



# Beginning Java 9 Fundamentals

Arrays, Objects, Modules, JShell,  
and Regular Expressions

—

*Second Edition*

—

Kishori Sharan

Apress®

# Beginning Java 9 Fundamentals

Arrays, Objects, Modules, JShell,  
and Regular Expressions

Second Edition



Kishori Sharan

Apress®

***Beginning Java 9 Fundamentals: Arrays, Objects, Modules, JShell, and Regular Expressions***

Kishori Sharan  
Montgomery, Alabama, USA

ISBN-13 (pbk): 978-1-4842-2843-2  
<https://doi.org/10.1007/978-1-4842-2902-6>

ISBN-13 (electronic): 978-1-4842-2902-6

Library of Congress Control Number: 2017958824

Copyright © 2017 by Kishori Sharan

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Cover image by Freepik ([www.freepik.com](http://www.freepik.com))

Managing Director: Welmoed Spahr  
Editorial Director: Todd Green  
Acquisitions Editor: Steve Anglin  
Development Editor: Matthew Moodie  
Technical Reviewer: Wallace Jackson  
Coordinating Editor: Mark Powers  
Copy Editor: Kezia Endsley

Distributed to the book trade worldwide by Springer Science+Business Media New York, 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail [orders-ny@springer-sbm.com](mailto:orders-ny@springer-sbm.com), or visit [www.springeronline.com](http://www.springeronline.com). Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail [rights@apress.com](mailto:rights@apress.com), or visit <http://www.apress.com/rights-permissions>.

Apress titles may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Print and eBook Bulk Sales web page at <http://www.apress.com/bulk-sales>.

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub via the book's product page, located at [www.apress.com/9781484228432](http://www.apress.com/9781484228432). For more detailed information, please visit <http://www.apress.com/source-code>.

Printed on acid-free paper

# Contents at a Glance

<b>About the Author .....</b>	<b>xxvii</b>
<b>About the Technical Reviewer .....</b>	<b>xxix</b>
<b>Acknowledgments .....</b>	<b>xxxi</b>
<b>Introduction .....</b>	<b>xxxiii</b>
<b>■ Chapter 1: Programming Concepts .....</b>	<b>1</b>
<b>■ Chapter 2: Setting Up the Environment .....</b>	<b>33</b>
<b>■ Chapter 3: Writing Java Programs .....</b>	<b>45</b>
<b>■ Chapter 4: Data Types.....</b>	<b>99</b>
<b>■ Chapter 5: Operators .....</b>	<b>145</b>
<b>■ Chapter 6: Statements.....</b>	<b>191</b>
<b>■ Chapter 7: Classes.....</b>	<b>223</b>
<b>■ Chapter 8: Methods .....</b>	<b>255</b>
<b>■ Chapter 9: Constructors .....</b>	<b>323</b>
<b>■ Chapter 10: Modules .....</b>	<b>355</b>
<b>■ Chapter 11: Object and Objects Classes.....</b>	<b>395</b>
<b>■ Chapter 12: Wrapper Classes .....</b>	<b>439</b>
<b>■ Chapter 13: Exception Handling .....</b>	<b>461</b>
<b>■ Chapter 14: Assertions .....</b>	<b>511</b>
<b>■ Chapter 15: Strings.....</b>	<b>521</b>

■ <b>Chapter 16: Dates and Times</b> .....	<b>549</b>
■ <b>Chapter 17: Formatting Data</b> .....	<b>631</b>
■ <b>Chapter 18: Regular Expressions</b> .....	<b>669</b>
■ <b>Chapter 19: Arrays</b> .....	<b>701</b>
■ <b>Chapter 20: Inheritance</b> .....	<b>755</b>
■ <b>Chapter 21: Interfaces</b> .....	<b>823</b>
■ <b>Chapter 22: Enum Types</b> .....	<b>895</b>
■ <b>Chapter 23: Java Shell</b> .....	<b>921</b>
■ <b>Appendix A: Character Encodings</b> .....	<b>985</b>
■ <b>Appendix B: Documentation Comments</b> .....	<b>997</b>
<b>Index</b> .....	<b>1023</b>

