

# Index

## ■ A

Abstract Window Toolkit (AWT), 111

Access modifiers

    package-private, 27

    private, 27

    protected, 27

    public, 27

Additive operators, 57–58

Animal classes

    animals speak, 106

    Carnivore interface, 105

    Cat class, 103

    Dog class, 104

    Herbivore interface, 105

    Mammal class, 102

    Mouse class, 104–105

    Predator interface, 105

    Scavenger interface, 105

Animation

    factors, 186

    frame rate, 185

    multiple objects

        class, color usage, 194–196

        Fireworks class, 192–193

        FireworksPanel class, 194

        fireworks program in action, 191

        Timer object, 191

    scoot ball game

        actionPerformed method, 188

        actionScootball, 190

        paint method, 190

        run method, 190

        Runnable interface, 188

        ScootBall class, 186–188

        ScootBallPanel class, 188–189

        starting state, 189

sprites

- definition, 196
- display, 202
- images of, 196
- MouseListener interface, 199
- MouseSprite object, 201
- MouseSpritePanel class, 199–200
- MouseSprites, 197–199
- sheet, 197
- specialization, 196
- stop feature, 202
- triggered animation, 197

timing, 185

Assignment operators, 66–67

■ **B**

Bitwise operator

- AND, 63
- complement, 54–55
- exclusive OR, 63
- inclusive OR, 64

Block comments, 31

Branching

- break statement, 88–89

continue statement, 89–91

return statement, 91–92

Breakpoint, Eclipse debugger, 208

■ **C**

Casting, 55–56

Classloading breakpoints, 209

Comments

- end-of-line, 31
- Javadoc, 32–33
- multi-line block, 31

Complexity code

- dense code, 19
- moderately simplified code, 19
- simplified code, 19

Composition technique, 98

Concurrent garbage collector, 260

Control flow

- if and if-else statements
  - complex if statement, 78
  - if-else statement, 78
  - larger else-if statement, 79
  - simple if statement, 78

## switch statements

- evaluation, 80

- issues, 81

- vs.* if statements, 80

■ **D**

## Data types

- arrays, 43–44

- boolean, 37

- char, 37

- enumerations, 45–48

- escaping characters, 39–41

- integer primitives, 35–36

- literals, 38–39

- null, 44–45

- primitive, definition, 35

- real primitives, 36

- string, 37–38

- wrapper classes, 41–43

## Debugging process

- methods, 206–2077

- without debugger, 207

## Document Object Model (DOM), 173

■ **E**

## Eclipse, 2

## Eclipse debugger

- breakpoint and variables, 208

- display, 208

- expanded view, value, 210

- Fireworks problem, 206

## Fireworks program

- corrected method, 217

- correct values, debugger, 218

- flawed method, 216

- incorrect values, debugger, 217

## line breakpoint

- code, 209

- conditional, 213

- condition enabled and specified, 214

- disable, 212

- executions, 210

- properties window, 213

- removable, 212

- settings, 209

- variables-conditional, 215

- procedure, 207
- stepping, 211–212
- tips and tricks, 215–216
- Eden space, 254
- Encapsulation, 96
- Entry point, 5
- Enumeration data types, 45–48
- Equality operators, 62–63
- Escaping characters, 39–41
- Exception breakpoints, 209
- Extensibility, 106
- Extensible Markup Language (XML)
  - and streams, 172
  - DOM and SAX, 173
  - Factory Classes, 184
  - reading
    - DefaultHandler, 181
    - with DOM, 179–181
    - with SAX, 183
    - XMLToConsoleHandler class, 181–183
  - structure
    - attributes, 169
    - basic rules, 170
    - document specifier, 170

- root element, 170
- single element, 169
- SOAP, 172
- storage format, 169
- UTF-8, 170
- XMLNS, 172
- writing
  - data source, 173–174
  - with DOM, 174–177
  - with strings, 178–179

## ■ F

- Fibonacci sequence, 267
- File object
  - deleting a directory, 159
  - deleting a file, 154–155
  - deleting multiple directories, 160–161
  - directory creation, 157–158
  - double backslashes, 152
  - empty file creation, 152, 153
  - exception, 153
  - exists() method, 153
  - mkdir method, 158
  - multiple directories creation, 160–161

