal targets, 212
 overview, 200

 running in Eclipse, 209
 Ant view and console output, 212
 automatic project builder, 213, 216
 runtime configuration, 210
 Ant editor, 207
 Ant view, 212
 Apache Axis distribution, 93
 Apache open source project, 200
 applications
 package dependencies, 40, 43
 running, 29
 ASCII vs. binary, 232
 auditing compliance, 269
 Checkstyle plug-in, 272
 automating with Ant, 276-277
 problems displayed, 274
 running rules, 273
 automatic compilation, 83
 automatic incremental builds, 198
 automatic project builder (Ant), 213, 216
 automergable conflicting change, 250
 Axis, 162

B

bin folder, 234
 binary vs. ASCII, 232
 BIRT (Business Intelligence and Reporting
 Tools), 14
 bookmarks, 143
 Bookmarks view, 49
 branches (project versions), 253

- breakpoints, 108
- Build Automatically option, 198
- build files
 - Ant, 200
 - building and testing complete projects, 217
- build machines, 197
- build paths, 84
 - configuring source folders, 85
 - dependencies, 86
 - libraries, 88-90
 - order and export, 92
- build tasks, 203-205
- builders, 85
- builds, 18
- Business Intelligence and Reporting Tools (BIRT), 14

- C**
- C/C++ Development Tools (CDT), 8
- capabilities, selecting, 48
- Catalog.java file, 27
- CDT (C/C++ Development Tools), 8
- Checkstyle, 272, 278
 - automating with Ant, 276-277
 - problems displayed, 274
 - running rules, 273
- class references (refactoring), 189
- classname attribute, 205
- classpath argument, 119
- classpath variables, 90
- code
 - breakpoints, 108
 - compiler warnings and errors, 268
 - debugging, 105
 - displaying code snippets, 112
 - executing a code snippet, 110
 - managing debug sessions, 115, 118
 - readability, 260
 - refactoring, 135, 177
 - composing methods, 190
 - extracting interfaces, 183
 - generalization, 193
 - introducing a factory pattern, 186-188
 - making method calls simpler, 192
 - moving features between objects, 191
 - operations, 179
 - organizing classes and packages, 194
 - organizing data, 192
 - preparing for, 179
 - renaming packages and classes, 180
 - searching for class references, 189
 - verifying and committing results, 190
 - when to refactor, 178
 - spell checker, 266-268
 - stepping through execution, 108
 - styles and templates, 263-265
 - test-first strategy, 167
 - white space, 261
- Code Conventions for the Java Programming Language*, 259
- Code Formatter, 261-263
- code standards, auditing compliance, 269, 272-277
- code templates, customizing, 25
- CodePro Advisor, 278
- coding conventions, 259
- coding standards
 - Code Formatter, 261-263
 - code style and templates, 263-265
 - compiler warnings and errors, 268
 - spell checker, 266-268
- Combined Hover, 66
- commands
 - refactoring, 180
 - composing methods, 190
 - generalization, 193
 - making method calls simpler, 192
 - moving features between objects, 191
 - organizing classes and packages, 194
 - organizing data, 192
 - stepping through execution, 108
 - synchronizing workflow with repositories (CVS), 249
- comments, spell checking, 266-268
- Concurrent Versions System. *See* CVS