# Contents

<b>About the Author .....</b>	<b>xxvii</b>
<b>About the Technical Reviewer .....</b>	<b>xxix</b>
<b>Acknowledgments .....</b>	<b>xxx</b>
<b>Introduction .....</b>	<b>xxiii</b>
<b>■ Chapter 1: Programming Concepts .....</b>	<b>1</b>
What Is Programming?.....	1
Components of a Programming Language.....	4
Programming Paradigms.....	4
Imperative Paradigm .....	6
Procedural Paradigm .....	6
Declarative Paradigm .....	7
Functional Paradigm.....	8
Logic Paradigm.....	8
Object-Oriented Paradigm .....	9
What Is Java? .....	12
The Object-Oriented Paradigm and Java.....	13
Abstraction .....	14
Encapsulation and Information Hiding.....	23
Inheritance.....	25
Polymorphism.....	26
Summary .....	31
<b>■ Chapter 2: Setting Up the Environment .....</b>	<b>33</b>
System Requirements .....	33
Installing JDK 9 .....	33

The JDK Directory Structure.....	34
Verifying the JDK Installation .....	37
Starting the JShell Tool .....	38
Installing NetBeans 9 .....	38
Configuring NetBeans .....	39
Summary.....	43
<b>■ Chapter 3: Writing Java Programs .....</b>	<b>45</b>
The Goal Statement.....	45
Using the JShell Tool .....	46
What Is a Java Program? .....	46
Writing the Source Code.....	47
Writing Comments.....	48
Declaring a Module .....	49
Declaring Types .....	51
Package Declaration.....	52
Import Declarations .....	53
Class Declaration.....	54
Types Have Two Names .....	59
Compiling the Source Code .....	60
Packaging the Compiled Code.....	62
Running a Java Program .....	64
Playing with Module Options.....	69
Listing Observable Modules .....	69
Limiting the Observable Modules .....	70
Describing a Module.....	71
Printing Module Resolution Details.....	72
Dry Running Your Program .....	73
Enhancing a Module Descriptor .....	73
Running Java Programs in Legacy Mode .....	75

Duplicate Modules on Module Path.....	78
Syntax for Command-Line Options.....	80
Writing Java Programs Using the NetBeans IDE .....	81
Creating a Java Project.....	81
Creating Modular JARs in NetBeans.....	88
NetBeans Project Directory Structure.....	89
Adding Classes to a Module .....	89
Customizing NetBeans Project Properties .....	89
Opening an Existing NetBeans Project .....	91
Behind the Scenes .....	91
Summary.....	95
■ <b>Chapter 4: Data Types</b> .....	<b>99</b>
What Is a Data Type? .....	99
What Is an Identifier? .....	100
Keywords.....	102
Data Types in Java.....	102
Primitive Data Types in Java .....	107
Integral Data Types .....	108
Floating-Point Data Types.....	118
Underscores in Numeric Literals .....	123
Java Compiler and Unicode Escape Sequence .....	123
A Short Break .....	125
Binary Representation of Integers.....	126
Diminished Radix Complement.....	127
Radix Complement .....	128
Binary Representation of Floating-Point Numbers.....	129
32-bit Single-Precision Floating-Point Format.....	131
Special Floating-Point Numbers.....	134
Signed Zeros.....	134
Signed Infinities.....	134



NaN.....	135
Denormals .....	136
<b>Rounding Modes .....</b>	<b>136</b>
Rounding Toward Zero .....	137
Rounding Toward Positive Infinity.....	137
Rounding Toward Negative Infinity .....	137
Rounding Toward Nearest.....	138
<b>IEEE Floating-Point Exceptions.....</b>	<b>138</b>
Division by Zero Exception.....	138
Invalid Operation Exception .....	138
Overflow Exception.....	139
Underflow Exception.....	139
Inexact Exception .....	139
<b>Java and IEEE Floating-Point Standards .....</b>	<b>140</b>
<b>Little-Endian and Big-Endian.....</b>	<b>140</b>
<b>Summary.....</b>	<b>141</b>
<b>■ Chapter 5: Operators .....</b>	<b>145</b>
<b>What Is an Operator?.....</b>	<b>145</b>
<b>Assignment Operator.....</b>	<b>147</b>
<b>Declaration, Initialization, and Assignment .....</b>	<b>149</b>
<b>Arithmetic Operators .....</b>	<b>150</b>
Addition Operator (+).....	151
Subtraction Operator (-).....	153
Multiplication Operator (*) .....	154
Division Operator (/).....	155
Modulus Operator (%).....	157
Unary Plus Operator (+).....	159
Unary Minus Operator (-).....	159
Compound Arithmetic Assignment Operators.....	160
Increment (++) and Decrement (--) Operators .....	161