- opening a file, 153
  - temporary file, 155–157
  - test directory, 152
- Fireworks program
- class, color usage, 194–196
  - display, 191
  - Fireworks class, 192–193
  - FireworksPanel class, 194
  - Timer object, 191
- Fractal tree program
- class, 273–274
  - drawing tree, 274–275
  - output, 276
- **G**
- G1, concurrent garbage collector, 260
- Garbage collection
- event-driven, 255
  - G1, 260
  - generations, 254–255
  - hints, 258
  - memory allocation
    - algorithm, marking and sweeping, 251–252
    - references, 251
    - settings, 252–253
    - TargetClickPanel class, 249–250
  - reference list, 259–260
  - scavenges and full collections, 255
  - singleton, 259
  - switches, 255, 257–258
- Generic specifier
- coding, 279–280
  - definition, 279
  - GenericRole class, 280
  - hashmap, 279
  - integerList, 282
  - interface, 281
  - LinkedList class, 282
  - multiple parameters, 280
  - Ordinary List, 282
  - super keyword, 281
  - type match error, 282
  - wildcard, 281
- Granularity
- extensibility, 106
  - maintainability, 106

■ I

Inheritance, 96

abstract classes

concrete class, 98

methods, 99–100

interfaces, 98

multiple inheritance

composition technique, 98

diamond problem, 97

static members, 100–101

Instantiation, 26

Integrated Development Environment (IDE). *See* Eclipse

■ J

Java Development Kit (JDK), 1–2

Java operators

additive, 57–58

assignment, 66–67

bitwise

AND, 63

complement, 54–55

exclusive OR, 63

inclusive OR, 64

casting, 55–56

equality, 62–63

equal method, 68–70

java.lang.Comparable, 71–75

logical

AND, 64

OR, 65

multiplicative, 57

operator precedence, 52

parentheses, 52–53

postfix, 53

relational, 60–62

shift, 58–60

sorting comparisons, 70

unary, 53

Java programs

args array, 9–10

creation

Eclipse’s New Project window, 3

Java Project, 3–4

main area, 5

date format, 11

Eclipse, 2

entry point, 5

JDK, 1–2

- main method, 6
  - New Java Class window display, 6–7
  - objects, 12
  - package, 8
  - plus sign-string concatenation operator, 10
  - Run button, 8–9
  - Run Configurations window, 10
  - String[] args, 9
  - Java Swing
    - application, 112
    - basics, 111–112
    - components, 116
    - content pane, 113
    - createAndShowGUI method, 116
    - JFrame object, 112
    - layout, 114
    - menu, 114
    - Minesweeper program (*see* MineSweeper)
    - MouseListener interface, 119
    - pack and setVisible methods, 113
  - Java syntax
    - access modifiers, 27
    - Average interface, 15–16
    - AverageImpl, 16–17
    - AverageTest class, 17–18
    - blocks, 30–31
    - classes, 22–23
    - comments, 31–33
    - complex code
      - dense code, 19
      - moderately simplified code, 19
      - simplified code, 19
    - constants, 24
    - constructors, 26
    - exceptions, 28–30
    - fields, 23–24
    - imports, 21–22
    - interfaces, 27–28
    - lines, 18–19
    - methods, 24–25
    - package declaration, 20–21
  - Java2D, 111
  - Javadoc comment, 32–33
  - JDK. *See* Java Development Kit (JDK)
- **L**
- Line breakpoints, 209

List processing (Lisp), 264

Logical operator

AND, 64

OR, 65

Looping

Do-while loops, 87–88

for loops

alternate for loops, 83

ArrayOutOfBounds, 83

compassPoints array, 84

control section, 83

enhanced for syntax, 84

increment code, 83

initialization code, 83

termination code, 83

while loops

complete while loop, 86

simple while loop, 85

Thread.sleep() method, 87

wait for event, 86

## ■ M

Maintainability, 106

Method breakpoints, 209

MineSweeper

AddMenu method, 127

class, 120

constructor, 124

createAndShowGUI method, 126

description, 119

field definitions, 129

finished game, 148

init method, 125

JPanel objects, 127

main method, 124

Mine class, 137

MineField

cascade methods, 135

class, 130

constructor, 133

countAdjacentMines methods, 137

init method, 134

MineIcon class, 144

MineSweeperActionListener class, 147

MineSweeperMouseListener class, 145

populate methods, 134

program, 121

recursion, 135



## MineSweeperHelper

- class, 138
- constructor, 140
- endGame method, 142
- newGame method, 143
- showAll method, 142
- updateButtons method, 141
- updateLabels method, 141