- configuring
 - Eclipse, feature installation, 141
 - shared user libraries, 93
 - linked library project, 95
 - source folders in Java project build paths, 85
 - spell checker with a user-defined dictionary, 266
 - Console view, 30
 - Content Assist, 67-68
 - activating, 208
 - completing a class name, 69
 - insert types and variable names, 68
 - Javadoc Content Assist, 73
 - method name completion, 70
 - source code templates, 71-72
 - continuous integration, 136, 197
 - continuous testing, 135, 155
 - creating
 - code templates for logging, 99
 - Java projects, 21-22
 - customizing code templates, 25
 - New Java Class wizard, 24
 - opening the Java perspective, 21
 - Package Explorer, 23
 - running your application, 29
 - saving changes, 28
 - writing methods, 27
 - patches (CVS), 255
 - reusable library of components, 62
 - shared user libraries, 93
 - linked library project, 95
 - workspaces, 35
 - Ctrl+1 shortcut, 74
 - Ctrl+F3 shortcut, 65
 - Ctrl+O shortcut, 65
 - Ctrl+Shift+H shortcut, 80
 - Ctrl+Shift+M shortcut, 76
 - Ctrl+Shift+O shortcut, 76
 - Ctrl+Shift+T shortcut, 63
 - Ctrl+Space shortcut, 68
 - Current Instruction Pointer, 108
 - customizing code templates (Java projects), 25
 - CVS (Concurrent Versions System), 225, 257
 - adding repository locations in the CVS perspective, 239
 - checking out modules not created by Eclipse, 241
 - decoration icons, 245
 - ignoring resources, 235
 - managing project versions, 252
 - branching and merging, 254
 - checking out a version, 253
 - tagging files, 253
 - patches, 255
 - project storage, 231
 - selecting projects/versions to check out, 242
 - sharing projects, 230-234
 - CVS metadata files, 237
 - team project sets, 237
 - synchronization, 244-248
 - keyword substitution, 252
 - non-automergable conflicts, 250
 - team repositories, 226
 - Team Synchronizing perspective, 246-248
 - tracking project revisions, 239
 - workflow, 229
 - CVS servers, 226
 - CVSNT server, 227
 - Cygwin tools, 267
- ## D
- DbUnit Web site, 174
 - Debug Call Stack, 108
 - Debug Perspective, 104
 - debugging
 - creating test suites, 173
 - customizing test configuration, 172
 - displaying code snippets, 112
 - executing a code snippet, 110
 - Java, 117, 120
 - remote Tomcat application Servlet, 121
 - managing debug sessions, 115, 118
 - remote applications, 120
 - running tests, 168-169
 - starting a debug session, 104
 - stepping through execution, 108
 - test strategy, 156
 - unsuccessful tests, 170
 - writing test cases, 161-167
 - declarative syntax, 200

- dependencies, 40, 43
 - Java project build paths, 86
 - Display view, 112
 - displaying code snippets during
 - debugging, 112
 - downloading
 - Eclipse SDK, 17
 - JavaToHtml converter, 148
- E**
- echo task, 204
 - Eclipse, 3
 - agile development, 14, 133
 - coding standards, 137
 - collective ownership, 136
 - continuous integration, 136
 - continuous testing, 135
 - refactoring, 135
 - self-adaptive process, 134
 - components, 12
 - Console view, 30
 - CVSNT server, 227
 - features, 140
 - installing, 141, 146
 - license agreements, 145
 - searching for, 142
 - updating, 151-152
 - installing plug-ins without features, 148-149
 - JUnit testing
 - creating test suites, 173
 - customizing test configuration, 172
 - running tests, 168-169
 - unsuccessful tests, 170
 - writing test cases, 161-167
 - plug-ins, 10
 - project guidelines, 259
 - RCP, 11
 - refactoring, 179
 - composing methods, 190
 - extracting interfaces, 183
 - generalization, 193
 - introducing a factory pattern, 186-188
 - making method calls simpler, 192
 - moving features between
 - objects, 191
 - organizing classes and
 - packages, 194
 - organizing data, 192
 - preparing for, 179
 - renaming packages and classes, 180
 - searching for class references, 189
 - verifying and committing
 - results, 190
 - running Ant, 209
 - Ant view and console output, 212
 - automatic project builder, 213, 216
 - runtime configuration, 210
 - runtime platform, 5
 - workbench, 45
 - individual preferences, 58
 - perspectives, 46-48
 - preferences, 57-58
 - resource editors, 55
 - team preferences, 59
 - views, 49-50, 53-54
 - Eclipse IDE, 4
 - features, 6
 - functional building blocks, 4
 - Eclipse Modeling Framework (EMF), 12
 - Eclipse Plug-in Central, 154
 - Eclipse SDK, downloading, 17
 - eclipse.exe file, 18
 - editors (Java editor), 62
 - embedded editors, 55
 - EMF (Eclipse Modeling Framework), 12
 - Enable HTTP proxy connection option, 150
 - Enclosing method, 191
 - errors (JUnit tests), 170
 - exception testing, 165
 - execution context, 110
 - exporting workbench preferences, 58
 - Expressions view, 113
 - external editors, 55
 - External Tools menu, 199
 - Extract Class refactoring, 191
 - Extract method, 191
 - extracting interfaces from UBL class
 - implementations, 180
 - Extreme Programming (XP), 134