String Concatenation Operator (+) .....	164
Relational Operators.....	169
Equality Operator (==).....	169
Inequality Operator (!=) .....	172
Greater Than Operator (>).....	172
Greater Than or Equal to Operator (>=).....	173
Less Than Operator (<).....	173
Less Than or Equal to Operator (<=) .....	174
Boolean Logical Operators .....	174
Logical NOT Operator (!) .....	175
Logical Short-Circuit AND Operator (&&) .....	175
Logical AND Operator (&).....	177
Logical Short-Circuit OR Operator (  ) .....	178
Logical OR Operator ( ).....	178
Logical XOR Operator (^).....	178
Compound Boolean Logical Assignment Operators.....	179
Ternary Operator (? :) .....	180
Bitwise Operators.....	180
Operators Precedence.....	184
Summary.....	186
<b>■ Chapter 6: Statements .....</b>	<b>191</b>
What Is a Statement? .....	191
Types of Statements.....	192
Declaration Statement.....	192
Expression Statement.....	192
Control Flow Statement.....	193
A Block Statement.....	194
The if-else Statement.....	195
The switch Statement .....	200

The for Statement .....	204
Initialization .....	205
Condition-Expression.....	206
Expression-List.....	207
The for-each Statement.....	209
The while Statement .....	210
The do-while Statement.....	212
The break Statement.....	214
The continue Statement .....	217
An Empty Statement.....	218
Summary.....	219
<b>■ Chapter 7: Classes.....</b>	<b>223</b>
What Is a Class? .....	223
Declaring a Class.....	224
Declaring Fields in a Class .....	225
Creating Instances of a Class .....	226
The null Reference Type .....	228
Using Dot Notation to Access Fields of a Class .....	229
Default Initialization of Fields.....	232
Access Level Modifiers for a Class.....	233
Import Declarations .....	236
Single-Type Import Declaration .....	237
Import-on-Demand Declaration.....	239
Import Declarations and Type Search Order .....	241
Automatic Import Declarations.....	247
Static Import Declarations .....	248
Summary.....	251

<b>■ Chapter 8: Methods</b> .....	<b>255</b>
What Is a Method? .....	255
Declaring Methods of a Class.....	255
Local Variables .....	260
Rule #1 .....	260
Rule #2 .....	261
Rule #3 .....	261
Rule #4 .....	261
Instance Methods and Class Methods.....	264
Invoking a Method.....	265
The Special main() Method .....	267
What Is this?.....	269
Access Levels for Class Members.....	275
Access Level: A Case Study.....	282
What Is a Var-Args Method? .....	288
Overloading a Var-Args Method .....	293
Var-Args Methods and the main() Method .....	294
Parameter-Passing Mechanisms.....	295
Pass By Value .....	296
Pass By Constant Value .....	299
Pass By Reference.....	299
Pass By Reference Value .....	303
Pass by Constant Reference Value .....	304
Pass by Result .....	304
Pass by Value Result.....	304
Pass By Name.....	305
Pass by Need.....	306
Parameter-Passing Mechanisms in Java .....	306
Summary.....	318

