

Contents

Figures	xix
Tables	xxi
Examples	xxiii
Foreword	xxvii
Preface	xxix
1 Basics of Java Programming	1
1.1 Introduction	2
1.2 Classes	2
Declaring Members: Fields and Methods	3
1.3 Objects	4
Class Instantiation, Reference Values, and References	4
Object Aliases	6
1.4 Instance Members	6
Invoking Methods	7
1.5 Static Members	7
1.6 Inheritance	10
1.7 Associations: Aggregation and Composition	12
1.8 Tenets of Java	13
<i>Review Questions</i>	13
1.9 Java Programs	15
1.10 Sample Java Application	16
Essential Elements of a Java Application	16
Compiling and Running an Application	17
1.11 Program Output	18
Formatted Output	18

1.12	The Java Ecosystem	21
	Object-Oriented Paradigm	22
	Interpreted: The JVM	23
	Architecture-Neutral and Portable Bytecode	23
	Simplicity	23
	Dynamic and Distributed	23
	Robust and Secure	24
	High Performance and Multithreaded	24
	<i>Review Questions</i>	25
	<i>Chapter Summary</i>	25
	<i>Programming Exercise</i>	26
2	Language Fundamentals	27
2.1	Basic Language Elements	28
	Lexical Tokens	28
	Identifiers	28
	Keywords	29
	Separators	29
	Literals	30
	Integer Literals	30
	Floating-Point Literals	31
	Underscores in Numerical Literals	32
	Boolean Literals	32
	Character Literals	32
	String Literals	34
	Whitespace	35
	Comments	35
	<i>Review Questions</i>	36
2.2	Primitive Data Types	37
	The Integer Types	38
	The char Type	38
	The Floating-Point Types	38
	The boolean Type	39
	<i>Review Questions</i>	40
2.3	Variable Declarations	40
	Declaring and Initializing Variables	41
	Reference Variables	41
2.4	Initial Values for Variables	42
	Default Values for Fields	42
	Initializing Local Variables of Primitive Data Types	43
	Initializing Local Reference Variables	43
	Lifetime of Variables	44
	<i>Review Questions</i>	45
	<i>Chapter Summary</i>	46
	<i>Programming Exercise</i>	46

3	Declarations	47
3.1	Class Declarations	48
3.2	Method Declarations	49
	Statements	50
	Instance Methods and the Object Reference this	50
	Method Overloading	52
3.3	Constructors	53
	The Default Constructor	54
	Overloaded Constructors	56
	<i>Review Questions</i>	56
3.4	Arrays	58
	Declaring Array Variables	59
	Constructing an Array	59
	Initializing an Array	60
	Using an Array	61
	Anonymous Arrays	63
	Multidimensional Arrays	64
	Sorting Arrays	68
	Searching Arrays	69
	<i>Review Questions</i>	70
3.5	Parameter Passing	72
	Passing Primitive Data Values	73
	Passing Reference Values	75
	Passing Arrays	77
	Array Elements as Actual Parameters	78
	final Parameters	80
3.6	Variable Arity Methods	81
	Calling a Variable Arity Method	82
	Variable Arity and Fixed Arity Method Calls	84
3.7	The main() Method	85
	Program Arguments	86
3.8	Enumerated Types	87
	Declaring Type-safe Enums	87
	Using Type-safe Enums	88
	Selected Methods for Enum Types	89
	<i>Review Questions</i>	90
	<i>Chapter Summary</i>	92
	<i>Programming Exercise</i>	93
4	Access Control	95
4.1	Java Source File Structure	96
4.2	Packages	97
	Defining Packages	98
	Using Packages	99
	Compiling Code into Packages	105