F

F3 shortcut, 63
 features, 140
 installing, 141, 146
 installing plug-ins without features,
 148-149
 license agreements, 145
 searching for, 142
 updating, 151-152
 files
 linked, 38
 filesets, 203
 fixtures (test), 165
 folders
 linked, 38
 .metadata folder, 34
 for loop template, 72

G-H

GEF (Graphical Editor Framework), 13
 Generate method comment option, 77
 generating getters and setters, 77
 getters, 77
 Graphical Editor Framework (GEF), 13
 HEAD, 254
 hierarchies, 78
 Hierarchy view, 80
 hot code replacement, 117
 HttpUnit Web site, 174

I

IDE (integrated development environment), 4
 importing
 projects into workspace (CVS), 244
 workbench preferences, 58
 individual preferences, 58
 init target, 217
 Inline method, 191
 installing
 CVSNT server, 227
 Eclipse, 18
 feature updates, 151-152
 features, 141, 146
 product location, 145
 plug-ins without features, 148-149

integrated development environment, 4
 features, 6
 functional building blocks, 4
 integration builds, 18
 internal targets, 212
 iterative development, 132

J

Jakarta Cactus standards, 265
 Java
 build paths, 84
 configuring source folders, 85
 dependencies, 86
 libraries, 88-0
 order and export, 92
 code formatter, 261-63
 code style and templates, 263-65
 compiler warnings and errors, 268
 creating a new project, 21-22
 customizing code templates, 25
 New Java Class wizard, 24
 opening the Java perspective, 21
 Package Explorer, 23
 running your application, 29
 saving changes, 28
 writing methods, 27
 debug preferences, 117
 hierarchy in code, 78
 specifying which version will run
 Eclipse, 19
 spell checker, 266-268
 task tags, 52
 Java Architecture for XML Binding
 (JAXB), 180
 Java Build Path, 68
 Java compiler settings, 97
 Java debugger, 120
 remote Tomcat application Servlet, 121
 Java Development Tools (JDT), 7, 140
 Java editor, 62
 Hover tab, 66
 Java runtime environment (JRE), 172
 Java SDK, 64
 java task, 204
 Javadoc build task, 203
 Javadoc comments, 73, 259, 264

Javadoc tool, customizing output, 221
 JavaToHtml converter, 148
 JBoss project, 143
 JDT (Java Development Tools), 7, 140
 JRE (Java runtime environment), 172
 JUnit, 155

- creating test suites, 173
- customizing test configuration, 172
- running tests, 134-135, 168-169
- test strategy, 156
- unsuccessful tests, 170
- writing test cases, 161-167

 junit task, 204
 Junit Test Case wizard, 163
 JUnit view, 169
 junitreport task, 205
 JVMs, 19

K - L

keyword substitution, 252
 label decorations, 245
 libraries

- Java project build paths, 88-90
- order and export, 92
- shared user libraries, 93
 - linked library project, 95

 license agreements (features), 145
 linked folders, variables, 39
 linked resource locations, 38
 logging, creating code templates, 99

M

managing

- debug sessions, 115, 118
- project versions (CVS), 252
 - branching and merging, 254
 - checking out a version, 253
 - tagging files, 253

 marker views, 50, 53
 metadata files (CVS), 237
 .metadata folder, 34
 MIDP (Mobile Information Device Profile), 86
 mkdir task, 205
 Mobile Information Device Profile (MIDP), 86

modifying code templates, 263
 Move Field refactoring, 191
 Move Method refactoring, 191

N-O

naming conventions, 259
 Navigator view, 37, 49
 New Java Class wizard, 24
 nightly builds, 18
 non-automergable conflicting change, 250
 non-automergable conflicts, 250
 non-conflicting changes, 250
 OASIS Universal Business Language (UBL), 41
 OASIS Web site, 157
 optimistic model (CVS), 229
 order and export libraries, Java project build paths, 92
 Outline view, 49, 62, 66, 207
 output folders, 22

