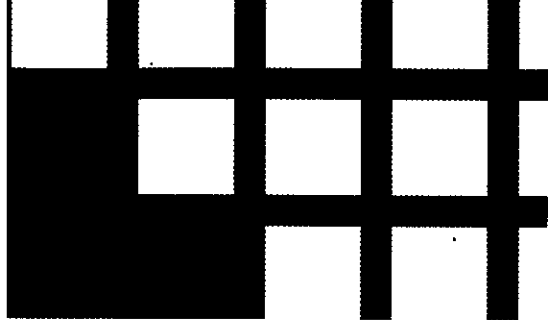


Stuart Reges | Marty Stepp

Building **Java**[™] Programs

A Back to Basics
Approach



Preface

iii

Chapter I Introduction to Java Programming

I

I.1 Basic Computing Concepts

2

Why Programming?

2

Hardware and Software

3

The Digital Realm

4

The Process of Programming

5

Why Java?

7

The Java Programming Environment

8

I.2 And Now—Java

10

String Literals (Strings)

13

`System.out.println`

14

Escape Sequences

15

Identifiers and Keywords

16

A Complex Example: `DrawFigures1`

18

Comments and Readability

19

I.3 Program Errors

21

Syntax Errors

22

Logic Errors (Bugs)

24

I.4 Procedural Decomposition

25

Static Methods

28

Flow of Control

31

Methods That Call Other Methods

33

An Example Runtime Error

35

I.5 Case Study: `DrawFigures`

37

Structured Version

37

Final Version Without Redundancy

38

Analysis of Flow of Execution

40

Chapter 2 Primitive Data and Definite Loops	53
2.1 Basic Data Concepts	54
Primitive Types	54
Expressions	55
Literals	57
Arithmetic Operators	58
Precedence	60
Mixing Types and Casting	62
2.2 Variables	63
Assignment/Declaration Variations	68
String Concatenation	70
Increment/Decrement Operators	73
Variables and Mixing Types	76
2.3 The for Loop	78
Tracing for Loops	80
print Versus println	83
Nested for Loops	85
2.4 Managing Complexity	87
Scope	88
Pseudocode	93
A Decrementing for Loop	95
Class Constants	96
2.5 Case Study: A Complex Figure	98
Problem Decomposition and Pseudocode	98
Initial Structured Version	100
Adding a Class Constant	101
Further Variations	104
 Chapter 3 Introduction to Parameters and Objects	 117
3.1 Parameters	118
The Mechanics of Parameters	121
Limitations of Parameters	124
Multiple Parameters	126
Parameters Versus Constants	128
Overloading of Methods	129
3.2 Methods That Return Values	130
The Math Class	131
Defining Methods That Return Values	133

3.3	Using Objects	136
	String Objects	137
	Point Objects	143
	Reference Semantics	146
	Multiple Objects	148
	Objects as Parameters to Methods	150
3.4	Interactive Programs	152
	Scanner Objects	153
	Sample Interactive Program	155
3.5	Case Study: Projectile Trajectory	156
	Unstructured Solution	160
	Structured Solution	162
Supplement 3G	Graphics (Optional)	174
3G.1	Introduction to Graphics	175
	DrawingPanel	175
	Drawing Lines and Shapes	176
	Colors	180
	Text and Fonts	185
3G.2	Procedural Decomposition with Graphics	188
	A Larger Example: DrawDiamonds	188
	Summary of Graphics Methods	191
3G.3	Case Study: Pyramids	192
	Unstructured Partial Solution	193
	Generalizing the Drawing of Pyramids	194
	Complete Structured Solution	195
Chapter 4	Conditional Execution	207
4.1	Loop Techniques	208
	Cumulative Sum	208
	Fencepost Loops	210
4.2	if/else Statements	211
	Relational Operators	214
	Cumulative Sum with if	215
	Fencepost with if	217
	Nested if/else Statements	219
4.3	Subtleties of Conditional Execution	224
	Object Equality	226
	Roundoff Errors	228