	Running Code from Packages	106
4.3	Searching for Classes	107
	<i>Review Questions</i>	110
4.4	Scope Rules	114
	Class Scope for Members	114
	Block Scope for Local Variables	117
4.5	Accessibility Modifiers for Top-Level Type Declarations	118
4.6	Non-Accessibility Modifiers for Classes	120
	abstract Classes	120
	final Classes	122
	<i>Review Questions</i>	123
4.7	Member Accessibility Modifiers	123
	public Members	124
	protected Members	126
	Default Accessibility for Members	127
	private Members	128
	<i>Review Questions</i>	129
4.8	Non-Accessibility Modifiers for Members	131
	static Members	132
	final Members	133
	abstract Methods	136
	synchronized Methods	136
	native Methods	137
	transient Fields	138
	volatile Fields	139
	<i>Review Questions</i>	140
	<i>Chapter Summary</i>	142
	<i>Programming Exercise</i>	142
5	Operators and Expressions	143
5.1	Conversions	144
	Widening and Narrowing Primitive Conversions	144
	Widening and Narrowing Reference Conversions	145
	Boxing and Unboxing Conversions	145
	Other Conversions	146
5.2	Type Conversion Contexts	147
	Assignment Context	147
	Method Invocation Context	148
	Casting Context of the Unary Type Cast Operator: (<i>type</i>)	148
	Numeric Promotion Context	149
5.3	Precedence and Associativity Rules for Operators	150
5.4	Evaluation Order of Operands	152
	Left-Hand Operand Evaluation First	152
	Operand Evaluation before Operation Execution	153
	Left-to-Right Evaluation of Argument Lists	154

5.5	Representing Integers	154
	Calculating Two's Complement	155
	Converting Binary Numbers to Decimals	157
	Converting Decimals to Binary Numbers	157
	Relationships among Binary, Octal, and Hexadecimal Numbers	157
5.6	The Simple Assignment Operator =	158
	Assigning Primitive Values	159
	Assigning References	159
	Multiple Assignments	159
	Type Conversions in an Assignment Context	160
	<i>Review Questions</i>	162
5.7	Arithmetic Operators: *, /, %, +, -	163
	Arithmetic Operator Precedence and Associativity	164
	Evaluation Order in Arithmetic Expressions	164
	Range of Numeric Values	164
	Unary Arithmetic Operators: -, +	167
	Multiplicative Binary Operators: *, /, %	167
	Additive Binary Operators: +, -	169
	Numeric Promotions in Arithmetic Expressions	170
	Arithmetic Compound Assignment Operators: *=, /=, %=, +=, -=	172
	<i>Review Questions</i>	173
5.8	The Binary String Concatenation Operator +	174
5.9	Variable Increment and Decrement Operators: ++, --	176
	The Increment Operator ++	176
	The Decrement Operator --	176
	<i>Review Questions</i>	178
5.10	Boolean Expressions	180
5.11	Relational Operators: <, <=, >, >=	180
5.12	Equality	181
	Primitive Data Value Equality: ==, !=	181
	Object Reference Equality: ==, !=	182
	Object Value Equality	183
5.13	Boolean Logical Operators: !, ^, &,	184
	Operand Evaluation for Boolean Logical Operators	185
	Boolean Logical Compound Assignment Operators: &=, ^=, =	185
5.14	Conditional Operators: &&,	186
	Short-Circuit Evaluation	187
5.15	Integer Bitwise Operators: ~, &, , ^	189
	Bitwise Compound Assignment Operators: &=, ^=, =	192
	<i>Review Questions</i>	192
5.16	The Conditional Operator: ? :	194
5.17	Other Operators: new, [], instanceof, ->	195
	<i>Review Questions</i>	196
	<i>Chapter Summary</i>	197
	<i>Programming Exercise</i>	197

6	Control Flow	199
6.1	Overview of Control Flow Statements	200
6.2	Selection Statements	200
	The Simple if Statement	200
	The if-else Statement	201
	The switch Statement	203
	<i>Review Questions</i>	210
6.3	Iteration Statements	213
	The while Statement	213
	The do-while Statement	214
	The for(;;) Statement	215
	The for(:) Statement	217
6.4	Transfer Statements	219
	Labeled Statements	220
	The break Statement	221
	The continue Statement	223
	The return Statement	224
	<i>Review Questions</i>	226
6.5	Stack-Based Execution and Exception Propagation	230
6.6	Exception Types	233
	The Exception Class	235
	The RuntimeException Class	236
	The Error Class	237
	Checked and Unchecked Exceptions	237
	Defining Customized Exceptions	238
6.7	Exception Handling: try, catch, and finally	238
	The try Block	240
	The catch Clause	240
	The finally Clause	245
6.8	The throw Statement	249
6.9	The throws Clause	251
	Overriding the throws Clause	253
6.10	Advantages of Exception Handling	254
	<i>Review Questions</i>	255
	<i>Chapter Summary</i>	258
	<i>Programming Exercises</i>	258
7	Object-Oriented Programming	263
7.1	Single Implementation Inheritance	264
	Relationships: is-a and has-a	266
	The Supertype–Subtype Relationship	267
7.2	Overriding Methods	268
	Instance Method Overriding	268
	Covariant return in Overriding Methods	273
	Overriding versus Overloading	273