P

Package Explorer, 23
 patches (CVS), 255
 path variables, 38
 PDE (Plug-in Development Environment), 9, 140
 perspectives, 20, 46

- Java perspective, 21
- perspective bar, 46
- saving specialized perspectives, 47
- selecting capabilities, 48

 Platform, 140
 platform runtime download configuration (Eclipse SDK), 17
 Plug-in Developer Environment (PDE), 9, 140
 plug-ins

- Checkstyle, 272
 - automating with Ant, 276-277
 - problems displayed, 274
 - running rules, 273
- grouping into features, 140
- installing without features, 148-149

 PMD, 143, 223, 278
 policy file, updating, 152

- portability
 - classpath variables, 90
 - features, 140
 - preferences (workbench)
 - individual preferences, 58
 - keyboard shortcuts, 58
 - scope, 57
 - team preferences, 59
 - Pretend to be a Unix CVS version option, 227
 - Problems view, 49
 - product location, 145
 - programming
 - agile software development, 129, 133
 - Agile Manifesto, 130-131
 - coding standards, 137
 - collective ownership, 136
 - continuous integration, 136
 - continuous testing, 135
 - iterative development, 132
 - refactoring, 135
 - self-adaptive process, 134
 - test-first strategy, 167
 - Project menu, build options, 199
 - projects
 - Ant, 200
 - building and testing complete projects, 217-221
 - Build Automatically option, 198
 - builders, 85
 - continuous integration, 198
 - CVS
 - managing versions, 252-254
 - patches, 255
 - dependencies, 40, 43
 - Eclipse, guidelines, 259
 - natures, 36
 - resources, 36
 - sharing (CVS), 230-234
 - CVS metadata files, 237
 - team project sets, 237
 - storage (CVS), 231
 - workspaces, 33
 - resources, 35
 - using multiple workspaces, 34
 - properties (Ant build files), 201
 - Properties vies, 49
 - property task, 201
 - Proxy settings, 150
- Q-R**
- qualified names, 22
 - Quick Assist completion lists, 87
 - Quick Fix, 74-76
 - Rational Unified Process (RUP), 129
 - RCP (Rich Client Platform), 3, 11
 - readability, 260
 - refactoring, 135, 177
 - composing methods, 190
 - extracting interfaces, 183
 - generalization, 193
 - introducing a factory pattern, 186-188
 - making method calls simpler, 192
 - moving features between objects, 191
 - operations, 179
 - organizing
 - classes and packages, 194
 - data, 192
 - preparing for, 179
 - previewing changes, 184
 - renaming packages and classes, 180
 - searching for class references, 189
 - verifying and committing results, 190
 - when to refactor, 178
 - references to undefined properties (Ant build files), 202
 - refreshing resources, 35
 - releases, 17
 - remote application debugging (remote Tomcat Servlets), 121, 124
 - repositories, 226
 - alternate views, 239
 - source control, 244-247
 - synchronization, 229
 - team, 83
 - resource editors, 55
 - Resource perspective, 20
 - resources
 - linked resource locations, 38
 - projects, 36
 - workspace, 35
 - Resume command, 108
 - Rich Client Platform (RCP), 3, 11
 - Rule of Three, 178
 - Run to Line command, 109
 - RUP (Rational Unified Process), 129

S

saving
 changes to Java projects, 28
 specialized perspectives, 47
 scope (workbench preferences), 57
 SCOWL dictionary word list, 267
 SDK (software development kit), 64
 SDO (Service Data Object), 12
 Search view, 49
 searching for feature updates, 142, 151-152
 Service Data Object (SDO), 12
 setUp() method, 163
 shared user libraries, 93-95
 shortcuts
 Ctrl+1, 74
 Ctrl+O, 65
 Ctrl+Shift+H, 80
 Ctrl+Shift+M, 76
 Ctrl+Shift+O, 76
 Ctrl+Space, 68
 -showlocation option, 19
 source code
 readability, 260
 templates, 71-72
 source control repositories, 244-247
 source folders, 22
 configuring in Java project build
 paths, 85
 src folder, 219
 stable builds, 18
 stack frame, execution context, 110
 starting debugging sessions, 104
 Step Into command, 109
 Step Over command, 109
 Step Return command, 109
 stepping through programs, 108
 storing projects (CVS), 231
 subtype hierarchy, 79
 supertype hierarchy, 79
 Suspend command, 108
 synchronization, 244-247
 keyword substitution, 252
 non-automergable conflicts, 250
 Team Synchronizing perspective, 248