## Multiplicative operators, 57

## ■ O

### Objects, 95–96

## ■ P

### Pass-by-reference and pass-by-value, 107

- IntegerWrapper class*, 107, 108
- passBy method*, 108
- PassByTest class*, 107

### Polymorphism, 101

### Postfix operators, 53

### Primitive data types

- definition, 35
- integer primitives, 35–36
- real primitives, 36

### Printpoints, 209

## ■ R

### Recursion

- avoidance, 265–266
- Fibonacci sequence calculation, 267
- fractals
  - drawing tree (*see* Fractal tree program)
  - Sierpinski triangle( *see* Sierpinski triangle)
- human language, 263
- Lisp, 264
- stop condition, 264–265
- usage, 266

### RegexTester

- arguments, 284
- class, 283
- empty parameters, 285
- output, 286
- populated parameters, 286
- Run configurations, 284

### Regular expressions

- definition, 279
- metacharacters, 287–288
- RegexTester
  - arguments, 284

- class, 283
- empty parameters, 285
- output, 286
- populated parameters, 286
- Run configurations, 284

string literal, 287

Relational operators, 60–62

## ■ S

SAX. *See* Simplified API for XML (SAX)

Scoot ball game

- actionPerformed method, 188

- actionScootball, 190

- paint method, 190

- run method, 190

Runnable interface, 188

ScootBall class, 186–188

ScootBallPanel class, 188–189

starting state, 189

Shift operators, 58–60

Shooting gallery game

- actionPerformed method, 236

- class, 232–233

- display, 231

- expansion, 245

- limitations, 246

- panel class, 234–236

- polygon object, 240

- resize prevention, 233

- row class, 238–240

- shooter class, 240–244

- static variables, 245

- target class, 238–240

- target sprites class, 244–245

Sierpinski triangle

- class, 268–269

- definition, 268

- drawing triangle, 269–271

- drawTriangle methods, 271

- output, 272

Simplified API for XML (SAX), 173

Simplified Object Access Protocol (SOAP),  
172

SOAP. *See* Simplified Object Access  
Protocol (SOAP)

Sprites

- definition, 196

- display, 202

- images of, 196

- MouseListener interface, 199
- MouseListener object, 201
- MouseListenerPanel class, 199–200
- MouseSprites, 197–199
  - sheet, 197
  - specialization, 196
  - stop feature, 202
  - triggered animation, 197

Static blocks, 30–31

Stepping commands

- step controls, debug toolbar, 211
- step into, 212
- step over, 212
- step return, 212

String concatenation operator, 10

String type, 37–38

Stub, 102

Survivor spaces, 254

## ■ T

TargetClick game

- actionPerformed method, 227
- class, 223–226
- display, 223

Panel class, 227–228

Target class, 229–230

Whac-a-Mole, 222

## ■ U

Unary operators, 53

User interface. *See* Java Swing

## ■ V

Video games

- design resources, 246–247

- mechanics

- game logic, 222

- game loop, 222

- user interface, 221–222

- shooting gallery

- actionPerformed method, 236

- class, 232–233

- display, 231

- expansion, 245

- limitations, 246

- paintComponent method, 240

- panel class, 234–236

- polygon object, 240

- resize prevention, 233

- row class, 238–240
- shooter class, 240–244
- static variables, 245
- target class, 238–240
- target sprites class, 244–245

### TargetClick

- actionPerformed method, 227
- class, 223–226
- display, 223
- Panel class, 227–228
- Target class, 229–230
- Whac-a-Mole, 222

## ■ W, X, Y, Z

Watchpoints, 209

Wrapper classes, 41–43

Writing and reading content

append content, 166

Hamlet.txt

- in console, 162

- original content, 162

- reverse, 165

- writing to file, 163

in-place reversing, 165

original and reversed content, 167

reverse method

- reverseByteArray method, 165

- StringBuffer class, 165

stream, 161

string object, 163

try-catch block, 163

XML. *See* Extensible Markup Language (XML)

XML NameSpace (XMLNS), 172

XMLNS. *See* XML NameSpace (XMLNS)