7.3	Hiding Members	275
	Field Hiding	275
	Static Method Hiding	275
7.4	The Object Reference <code>super</code>	276
	<i>Review Questions</i>	279
7.5	Chaining Constructors Using <code>this()</code> and <code>super()</code>	282
	The <code>this()</code> Constructor Call	282
	The <code>super()</code> Constructor Call	285
	<i>Review Questions</i>	288
7.6	Interfaces	290
	Defining Interfaces	290
	Abstract Methods in Interfaces	291
	Implementing Interfaces	291
	Extending Interfaces	294
	Interface References	296
	Default Methods in Interfaces	297
	Static Methods in Interfaces	300
	Constants in Interfaces	302
	<i>Review Questions</i>	304
7.7	Arrays and Subtyping	309
	Arrays and Subtype Covariance	309
	Array Store Check	311
7.8	Reference Values and Conversions	311
7.9	Reference Value Assignment Conversions	312
7.10	Method Invocation Conversions Involving References	315
	Overloaded Method Resolution	316
7.11	Reference Casting and the <code>instanceof</code> Operator	320
	The Cast Operator	320
	The <code>instanceof</code> Operator	321
	<i>Review Questions</i>	325
7.12	Polymorphism and Dynamic Method Lookup	329
7.13	Inheritance versus Aggregation	331
7.14	Basic Concepts in Object-Oriented Design	334
	Encapsulation	335
	Cohesion	335
	Coupling	336
	<i>Review Questions</i>	336
	<i>Chapter Summary</i>	338
	<i>Programming Exercises</i>	339
8	Fundamental Classes	341
8.1	Overview of the <code>java.lang</code> Package	342
8.2	The <code>Object</code> Class	342
	<i>Review Questions</i>	346

8.3	The Wrapper Classes	346
	Common Wrapper Class Constructors	347
	Common Wrapper Class Utility Methods	348
	Numeric Wrapper Classes	351
	The Character Class	354
	The Boolean Class	355
	<i>Review Questions</i>	355
8.4	The String Class	357
	Immutability	357
	Creating and Initializing Strings	357
	The CharSequence Interface	360
	Reading Characters from a String	361
	Comparing Strings	363
	Character Case in a String	364
	Concatenation of Strings	364
	Joining of CharSequence Objects	365
	Searching for Characters and Substrings	367
	Extracting Substrings	369
	Converting Primitive Values and Objects to Strings	369
	Formatted Strings	370
	<i>Review Questions</i>	371
8.5	The StringBuilder and StringBuffer Classes	374
	Thread-Safety	374
	Mutability	374
	Constructing String Builders	374
	Reading and Changing Characters in String Builders	375
	Constructing Strings from String Builders	375
	Appending, Inserting, and Deleting Characters in String Builders	376
	Controlling String Builder Capacity	378
	<i>Review Questions</i>	379
	<i>Chapter Summary</i>	382
	<i>Programming Exercises</i>	382
9	Object Lifetime	383
9.1	Garbage Collection	384
9.2	Reachable Objects	384
9.3	Facilitating Garbage Collection	386
9.4	Object Finalization	390
9.5	Finalizer Chaining	391
9.6	Invoking Garbage Collection Programmatically	393
	<i>Review Questions</i>	396
9.7	Initializers	399
9.8	Field Initializer Expressions	400
	Declaration Order of Initializer Expressions	401
9.9	Static Initializer Blocks	402