synchronizing work with repositories
 (CVS), 229
 sysout template, 73
 System Editor, 56

T

tagging files (CVS), 253
 targets (Ant projects), 200
 Task markers, 52
 task tags, 52
 taskdef task, 205
 tasks
 Ant library, 203-205
 building the UBL project, 206
 Ant projects, 200
 Tasks view, 49
 team development
 CVS
 project storage, 231
 sharing projects, 230-234, 237
 workflow, 229
 CVS servers, 226
 team ownership, 225-226
 team preferences, 59
 team project sets (CVS), 237
 team repositories, 83, 226
 Team Synchronizing perspective, 47, 246-248
 tear-off views, 54
 templates, 71
 Terminate command, 108
 test target, 220
 testing
 before committing changes to the
 repository (CVS), 229
 creating test suites, 173
 customizing test configuration, 172
 fixtures, 165
 running tests, 168-169
 strategy, 156
 unsuccessful, 170
 writing test cases, 161-167
 Text Editor, 56
 Tomcat, debugging a remote Tomcat Servlet,
 121, 124
 troubleshooting (Quick Fix), 74-76
 type hierarchy, 78

U

UBL (OASIS Universal Business Language),
41, 157-158
 projects
 adding and configuring a custom
 builder, 215
 building, 206
 writing test cases, 161
UML2 (Unified Modeling Language 2.0), 13
undefined properties, references to, 202
Unified Modeling Language 2.0 (UML2), 13
Universal Business Language. *See* UBL
updating features, 151-152
upper camel case form, 69
Use Step Filters command, 109

V

variables
 classpath, 90
 linked folders, 39
VE (Visual Editor), 13
views
 tear-off views, 54
 workbench, 49-50
 marker views, 50, 53
 view layout, 54
Visual Editor (VE), 13
-vm option, 19

W

Web services, 162
Web sites
 Agile Manifesto, 130
 Ant editor, 136
 CVS, 226
 CVSNT, 227
 Cygwin, 267
 DbUnit, 174
 Eclipse, 3
 Eclipse Community Projects, 257
 Eclipse Distilled companion site, 165
 Eclipse Plug-in Central, 154
 Eclipse projects, 12
 HttpUnit, 174

 Java, 64
 JBoss project, 143
 JUnit testing, 134
 OASIS, 157
 PMD project, 143
Web Tools Platform project (WTP project), 10
Welcome page, 19
white space, 261
Windows, eclipse.exe file, 18
workbench, 45
 adding a dictionary to, 267
 individual preferences, 58
 perspectives, 46
 perspective bar, 46
 saving specialized perspectives, 47
 selecting capabilities, 48
 preferences
 keyboard shortcuts, 58
 scope, 57
 resource editors, 55
 setting update preferences, 150
 team preferences, 59
 views, 49-50
 marker views, 50, 53
 Outline view, 62, 66
 view layout, 54
workflow
 synchronization, 244-247
 keyword substitution, 252
 non-automergable conflicts, 250
 Team Synchronizing
 perspective, 248
workspace launcher dialog, 33
workspaces, 33
 importing projects into (CVS), 244
 resources, 35
 using multiple workspaces, 34
writing tests, 155
 creating test suites, 173
 customizing test configuration, 172
 running your test, 168-169
 test cases, 161-167
 test strategy, 156
 unsuccessful tests, 170
WTP (Web Tools Platform) project, 10
 J2EE Standard Tools subproject, 11
 Web Standard Tools subproject, 10

X-Z

- XML Schema Infoset (XSD), 13
- XP (Extreme Programming), 134
- XSD (XML Schema Infoset), 13

