

IntelliJ IDEA

IN ACTION

Duane K. Fields
Stephen Saunders
Eugene Belyaev



 MANNING

contents

preface xix
acknowledgments xxi
about this book xxiii

1

Getting started with IDEA 1

- 1.1 Installing and running IDEA 2
 - Downloading the latest version* 2
 - *Installing IDEA* 4
 - Running IDEA for the first time* 6
- 1.2 Reviewing the IDEA interface 7
 - The main IDEA workspace* 7
 - *The main menu bar* 8
 - The IDEA toolbar* 8
 - *The source code editor* 9
 - The tool windows* 9
- 1.3 Implementing “Hello, World” 10
 - Creating a project* 10
 - *Making a Java class* 15
 - Building the project* 17
 - *Running the project* 17
- 1.4 The plot thickens 19
 - Welcome to ACME Incorporated* 19
 - Starting the ACME project* 19
- 1.5 Summary 20

- ## 2 **Introducing the IDEA editor** 22
- 2.1 Exploring the IDEA interface 23
 - Hanging out in the gutter* 24
 - *Using the marker bar* 24
 - Using the editor tabs* 25
 - *Exploring the status bar* 30
 - Exploring the tool windows* 32
 - 2.2 Using the IDEA editor 40
 - Writing the first ACME classes and interfaces* 40
 - *Opening files into the editor* 42
 - *Saving your work* 45
 - Printing your file* 47
 - *Navigating in the editor* 48
 - Making text selections* 50
 - *Using IDEA's undo and redo mechanism* 52
 - *Cutting, copying, and pasting* 53
 - Searching for and replacing text* 55
 - 2.3 Summary 61
- ## 3 **Using the IDEA editor** 63
- 3.1 What makes IDEA the intelligent editor? 64
 - 3.2 Using code folding 64
 - 3.3 Navigating through your Java code 66
 - Navigating between methods* 67
 - *Navigating to a symbol's declaration* 67
 - *Navigating to a symbol by name* 69
 - Navigating with the structure view pop-up* 70
 - Navigating to an overridden/implemented or overriding/implementing method* 71
 - 3.4 Analyzing your Java code in real-time 72
 - How IDEA alerts you to problems in your code* 73
 - *Monitoring the status of the current document* 73
 - *Navigating between problems in the current file* 74
 - *Controlling the reparse delay* 74
 - *Configuring IDEA's warning levels* 74
 - 3.5 Getting help from the JavaDoc and API references 76
 - Viewing method parameters* 76
 - *Viewing the JavaDoc* 77
 - Creating JavaDoc comments* 78
 - *Generating your project's JavaDoc reference* 79
 - 3.6 Code completion 80
 - Using IDEA's code-completion features to do your work for you* 80
 - Completing brackets, braces, parentheses, and quotation marks with smart completion* 85
 - *Commenting out code* 85
 - Reformatting code* 86
 - *Customizing IDEA's code completion settings* 87

- 3.7 Using IDEA's code-generation tools 89
 - Generating constructors* 89
 - *Generating accessor and mutator methods* 90
 - *Generating hashCode and equals methods* 91
 - Overriding methods in your superclass* 94
 - *Implementing methods of an interface* 94
 - *Creating delegation methods* 95
 - Enclosing a block of code* 96
 - *Customizing code generated by IDEA* 96
- 3.8 Programming by intention 97
 - What are intention actions?* 97
 - *Why and when IDEA suggests intention actions* 98
 - *Using intention actions to fix errors* 98
 - Using intention actions for code completion* 98
 - Choosing an intention action to execute* 99
 - Disabling intention alerts* 100
 - *Exploring some common intention actions* 100
- 3.9 Continuing the ACME project 103
- 3.10 Summary 105

4 **Managing projects 107**

- 4.1 Understanding IDEA's project strategy 108
 - Examining the IDEA project hierarchy* 108
 - *Selecting different types of modules* 109
 - *Selecting a project structure* 110
- 4.2 Working with projects 111
 - Creating a new project* 111
 - *Managing project settings* 113
 - Working with project files* 118
- 4.3 Working with modules 119
 - Managing project modules* 119
 - *Creating a Java module with the module wizard* 121
 - *Managing Java module settings* 124
- 4.4 Working with libraries 128
 - Understanding library basics* 128
 - *Adding libraries to the project* 130
 - *Migrating projects from IDEA 3.x* 132
 - Sharing projects with others* 133
 - *Using path variables* 133
- 4.5 Using the Project tool window 135
 - Understanding the Project and Packages views* 135
 - Configuring the Project window* 136
- 4.6 Summary 141