Contents

	Factoring if/else Statements	230
	Min/Max Loops	232
4.4	Text Processing	235
	The char Type	235
	System.out.printf (Optional)	238
4.5	Methods with Conditional Execution	243
	Preconditions and Postconditions	243
	Throwing Exceptions	243
	Revisiting Return Values	247
4.6	Case Study: Body Mass Index	251
	One-Person Unstructured Solution	252
	Two-Person Unstructured Solution	254
	Two-Person Structured Solution	256
Chapter 5	Program Logic and Indefinite Loops	271
5.1	The while Loop	272
	A Loop to Find the Smallest Divisor	273
	Sentinel Loops	274
	Random Numbers	276
5.2	The boolean Type	279
	Logical Operators	281
	Short-Circuited Evaluation	284
	Boolean Variables and Flags	288
	Boolean Zen	289
5.3	User Errors	292
	Scanner Lookahead	292
	Handling User Errors	294
5.4	Indefinite Loop Variations	295
	The do/while Loop	295
	Break and “Forever” Loops (Optional)	299
5.5	Assertions and Program Logic	300
	Reasoning About Assertions	303
	A Detailed Assertions Example	304
	The Java assert Statement	308
5.6	Case Study: NumberGuess	309
	Initial Version Without Hinting	310
	Randomized Version with Hinting	312
	Final Robust Version	314

Chapter 6 File Processing	331
6.1 File-Reading Basics	332
Data, Data Everywhere	332
Files and File Objects	332
Reading a File with a Scanner	335
6.2 Details of Token-Based Processing	339
Structure of Files and Consuming Input	341
Scanner Parameters	345
Paths and Directories	346
A More Complex Input File	348
6.3 Line-Based Processing	350
String Scanners and Line/Token Combinations	351
6.4 Advanced File Processing	355
Output Files with PrintStream	355
try/catch Statements	359
6.5 Case Study: Weighted GPA	364
 Chapter 7 Arrays	 375
7.1 Array Basics	376
Constructing and Traversing an Array	376
Accessing an Array	380
A Complete Array Program	382
Random Access	386
Arrays and Methods	391
The For-Each Loop	394
Initializing Arrays	396
Limitations of Arrays	397
7.2 Array-Traversal Algorithms	398
Printing an Array	398
Searching and Replacing	400
Testing for Equality	402
Reversing an Array	405
7.3 Advanced Array Techniques	408
Shifting Values in an Array	408
Arrays of Objects	413
Command-Line Arguments	414

Contents

7.4	Multidimensional Arrays (Optional)	414
	Rectangular Two-Dimensional Arrays	415
	Jagged Arrays	417
7.5	Case Study: Hours Worked	420
	Version 1: Reading the Input File	421
	Version 2: Cumulative Sum	424
	Version 3: Row Sum and Column Print	426

Chapter 8 Classes **437**

8.1	Object-Oriented Programming Concepts	438
	Classes and Objects	439
8.2	Object State: Fields	441
8.3	Object Behavior: Methods	443
	The Implicit Parameter	446
	Mutators and Accessors	447
8.4	Object Initialization: Constructors	449
8.5	Encapsulation	453
	Private Fields	455
	Class Invariants	457
8.6	More Instance Methods	460
	The toString Method	460
	The equals Method	462
	The instanceof Keyword	466
8.7	The Keyword this	469
	Multiple Constructors	472
8.8	More Classes	475
	Participant	475
	TimeSpan	477
	Changing Internal Implementations	480
8.9	Case Study: Designing a Stock Class	482
	Object-Oriented Design Heuristics	483
	Stock Fields and Method Headers	485
	Stock Method and Constructor Implementation	486