<b>■ Chapter 9: Constructors</b> .....	<b>323</b>
What Is a Constructor? .....	323
Declaring a Constructor.....	323
Overloading a Constructor.....	326
Writing Code for a Constructor .....	327
Calling a Constructor from Another Constructor .....	330
Using a return Statement Inside a Constructor .....	332
Access Level Modifier for a Constructor .....	333
Default Constructor .....	337
A static Constructor.....	338
Instance Initialization Block .....	338
static Initialization Block .....	339
The final Keyword.....	341
final Local Variables.....	342
final Parameters .....	343
final Instance Variables.....	343
final Class Variables.....	346
final Reference Variables.....	346
Compile-Time vs. Runtime final Variables .....	347
Generic Classes.....	347
Summary.....	350
<b>■ Chapter 10: Modules</b> .....	<b>355</b>
What Is a Module?.....	355
Declaring Modules .....	356
Declaring Module Dependence .....	358
An Example of Module Dependence.....	360
Troubleshooting .....	366
Implicit Dependence.....	368
Optional Dependence .....	373

Opening Modules and Packages .....	373
Open Modules.....	375
Opening Packages.....	375
Splitting Packages Across Modules .....	376
Restrictions in Module Declarations.....	377
Types of Modules .....	377
Normal Modules .....	379
Open Modules.....	379
Automatic Modules.....	379
Unnamed Modules.....	383
Aggregator Modules.....	384
Knowing about Modules at Runtime .....	384
Migration Path to JDK 9 .....	386
Disassembling Module Definitions .....	388
Summary.....	391
<b>■ Chapter 11: Object and Objects Classes.....</b>	<b>395</b>
The Object Class.....	395
Rule #1 .....	396
Rule #2 .....	397
What Is the Class of an Object?.....	399
Computing the Hash Code of an Object.....	400
Comparing Objects for Equality.....	404
String Representation of an Object .....	410
Cloning Objects .....	414
Finalizing an Object.....	422
Immutable Objects .....	424
The Objects Class.....	429
Bounds Checks.....	430
Comparing Objects .....	430

Computing Hash Code .....	431
Checking for Null .....	432
Validating Arguments.....	432
Obtaining String Representation of Objects .....	433
Using the Objects Class .....	433
Summary .....	435
<b>■ Chapter 12: Wrapper Classes .....</b>	<b>439</b>
Wrapper Classes .....	439
Numeric Wrapper Classes .....	442
The Character Wrapper Class .....	445
The Boolean Wrapper Class.....	447
Unsigned Numeric Operations.....	447
Autoboxing and Unboxing.....	449
Beware of Null Values .....	452
Overloaded Methods and Autoboxing/Unboxing.....	452
Comparison Operators and Autoboxing/Unboxing.....	455
Collections and Autoboxing/Unboxing .....	457
Summary.....	458
<b>■ Chapter 13: Exception Handling .....</b>	<b>461</b>
What Is an Exception? .....	461
An Exception Is an Object.....	464
Using a try-catch Block.....	464
Transfer of Control.....	467
Exception Class Hierarchy.....	469
Arranging Multiple catch Blocks .....	470
A Multi-catch Block .....	473
Checked and Unchecked Exceptions.....	474
Checked Exception: Catch or Declare.....	477
Checked Exceptions and Initializers.....	484

Throwing an Exception.....	485
Creating an Exception Class.....	486
The finally Block.....	490
Rethrowing an Exception .....	494
Analysis of Rethrown Exceptions .....	497
Throwing Too Many Exceptions.....	497
Accessing the Stack of a Thread.....	499
The try-with-resources Block.....	502
Summary.....	509
<b>■ Chapter 14: Assertions .....</b>	<b>511</b>
What Is an Assertion?.....	511
Testing Assertions .....	513
Enabling/Disabling Assertions.....	515
Using Assertions.....	517
Checking for Assertion Status .....	518
Summary.....	519
<b>■ Chapter 15: Strings.....</b>	<b>521</b>
What Is a String? .....	521
String Literals.....	522
Escape Sequence Characters in String Literals .....	522
Unicode Escapes in String Literals.....	523
What Is a CharSequence? .....	523
Creating String Objects .....	523
Length of a String.....	524
String Literals Are String Objects .....	524
String Objects Are Immutable .....	525
Comparing Strings.....	526
String Pool.....	528