- ## 5 ***Building and running applications*** 142
- 5.1 Building a project 143
 - How IDEA builds your project* 143
 - *Setting up a compiler* 145
 - *Building an application under IDEA* 149
 - Reviewing the results of the build* 150
 - 5.2 Extending IDEA's build system with Ant 151
 - Introducing Ant* 152
 - *Improving your build process with Ant* 155
 - *Working with Ant build files* 156
 - Executing Ant targets* 160
 - *Following Ant's progress in the Messages window* 162
 - *Controlling Ant's behavior* 165
 - 5.3 Running your project 171
 - Managing Run/Debug configuration profiles* 172
 - *Executing a Run configuration* 177
 - *Using the Run window console* 179
 - 5.4 Expanding the ACME project 181
 - Adding a library to the ACME project* 181
 - Improving and running the ACME project* 182
 - 5.5 Summary 184
- ## 6 ***Debugging applications*** 185
- 6.1 Introducing the debugging process 186
 - Finding and fixing bugs with a debugger* 186
 - *Preparing your code for debugging* 188
 - *Debugging your source code* 190
 - 6.2 Working with breakpoints 191
 - Managing breakpoints* 192
 - *Working with line number breakpoints* 195
 - *Working with method breakpoints* 196
 - Working with exception breakpoints* 196
 - *Working with field watchpoints* 197
 - *Setting conditional breakpoints* 199
 - Configuring breakpoint actions* 201
 - 6.3 Debugging an application 202
 - Executing an application under the debugger* 202
 - Debugging an application on a remote server* 205
 - Stepping through the program* 208
 - Working with threads* 211
 - 6.4 Viewing runtime data in the debugger 214
 - Understanding the Java call stack* 214
 - *Inspecting a stack frame* 216
 - *Working with watches* 221
 - Altering your program while debugging* 222

- 6.5 Configuring the debugger 223
 - Managing display preferences* 223
 - *Limiting the scope of debugging* 224
 - *Customizing the data display view* 225
 - Improving the speed of the debugger* 227
- 6.6 Improving the quality of the ACME project 229
- 6.7 Summary 230

7 **Testing applications with JUnit** 231

- 7.1 Testing applications with JUnit 232
 - Understanding the JUnit philosophy* 232
 - Exploring the JUnit API* 233
- 7.2 Adding test cases to your project 236
 - Creating a test case from a file template* 237
 - Adding the JUnit library to your Classpath* 237
- 7.3 Running test cases in IDEA 238
 - Creating a Run/Debug configuration for your test* 238
 - Running your unit test configuration* 241
- 7.4 Working with IDEA's JUnit test runner 242
 - Exploring the JUnit tool window* 243
 - *Monitoring testing progress* 245
 - *Managing the testing session* 246
 - Analyzing test results* 246
- 7.5 Improving the quality of the ACME project 252
- 7.6 Summary 253

8 **Using version control** 254

- 8.1 Configuring your project for version control 255
 - Understanding version control basics* 255
 - Enabling version control support in IDEA* 257
 - Configuring IDEA to use CVS* 259
 - Configuring other types of version control systems* 264
- 8.2 Working with files under CVS 267
 - Retrieving files from the repository* 267
 - *Working with files under CVS control* 271
 - *Committing your changes* 276
 - Working with branches and tags* 279
 - Viewing change history* 281

- 8.3 Using IDEA's Local History 289
 - Understanding IDEA's Local History* 289
 - *How IDEA tracks your changes* 290
 - *Exploring your Local History* 291
- 8.4 Summary 293

9 **Analyzing and refactoring applications 295**

- 9.1 Analyzing your code 296
 - Tracking down dependencies* 296
 - *Exploring code with the Structure view* 298
 - *Exploring the code hierarchy* 300
- 9.2 Locating potential problems with the code inspector 303
 - Launching the inspector* 304
 - *Specifying inspections to perform* 306
 - *Viewing the inspection results* 308
- 9.3 Other advanced code analysis features 309
 - Using Structural Search and Replace* 309
 - *Analyzing dependencies* 314
 - *Locating duplicate code* 316
- 9.4 Improving code design through refactoring 317
 - Performing a refactoring* 317
 - *Renaming code symbols to improve readability* 320
 - *Refactoring to improve class or package organization* 321
 - *Working with fields, variables, and constants* 325
 - *Refactoring method calls to improve usability* 328
 - *Restructuring class hierarchies* 333
 - *Migrating source code to new package or class names* 338
- 9.5 Summary 339