CONTENTS

xvii

Declaration Order of Static Initializers	403
9.10 Instance Initializer Blocks	404
Declaration Order of Instance Initializers	405
9.11 Constructing Initial Object State	406
<i>Review Questions</i>	409
<i>Chapter Summary</i>	411
10 The ArrayList<E> Class and Lambda Expressions	413
10.1 The ArrayList<E> Class	414
Lists	414
Declaring References and Constructing ArrayLists	415
Modifying an ArrayList	419
Querying an ArrayList	422
Traversing an ArrayList	423
Converting an ArrayList to an Array	424
Sorting an ArrayList	425
Arrays versus ArrayList	425
<i>Review Questions</i>	430
10.2 Lambda Expressions	433
Behavior Parameterization	434
Functional Interfaces	442
Defining Lambda Expressions	444
Type Checking and Execution of Lambda Expressions	450
Filtering Revisited: The Predicate<T> Functional Interface	451
<i>Review Questions</i>	455
<i>Chapter Summary</i>	458
<i>Programming Exercise</i>	458
11 Date and Time	461
11.1 Basic Date and Time Concepts	462
11.2 Working with Temporal Classes	462
Creating Temporal Objects	464
Querying Temporal Objects	468
Comparing Temporal Objects	470
Creating Modified Copies of Temporal Objects	470
Temporal Arithmetic	474
11.3 Working with Periods	476
Creating Periods	476
Querying Periods	478
Creating Modified Copies of Periods	479
More Temporal Arithmetic	479
<i>Review Questions</i>	483
11.4 Formatting and Parsing	486
Default Formatters	487

Predefined Formatters	488
Localized Formatters	490
Customized Formatters	495
<i>Review Questions</i>	500
<i>Chapter Summary</i>	502
<i>Programming Exercise</i>	503
A Taking the Java SE 8 Programmer I Exam	507
A.1 Preparing for the Exam	507
A.2 Registering for the Exam	508
Contact Information	509
Obtaining an Exam Voucher	509
Signing Up for the Test	509
After Taking the Exam	509
A.3 How the Exam Is Conducted	510
The Testing Locations	510
Utilizing the Allotted Time	510
The Exam Program	510
The Exam Result	511
A.4 The Questions	511
Assumptions about the Exam Questions	511
Types of Questions Asked	512
Types of Answers Expected	512
Topics Covered by the Questions	513
B Exam Topics: Java SE 8 Programmer I	515
C Annotated Answers to Review Questions	519
D Solutions to Programming Exercises	553
E Mock Exam: Java SE 8 Programmer I	571
F Annotated Answers to Mock Exam I	605
Index	619

Figures

1.1	UML Notation for Classes	3
1.2	UML Notation for Objects	5
1.3	Aliases	6
1.4	Class Diagram Showing Static Members of a Class	8
1.5	Members of a Class	9
1.6	Class Diagram Depicting Inheritance Relationship	10
1.7	Class Diagram Depicting Associations	12
1.8	Class Diagram Depicting Composition	13
2.1	Primitive Data Types in Java	37
3.1	Array of Arrays	67
3.2	Parameter Passing: Primitive Data Values	75
3.3	Parameter Passing: Reference Values	76
3.4	Parameter Passing: Arrays	78
4.1	Java Source File Structure	96
4.2	Package Hierarchy	97
4.3	File Hierarchy	107
4.4	Searching for Classes	108
4.5	Block Scope	117
4.6	Public Accessibility for Members	124
4.7	Protected Accessibility for Members	127
4.8	Default Accessibility for Members	128
4.9	Private Accessibility for Members	129
5.1	Widening Primitive Conversions	144
5.2	Converting among Binary, Octal, and Hexadecimal Numbers	158
5.3	Overflow and Underflow in Floating-Point Arithmetic	166
5.4	Numeric Promotion in Arithmetic Expressions	170
6.1	Activity Diagram for <code>if</code> Statements	201
6.2	Activity Diagram for a <code>switch</code> Statement	204
6.3	Activity Diagram for the <code>while</code> Statement	213
6.4	Activity Diagram for the <code>do-while</code> Statement	214
6.5	Activity Diagram for the <code>for</code> Statement	215
6.6	Enhanced <code>for</code> Statement	217
6.7	Method Execution	231