<b>String Operations</b> .....	<b>530</b>
Getting the Character at an Index.....	530
Testing Strings for Equality.....	531
Testing a String to be Empty.....	531
Changing the Case.....	532
Searching for a String.....	532
Representing Values as Strings.....	532
Getting a Substring.....	533
Trimming a String.....	533
Replacing Part of a String.....	533
Matching the Start and End of a String .....	534
<b>Splitting and Joining Strings</b> .....	<b>535</b>
<b>Strings in a switch Statement</b> .....	<b>536</b>
<b>Testing a String for Palindrome</b> .....	<b>538</b>
<b>StringBuilder and StringBuffer</b> .....	<b>539</b>
<b>String Concatenation Operator (+)</b> .....	<b>543</b>
<b>Language-Sensitive String Comparison</b> .....	<b>543</b>
<b>Summary</b> .....	<b>544</b>
<b>■ Chapter 16: Dates and Times</b> .....	<b>549</b>
The Date-Time API.....	549
Design Principles.....	550
A Quick Example .....	551
Evolution of Timekeeping .....	552
Time Zones and Daylight Savings Time.....	555
Calendar Systems .....	556
The Julian Calendar.....	556
The Gregorian Calendar .....	557
ISO-8601 Standards for Datetime .....	558
Exploring the New Date-Time API .....	560
The ofXxx() Methods .....	560

The from() Methods .....	560
The withXxx() Methods .....	561
The getXxx() Methods .....	561
The toXxx() Methods .....	561
The atXxx() Methods .....	562
The plusXxx() and minusXxx() Methods.....	562
The multipliedBy(), dividedBy(), and negated() Methods .....	562
<b>Instants and Durations .....</b>	<b>563</b>
Dividing a Duration by Another Duration.....	566
Converting and Retrieving Duration Parts .....	566
Truncating Duration .....	567
<b>Human-Scale Time .....</b>	<b>568</b>
The ZoneOffset Class.....	568
The ZonedDateTime Class .....	570
Useful Datetime-Related Enums.....	572
Local Date, Time, and Datetime .....	577
Offset Time and Datetime .....	582
Zoned Datetime .....	583
<b>Same Instant, Different Times .....</b>	<b>587</b>
<b>Clocks.....</b>	<b>587</b>
<b>Periods .....</b>	<b>589</b>
<b>Period Between Two Dates and Times .....</b>	<b>591</b>
<b>Partials .....</b>	<b>593</b>
<b>Adjusting Dates .....</b>	<b>595</b>
<b>Querying Datetime Objects.....</b>	<b>600</b>
<b>Non-ISO Calendar Systems .....</b>	<b>605</b>
<b>Formatting Dates and Times .....</b>	<b>607</b>
Using Predefined Formatters.....	607
Using the format() Method of Datetime Classes.....	609
Using User-Defined Patterns .....	610

Using Locale Specific Formats .....	615
Using the DateTimeFormatterBuilder Class.....	617
<b>Parsing Dates and Times.....</b>	<b>618</b>
<b>Legacy Datetime Classes .....</b>	<b>621</b>
The Date Class.....	621
The Calendar Class.....	622
The add() Method.....	623
The roll() Method .....	624
<b>Interoperability with Legacy Datetime Classes .....</b>	<b>625</b>
<b>Summary.....</b>	<b>629</b>
<b>■ Chapter 17: Formatting Data .....</b>	<b>631</b>
<b>Formatting Dates.....</b>	<b>631</b>
Using Predefined Date Formats.....	632
Using Custom Date Formats.....	635
Parsing Dates .....	637
<b>Formatting Numbers .....</b>	<b>639</b>
Using Predefined Number Formats .....	640
Using Custom Number Formats.....	641
Parsing Numbers .....	642
<b>printf-Style Formatting.....</b>	<b>643</b>
The Big Picture .....	643
The Details.....	646
Referencing an Argument Inside a Format Specifier.....	648
Using Flags in a Format Specifier.....	652
Conversion Characters .....	653
<b>Summary.....</b>	<b>667</b>
<b>■ Chapter 18: Regular Expressions .....</b>	<b>669</b>
<b>What Is a Regular Expression?.....</b>	<b>669</b>
<b>Metacharacters .....</b>	<b>672</b>
<b>Character Classes .....</b>	<b>673</b>
Predefined Character Classes .....	674