10 **Developing Swing applications 341**

- 10.1 Understanding the GUI Designer 342
 - The GUI-building process* 342
 - Working with the user interface* 343
- 10.2 Working with forms 345
 - Creating a new form* 345
 - *Placing components into the workspace* 345
 - *Setting component properties* 347
 - Laying out a form* 348
- 10.3 Designing an ACME GUI 351
 - Creating a new GUI form in IDEA* 351
 - *Manually creating the basic layout* 351
 - *Applying the grid layout* 352
 - Setting component properties* 352
 - Touching up the final interface* 352

- 10.4 Understanding properties 353
 - Spanning rows and columns* 354
 - *Setting an anchor point (cell alignment)* 354
 - *Using spacers to control layout* 355
 - Controlling the grid spacing* 356
 - *Setting container margins* 356
 - *Setting sizing policies* 356
 - Setting fill policies* 357
 - *Adding borders* 357
- 10.5 Adding functionality to forms 357
 - Binding forms and their components* 358
 - *Creating a constructor* 359
 - *Generating getter/setter methods for fields bound to data* 361
 - *Invoking your form* 361
- 10.6 Adding functionality to the ACME GUI 362
 - Binding the ACME GUI to a form class* 362
 - *Creating a constructor* 363
 - *Implementing the Convert functionality* 364
 - Providing an entry point* 365
- 10.7 Building and running your form 366
 - Generating the GUI to binary or source* 367
 - *Including the forms library* 367
 - *Compiling with Ant* 368
- 10.8 Summary 369

11 **Developing J2EE applications** 370

- 11.1 Working with web applications 371
 - Understanding web modules* 371
 - *Creating a new web module* 372
 - *Configuring a web module* 373
 - Using the J2EE tab of the Project tool window* 377
 - Working with servlets and filters* 379
 - Working with JavaServer Pages* 380
 - Implementing an ACME web application* 386
- 11.2 Working with EJBs 390
 - Working with EJB modules* 391
 - Working with J2EE application modules* 399
- 11.3 Understanding application server integration 402
 - Integrating IDEA with Tomcat* 403
 - Integrating IDEA with WebLogic* 404
 - Integrating IDEA with generic application servers* 404
 - Installing Tomcat to support the ACME web application* 405

- 11.4 Running and debugging web applications 406
 - Running your web application* 408
 - Running the ACME web application on Tomcat* 408
 - Debugging your web application* 410
- 11.5 Working with web content (IDEA 5.0 and higher) 412
 - Supported file types and content* 412
 - *Basic editor features* 415
 - *Coding assistance for web content* 417
 - Navigating through web content files* 421
 - Refactoring web content* 422
- 11.6 Summary 423

12

Customizing IDEA 425

- 12.1 Configuring IDEA's options and settings 426
 - Customizing the interface* 429
- 12.2 Customizing your code style 430
 - How IDEA uses your code style* 431
 - *Variable naming and general formatting* 432
 - *Controlling indentation and braces* 434
 - *Controlling whitespace* 436
 - Customizing import statements* 437
 - *Controlling line wraps* 439
 - *Adherence to a code standard—made easy* 439
- 12.3 Customizing your color scheme 440
 - How IDEA uses color schemes* 440
 - *Editor properties* 440
 - Changing font settings* 445
- 12.4 Customizing keyboard shortcuts 447
 - Using keyboard shortcuts* 448
 - *Navigating the interface with the keyboard* 448
 - *Selecting a keymap* 448
 - Creating a new keymap* 450
 - *Defining keyboard shortcuts* 450
 - *Defining mouse shortcuts* 452
 - Defining quick lists* 452
- 12.5 Working with non-Java file types 453
 - Modifying file types settings* 453
 - *Registering file extensions* 454
 - *Creating a custom file type* 454
- 12.6 Using file templates 455
 - Creating a new file from a template* 456
 - Working with template includes* 459

12.7 Increasing the amount of memory
allocated to IDEA 459

12.8 Summary 460

13 ***Extending IDEA*** 461

13.1 Working with bookmarks 462

Placing and using numbered bookmarks 463

Placing and using anonymous bookmarks 463

Managing your bookmarks collection 464

13.2 Working with ToDo lists 465

Creating custom ToDo list items 467

Using ToDo filters 468

13.3 The IDEA Commander 468

Working with the IDEA Commander 469

File operations 471

13.4 Integrating external tools with IDEA 472

Managing the tool list 472 ■ *Accessing external tools from within*

IDEA 476 ■ *Reacting to the results of external tools* 478

13.5 Using IDEA's open APIs 479

13.6 Summary 480

appendix Getting help with IDEA 481

index 487