6.8	Exception Propagation	233
6.9	Partial Exception Inheritance Hierarchy	234
6.10	The try-catch-finally Construct	239
6.11	Exception Handling (Scenario 1)	241
6.12	Exception Handling (Scenario 2)	243
6.13	Exception Handling (Scenario 3)	245
7.1	Inheritance Hierarchy	267
7.2	Inheritance Hierarchy for Example 7.2 and Example 7.3	269
7.3	Inheritance Hierarchies	295
7.4	Inheritance Relationships for Interface Constants	303
7.5	Reference Type Hierarchy: Arrays and Subtype Covariance	310
7.6	Type Hierarchy That Illustrates Polymorphism	330
7.7	Implementing Data Structures by Inheritance and Aggregation	332
8.1	Partial Inheritance Hierarchy in the java.lang Package	342
8.2	Converting Values among Primitive, Wrapper, and String Types	347
9.1	Memory Organization at Runtime	386
10.1	Partial ArrayList Inheritance Hierarchy	415

Tables

1.1	Terminology for Class Members	9
1.2	Format Specifier Examples	20
2.1	Keywords in Java	29
2.2	Reserved Literals in Java	29
2.3	Reserved Keywords Not Currently in Use	29
2.4	Separators in Java	29
2.5	Examples of Literals	30
2.6	Examples of Decimal, Binary, Octal, and Hexadecimal Literals	30
2.7	Examples of Character Literals	33
2.8	Escape Sequences	33
2.9	Examples of Escape Sequence <code>\ddd</code>	34
2.10	Range of Integer Values	38
2.11	Range of Character Values	38
2.12	Range of Floating-Point Values	39
2.13	Boolean Values	39
2.14	Summary of Primitive Data Types	39
2.15	Default Values	42
3.1	Parameter Passing by Value	73
4.1	Accessing Members within a Class	115
4.2	Summary of Accessibility Modifiers for Top-Level Types	120
4.3	Summary of Non-Accessibility Modifiers for Classes	122
4.4	Summary of Accessibility Modifiers for Members	129
4.5	Summary of Non-Accessibility Modifiers for Members	139
5.1	Selected Conversion Contexts and Conversion Categories	147
5.2	Operator Summary	151
5.3	Representing Signed byte Values Using Two's Complement	155
5.4	Examples of Truncated Values	162
5.5	Arithmetic Operators	164
5.6	Examples of Arithmetic Expression Evaluation	169
5.7	Arithmetic Compound Assignment Operators	172
5.8	Relational Operators	180
5.9	Primitive Data Value Equality Operators	181
5.10	Reference Equality Operators	182

5.11	Truth Values for Boolean Logical Operators	184
5.12	Boolean Logical Compound Assignment Operators	185
5.13	Conditional Operators	186
5.14	Truth Values for Conditional Operators	186
5.15	Integer Bitwise Operators	189
5.16	Result Table for Bitwise Operators	190
5.17	Examples of Bitwise Operations	190
5.18	Bitwise Compound Assignment Operators	192
6.1	The return Statement	225
7.1	Overriding versus Overloading	274
7.2	Same Signature for Subclass and Superclass Method	276
7.3	Types and Values	309
10.1	Summary of Arrays versus ArrayLists	426
10.2	Selected Functional Interfaces from the <code>java.util.function</code> Package	444
11.1	Selected Common Method Prefix of the Temporal Classes	463
11.2	Selected ISO-Based Predefined Formatters for Date and Time	489
11.3	Format Styles for Date and Time	490
11.4	Combination of Format Styles and Localized Formatters	491
11.5	Selected Date/Time Pattern Letters	496

Examples

1.1	Basic Elements of a Class Declaration	4
1.2	Static Members in Class Declaration	8
1.3	Defining a Subclass	11
1.4	An Application	16
1.5	Formatted Output	21
2.1	Default Values for Fields	42
2.2	Flagging Uninitialized Local Variables of Primitive Data Types	43
2.3	Flagging Uninitialized Local Reference Variables	44
3.1	Using the <code>this</code> Reference	51
3.2	Namespaces	54
3.3	Using Arrays	62
3.4	Using Anonymous Arrays	64
3.5	Using Multidimensional Arrays	67
3.6	Passing Primitive Values	74
3.7	Passing Reference Values	75
3.8	Passing Arrays	77
3.9	Array Elements as Primitive Data Values	79
3.10	Array Elements as Reference Values	79
3.11	Calling a Variable Arity Method	83
3.12	Passing Program Arguments	86
3.13	Using Enums	88
4.1	Defining Packages and Using Type Import	99
4.2	Single Static Import	102
4.3	Avoiding the Interface Constant Antipattern	102
4.4	Importing Enum Constants	103
4.5	Shadowing Static Import	104
4.6	Conflict in Importing Static Method with the Same Signature	105
4.7	Class Scope	116
4.8	Accessibility Modifiers for Classes and Interfaces	118
4.9	Abstract Classes	121
4.10	Public Accessibility of Members	125
4.11	Accessing Static Members	132
4.12	Using <code>final</code> Modifier	134