More Powers to Regular Expressions.....	674
Compiling Regular Expressions .....	674
Creating a Matcher .....	676
Matching the Pattern.....	676
Querying a Match .....	679
Beware of Backslashes.....	679
Quantifiers in Regular Expressions .....	680
Matching Boundaries .....	681
Groups and Back Referencing.....	682
Using Named Groups.....	688
Resetting the Matcher .....	690
Final Words on E-Mail Validations .....	691
Find-and-Replace Using Regular Expressions .....	691
Streams of Matched Results .....	695
Summary.....	697
<b>■ Chapter 19: Arrays .....</b>	<b>701</b>
What Is an Array? .....	701
Arrays Are Objects.....	703
Accessing Array Elements.....	704
Length of an Array .....	705
Initializing Array Elements.....	706
Beware of Reference Type Arrays.....	708
Explicit Array Initialization .....	709
Limitations of Using Arrays .....	710
Simulating Variable-Length Arrays.....	714
Passing an Array as a Parameter .....	717
Array Parameter Reference .....	722
Elements of the Array Parameter.....	723
The Object Referred by the Array Parameter Elements .....	724

Command-Line Arguments.....	726
Multi-Dimensional Arrays.....	730
Accessing Elements of a Multi-Dimensional Array.....	734
Initializing Multi-Dimensional Arrays.....	734
Enhanced for Loop for Arrays.....	735
Array Declaration Syntax.....	736
Runtime Array Bounds Checks.....	737
What Is the Class of an Array Object? .....	738
Array Assignment Compatibility .....	740
Converting an ArrayList/Vector to an Array .....	742
Performing Array Operations.....	743
Converting Arrays to Another Type.....	745
Searching an Array .....	746
Comparing Arrays.....	746
Copying Arrays.....	748
Filling Arrays.....	748
Computing Hash Code .....	749
Performing Parallel Accumulation .....	749
Sorting Arrays.....	750
Summary.....	750
<b>■ Chapter 20: Inheritance.....</b>	<b>755</b>
What Is Inheritance? .....	755
Inheriting Classes.....	756
The Object Class Is the Default Superclass.....	759
Inheritance and Hierarchical Relationship .....	759
What Is Inherited by a Subclass? .....	760
Upcasting and Downcasting.....	762
The instanceof Operator.....	766

Binding .....	768
Early Binding .....	769
Late Binding.....	772
Method Overriding.....	775
Method Overriding Rule #1.....	777
Method Overriding Rule #2.....	777
Method Overriding Rule #3.....	777
Method Overriding Rule #4.....	777
Method Overriding Rule #5.....	778
Method Overriding Rule #6.....	779
Accessing Overridden Method .....	782
Method Overloading .....	784
Inheritance and Constructors.....	788
Method Hiding .....	796
Field Hiding .....	798
Disabling Inheritance .....	802
Abstract Classes and Methods.....	803
Method Overriding and Generic Method Signatures .....	811
Typo Danger in Method Overriding.....	813
Is-a, has-a, and part-of Relationships.....	814
No Multiple Inheritance of Classes.....	817
Summary.....	817
<b>■ Chapter 21: Interfaces.....</b>	<b>823</b>
What Is an Interface? .....	823
Proposed Solution #1.....	826
Proposed Solution #2.....	827
Proposed Solution #3.....	828
An Ideal Solution.....	828

Declaring an Interface.....	833
Declaring Interface Members.....	834
Constant Fields Declarations.....	834
Methods Declarations.....	836
Nested Type Declarations.....	846
An Interface Defines a New Type.....	848
Implementing an Interface.....	851
Implementing Interface Methods.....	855
Implementing Multiple Interfaces.....	858
Implementing an Interface Partially.....	861
The Supertype-Subtype Relationship.....	863
Interface Inheritance.....	864
The Superinterface-Subinterface Relationship.....	870
Inheriting Conflicting Implementations.....	870
The Superclass Always Wins.....	871
The Most Specific Superinterface Wins.....	873
The Class Must Override the Conflicting Method.....	874
The instanceof Operator.....	875
Marker Interfaces.....	879
Functional Interfaces.....	880
Comparing Objects.....	880
Using the Comparable Interface.....	880
Using the Comparator Interface.....	883
Polymorphism—One Object, Many Views.....	887
Dynamic Binding and Interfaces.....	889
Summary.....	890
■ <b>Chapter 22: Enum Types.....</b>	<b>895</b>
What Is an Enum Type?.....	895
The Superclass of an Enum Type.....	899