Chapter 9 Inheritance and Interfaces **501**

9.1	Inheritance Basics	502
	Non-Programming Hierarchies	503
	Extending a Class	505

Overriding Methods	508
Polymorphism	510
9.2 The Mechanics of Polymorphism	512
Diagramming Polymorphic Code	514
9.3 Interacting with the Superclass	517
Inherited Fields	517
Calling a Superclass's Constructor	519
Calling Overridden Methods	520
Another Example: MeteredPoint	522
The Object Class	524
9.4 Inheritance and Design	525
A Misuse of Inheritance: Point3D	525
Is-a Versus Has-a Relationships	527
Graphics2D (Optional)	528
9.5 Interfaces	530
An Interface for Shape Classes	531
Implementing the Shape Interface	533
Benefits of Interfaces	535
Interfaces in the Java Class Libraries	537
9.6 Case Study: Designing a Hierarchy of Financial Classes	537
Designing the Classes	538
Initial Redundant Implementation	541
Abstract Classes	544
 Chapter 10 ArrayLists	 559
10.1 ArrayLists	560
Basic ArrayList Operations	561
ArrayList Searching Methods	564
A Complete ArrayList Program	566
Adding To and Removing From an ArrayList	567
Using the For-Each Loop with ArrayLists	571
Wrapper Classes	572
10.2 The Comparable Interface	575
Natural Ordering and compareTo	577
Implementing Comparable	580
10.3 Case Study: Vocabulary Comparison	584
Version 1: Compute Vocabulary	584
Version 2: Compute Overlap	590
Version 3: Complete Program	594

Chapter 11 Java Collections Framework	603
11.1 Lists	604
Collections	604
LinkedList Versus ArrayList	605
Iterators	608
Abstract Data Types (ADTs)	611
LinkedList Case Study: Sieve	613
11.2 Sets	616
Set Concepts	617
TreeSet Versus HashSet	619
Set Operations	620
Set Case Study: Lottery	622
11.3 Maps	624
Basic Map Operations	625
Map Views (keySet and values)	626
TreeMap Versus HashMap	628
Map Case Study: Word Count	628
Collection Overview	630
Chapter 12 Recursion	639
12.1 Thinking Recursively	640
A Non-Programming Example	640
An Iterative Solution Converted to Recursion	643
Structure of Recursive Solutions	645
12.2 A Better Example of Recursion	647
Mechanics of Recursion	648
12.3 Recursive Functions	656
Integer Exponentiation	656
Greatest Common Divisor	658
Directory Crawler	663
12.4 Recursive Graphics (Optional)	667
12.5 Case Study: Prefix Evaluator	672
Infix, Prefix, and Postfix Notation	672
Evaluating Prefix Expressions	673
Complete Program	676

Chapter 13 Searching and Sorting	685
13.1 Searching and Sorting in the Java Class Libraries	686
Binary Search	686
Sorting	689
Shuffling	690
Custom Ordering with Comparators	691
13.2 Program Efficiency	695
Empirical Analysis	698
Complexity Classes	700
13.3 Implementing Searching and Sorting Algorithms	703
Sequential Search	704
Binary Search	704
Recursive Binary Search	708
Searching Objects	711
Selection Sort	711
13.4 Case Study: Implementing Merge Sort	714
Splitting and Merging Arrays	715
Recursive Merge Sort	718
Complete Program	720
 Chapter 14 Graphical User Interfaces	 727
14.1 GUI Basics	728
Graphical Input and Output with Option Panes	728
Working with Frames	731
Buttons, Labels, and Text Fields	735
Changing a Frame's Layout	737
Handling an Action Event	739
14.2 Laying Out Components	740
Layout Managers	741
Composite Layouts	744
14.3 Interaction Between Components	748
Example 1: BMI GUI	748
Object-Oriented GUIs	749
Example 2: Credit Card GUI	752
14.4 Additional Components and Events	756
Text Areas, Scrollbars, and Fonts	757
Icons	759
Mouse Events	760

Contents

14.5 2D Graphics	765
Drawing onto Panels	765
Animation with Timers	767
14.6 Case Study: Implementing DrawingPanel	770
Initial Version Without Events	770
Second Version with Events	772
 Appendix A Answers to Self-Check Problems	 783
 Appendix B Java Summary	 843
 Appendix C Javadoc Comments and the Java API Specification	 857
 Index	 863