4.13	Synchronized Methods	137
5.1	Evaluation Order of Operands and Arguments	153
5.2	Numeric Promotion in Arithmetic Expressions	171
5.3	Short-Circuit Evaluation Involving Conditional Operators	187
5.4	Bitwise Operations	191
6.1	Fall-Through in a switch Statement	205
6.2	Using break in a switch Statement	206
6.3	Nested switch Statement	207
6.4	Strings in switch Statement	208
6.5	Enums in switch Statement	209
6.6	The break Statement	221
6.7	Labeled break Statement	222
6.8	continue Statement	223
6.9	Labeled continue Statement	224
6.10	The return Statement	225
6.11	Method Execution	230
6.12	The try-catch Construct	241
6.13	Exception Propagation	243
6.14	The try-catch-finally Construct	246
6.15	The try-finally Construct	247
6.16	The finally Clause and the return Statement	248
6.17	Throwing Exceptions	250
6.18	The throws Clause	252
7.1	Extending Classes: Inheritance and Accessibility	265
7.2	Overriding, Overloading, and Hiding	270
7.3	Using the super Keyword	277
7.4	Constructor Overloading	282
7.5	The this() Constructor Call	284
7.6	The super() Constructor Call	285
7.7	Implementing Interfaces	292
7.8	Default Methods in Interfaces	297
7.9	Default Methods and Multiple Inheritance	299
7.10	Static Methods in Interfaces	301
7.11	Constants in Interfaces	302
7.12	Inheriting Constants in Interfaces	304
7.13	Assigning and Passing Reference Values	312
7.14	Choosing the Most Specific Method (Simple Case)	317
7.15	Overloaded Method Resolution	318
7.16	The instanceof and Cast Operators	322
7.17	Using the instanceof Operator	323
7.18	Polymorphism and Dynamic Method Lookup	330
7.19	Implementing Data Structures by Inheritance and Aggregation	332
8.1	Methods in the Object Class	344
8.2	String Representation of Integers	353
8.3	String Construction and Equality	359
8.4	Reading Characters from a String	362

EXAMPLES

xxv

9.1	Garbage Collection Eligibility	388
9.2	Using Finalizers	392
9.3	Invoking Garbage Collection	394
9.4	Initializer Expression Order and Method Calls	402
9.5	Static Initializers and Forward References	403
9.6	Instance Initializers and Forward References	405
9.7	Object State Construction	406
9.8	Initialization Anomaly under Object State Construction	408
10.1	Using an ArrayList	427
10.2	Implementing Customized Methods for Filtering an ArrayList	434
10.3	Implementing an Interface for Filtering an ArrayList	436
10.4	User-Defined Functional Interface for Filtering an ArrayList	439
10.5	Using the Predicate<T> Functional Interface for Filtering an ArrayList	441
10.6	Accessing Members in an Enclosing Object	447
10.7	Accessing Local Variables in an Enclosing Method	449
10.8	Filtering an ArrayList	452
11.1	Creating Temporal Objects	467
11.2	Using Temporal Objects	472
11.3	Temporal Arithmetic	475
11.4	Period-Based Loop	481
11.5	More Temporal Arithmetic	482
11.6	Using Default Date and Time Formatters	488
11.7	Using Predefined Format Styles with Time-Based Values	492
11.8	Using Predefined Format Styles with Date-Based Values	493
11.9	Using Predefined Format Styles with Date and Time-Based Values	494
11.10	Formatting and Parsing with Letter Patterns	497
11.11	Formatting with Date/Time Letter Patterns	499