Using Enum Types in switch Statements .....	903
Associating Data and Methods to Enum Constants.....	903
Associating a Body to an Enum Constant.....	905
Comparing Two Enum Constants.....	910
Nested Enum Types .....	911
Implementing an Interface to an Enum Type.....	913
Reverse Lookup for Enum Constants .....	914
Range of Enum Constants .....	914
Summary.....	916
<b>■ Chapter 23: Java Shell.....</b>	<b>921</b>
What Is the Java Shell? .....	922
The JShell Architecture .....	923
Starting the JShell Tool .....	924
Exiting the JShell Tool .....	927
What Are Snippets and Commands? .....	927
Evaluating Expressions .....	929
Listing Snippets.....	931
Editing Snippets .....	935
Rerunning Previous Snippets .....	937
Declaring Variables.....	937
Import Statements.....	940
Method Declarations .....	944
Type Declarations .....	945
Setting the Execution Environment .....	948
No Checked Exceptions.....	950
Auto-Completion .....	950
Snippets and Commands History .....	954
Reading JShell Stack Trace .....	955



Reusing JShell Sessions .....	956
Resetting the JShell State.....	958
Reloading the JShell State .....	958
Configuring JShell .....	961
Setting the Snippet Editor.....	961
Setting Feedback Mode.....	962
Creating Custom feedback Modes.....	965
Setting Up Startup Snippets .....	969
Using JShell Documentation .....	972
The JShell API.....	974
Creating a JShell .....	975
Working with Snippets.....	976
Handling Snippet Events.....	978
An Example.....	978
Summary .....	982
<b>■ Appendix A: Character Encodings .....</b>	<b>985</b>
ASCII.....	986
8-Bit Character Sets.....	990
Universal Multiple-Octet Coded Character Set (UCS) .....	991
UCS-2.....	992
UCS-4.....	992
UTF-16 (UCS Transformation Format 16).....	992
UTF-8 (UCS Transformation Format 8).....	993
Java and Character Encodings.....	994
<b>■ Appendix B: Documentation Comments .....</b>	<b>997</b>
Writing Documentation Comments .....	998
List of Block and Inline Tags.....	1000
@author <author-name(s)> .....	1001
@deprecated <explanation-text> .....	1001
@exception <class-name> <description> .....	1001
@param <parameter-name> <description> .....	1002

@return <description>.....	1002
@see <reference> .....	1002
@serial <field-description or include/exclude>.....	1003
@serialData <data-description>.....	1004
@serialField <field-name> <field-type> <field-description>.....	1004
@since <description> .....	1005
@throws <class-name> <description>.....	1005
@version <version-text> .....	1006
{@code <text>} .....	1006
{@docRoot}.....	1006
{@inheritDoc}.....	1006
{@link <package.class#member> <label>} .....	1008
{@linkplain <package.class#member> <label>}.....	1008
{@literal <text>}.....	1008
{@value <package.class#field>}.....	1008
@hidden .....	1009
{@index <keyword> <description>}.....	1009
@provides <service-type> <description>.....	1010
@uses <service-type> <description>.....	1010
Documenting Packages.....	1010
com/jdojo/utility/package-info.java file .....	1011
com/jdojo/utility/package.html file .....	1011
Overview Documentation .....	1012
Including Unprocessed Files in Documentation .....	1012
Skipping Source Files Processing .....	1012
An Example of Documentation Comments .....	1012
Running the javadoc Tool .....	1015
Generated Documentation Files .....	1017
Viewing Generated HTML Documentation.....	1017
Searching Javadoc .....	1019
Summary.....	1020
<b>Index.....</b>	<b>1023</b>

# About the Author



**Kishori Sharan** works as a senior software engineer lead at IndraSoft, Inc. He earned a master's of science degree in computer information systems from Troy State University, Alabama. He is a Sun-certified Java 2 programmer and has over 20 years of experience in developing enterprise applications and providing training to professional developers using the Java platform.

# About the Technical Reviewer

**Wallace Jackson** has been writing for leading multimedia publications about his work in new media content development since the advent of *Multimedia Producer Magazine* nearly two decades ago. He has authored a half-dozen Android book titles for Apress, including four titles in the popular Pro Android series. Wallace received his undergraduate degree in business economics from the University of California at Los Angeles and a graduate degree in MIS design and implementation from the University of Southern California. He is currently the CEO of Mind Taffy Design, a new media content production and digital campaign design and development agency.

# Acknowledgments

I would like to thank my family members and friends for their encouragement and support: my mom Pratima Devi; my elder brothers, Janki Sharan and Dr. Sita Sharan; my nephews, Gaurav and Saurav; my sister Ratna; my friends Karthikeya Venkatesan, Rahul Nagpal, Ravi Datla, Mahbub Choudhury, Richard Castillo; and many more friends not mentioned here.

My wife, Ellen, was always patient when I spent long hours at my computer desk working on this book. I want to thank her for all of her support in writing this book.

My special thanks to my friend Preethi Vasudev for offering her valuable time for providing solutions to the exercises in this book. She likes programming challenges—particularly Google Code Jam. I bet she enjoyed solving the exercises in each chapter of this book.

My sincere thanks are due to the wonderful team at Apress for their support during the publication of this book. Thanks to Mark Powers, the Editorial Operations Manager, for providing excellent support. Last but not least, my sincere thanks to Steve Anglin, the Lead Editor at Apress, for taking the initiative for the publication of this book.

# Introduction

## How This Book Came About

My first encounter with the Java programming language was during a one-week Java training session in 1997. I did not get a chance to use Java in a project until 1999. I read two Java books and took a Java 2 Programmer certification examination. I did very well on the test, scoring 95 percent. The three questions that I missed on the test made me realize that the books that I had read did not adequately cover details of all the topics necessary about Java. I made up my mind to write a book on the Java programming language. So, I formulated a plan to cover most of the topics that a Java developer needs to use the Java programming language effectively in a project, as well as to get a certification. I initially planned to cover all essential topics in Java in 700 to 800 pages.

As I progressed, I realized that a book covering most of the Java topics in detail could not be written in 700 to 800 pages. One chapter alone that covered data types, operators, and statements spanned 90 pages. I was then faced with the question, “Should I shorten the content of the book or include all the details that I think a Java developer needs?” I opted for including all the details in the book, rather than shortening its content to keep the number of pages low. It has never been my intent to make lots of money from this book. I was never in a hurry to finish this book because that rush could have compromised the quality and the coverage of its contents. In short, I wrote this book to help the Java community understand and use the Java programming language effectively, without having to read many books on the same subject. I wrote this book with the plan that it would be a comprehensive one-stop reference for everyone who wants to learn and grasp the intricacies of the Java programming language.

One of my high school teachers used to tell us that if one wanted to understand a building, one must first understand the bricks, steel, and mortar that make up the building. The same logic applies to most of the things that we want to understand in our lives. It certainly applies to an understanding of the Java programming language. If you want to master the Java programming language, you must start by understanding its basic building blocks. I have used this approach throughout this book, endeavoring to build each topic by describing the basics first. In the book, you will rarely find a topic described without first learning its background. Wherever possible, I have tried to correlate the programming practices with activities in our daily life. Most of the books about the Java programming language available in the market either do not include any pictures at all or have only a few. I believe in the adage, “A picture is worth a thousand words.” To a reader, a picture makes a topic easier to understand and remember. I have included plenty of illustrations in the book to aid readers in understanding and visualizing the contents. Developers who have little or no programming experience have difficulty in putting things together to make it a complete program. Keeping them in mind, the book contains over 290 complete Java programs that are ready to be compiled and run.

I spent countless hours doing research for writing this book. My main source of research was the Java Language Specification, whitepapers, and articles on Java topics, and Java Specification Requests (JSRs). I also spent quite a bit of time reading the Java source code to learn more about some of the Java topics. Sometimes, it took a few months researching a topic before I could write the first sentence on the topic. Finally, it was always fun to play with Java programs, sometimes for hours, to add them to the book.