

# The Complete Reference

Schildt's classic  
Java reference,  
updated and  
expanded for  
Java SE 6

# Java<sup>TM</sup> Seventh Edition

- ▲ Comprehensive guide to the entire Java language
- ▲ Includes coverage of applets, servlets, Swing, JavaBeans, the AWT, and collections
- ▲ Hundreds of examples and sample applications



**Herbert Schildt**

Top-selling programming  
author with more than  
3.5 million books sold  
worldwide

**Mc  
Graw  
Hill** Osborne

**Java<sup>™</sup> :**  
**The Complete Reference,**  
**Seventh Edition**

## About the Author

**Herbert Schildt** is a leading authority on the Java, C, C++, and C# languages, and is a master Windows programmer. His programming books have sold more than 3.5 million copies worldwide and have been translated into all major foreign languages. He is the author of the best-selling *The Art of Java*, *Java: A Beginner's Guide*, and *Swing: A Beginner's Guide*. Among his other bestsellers are *C++: The Complete Reference*, *C++: A Beginner's Guide*, *C#: The Complete Reference*, and *C#: A Beginner's Guide*. Schildt holds both graduate and undergraduate degrees from the University of Illinois. He can be reached at his consulting office at (217) 586-4683. His Web site is **[www.HerbSchildt.com](http://www.HerbSchildt.com)**.

# **Java<sup>TM</sup> :** **The Complete Reference,** **Seventh Edition**

Herbert Schildt



New York Chicago San Francisco  
Lisbon London Madrid Mexico City  
Milan New Delhi San Juan  
Seoul Singapore Sydney Toronto

Copyright © 2007 by The McGraw-Hill Companies. All rights reserved. Except as permitted under the United States Copyright Act of 1976, no part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written permission of the publisher.

ISBN: 978-0-07-163177-8

MHID: 0-07-163177-1

The material in this eBook also appears in the print version of this title: ISBN: 978-0-07-226385-5,  
MHID: 0-07-226385-7.

All trademarks are trademarks of their respective owners. Rather than put a trademark symbol after every occurrence of a trademarked name, we use names in an editorial fashion only, and to the benefit of the trademark owner, with no intention of infringement of the trademark. Where such designations appear in this book, they have been printed with initial caps.

McGraw-Hill eBooks are available at special quantity discounts to use as premiums and sales promotions, or for use in corporate training programs. To contact a representative please e-mail us at [bulksales@mcgraw-hill.com](mailto:bulksales@mcgraw-hill.com).

Information has been obtained by McGraw-Hill from sources believed to be reliable. However, because of the possibility of human or mechanical error by our sources, McGraw-Hill, or others, McGraw-Hill does not guarantee the accuracy, adequacy, or completeness of any information and is not responsible for any errors or omissions or the results obtained from the use of such information.

#### TERMS OF USE

This is a copyrighted work and The McGraw-Hill Companies, Inc. ("McGrawHill") and its licensors reserve all rights in and to the work. Use of this work is subject to these terms. Except as permitted under the Copyright Act of 1976 and the right to store and retrieve one copy of the work, you may not decompile, disassemble, reverse engineer, reproduce, modify, create derivative works based upon, transmit, distribute, disseminate, sell, publish or sublicense the work or any part of it without McGraw-Hill's prior consent. You may use the work for your own noncommercial and personal use; any other use of the work is strictly prohibited. Your right to use the work may be terminated if you fail to comply with these terms.

THE WORK IS PROVIDED "AS IS." McGRAW-HILL AND ITS LICENSORS MAKE NO GUARANTEES OR WARRANTIES AS TO THE ACCURACY, ADEQUACY OR COMPLETENESS OF OR RESULTS TO BE OBTAINED FROM USING THE WORK, INCLUDING ANY INFORMATION THAT CAN BE ACCESSED THROUGH THE WORK VIA HYPERLINK OR OTHERWISE, AND EXPRESSLY DISCLAIM ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. McGraw-Hill and its licensors do not warrant or guarantee that the functions contained in the work will meet your requirements or that its operation will be uninterrupted or error free. Neither McGraw-Hill nor its licensors shall be liable to you or anyone else for any inaccuracy, error or omission, regardless of cause, in the work or for any damages resulting therefrom. McGraw-Hill has no responsibility for the content of any information accessed through the work. Under no circumstances shall McGraw-Hill and/or its licensors be liable for any indirect, incidental, special, punitive, consequential or similar damages that result from the use of or inability to use the work, even if any of them has been advised of the possibility of such damages. This limitation of liability shall apply to any claim or cause whatsoever whether such claim or cause arises in contract, tort or otherwise.

---

# Contents at a Glance

---

## Part I The Java Language

<b>1</b>	The History and Evolution of Java .....	3
<b>2</b>	An Overview of Java .....	15
<b>3</b>	Data Types, Variables, and Arrays .....	33
<b>4</b>	Operators .....	57
<b>5</b>	Control Statements .....	77
<b>6</b>	Introducing Classes .....	105
<b>7</b>	A Closer Look at Methods and Classes .....	125
<b>8</b>	Inheritance .....	157
<b>9</b>	Packages and Interfaces .....	183
<b>10</b>	Exception Handling .....	205
<b>11</b>	Multithreaded Programming .....	223
<b>12</b>	Enumerations, Autoboxing, and Annotations (Metadata) .....	255
<b>13</b>	I/O, Applets, and Other Topics .....	285
<b>14</b>	Generics .....	315

---

## Part II The Java Library

<b>15</b>	String Handling .....	359
<b>16</b>	Exploring java.lang .....	385
<b>17</b>	java.util Part 1: The Collections Framework .....	437
<b>18</b>	java.util Part 2: More Utility Classes .....	503
<b>19</b>	Input/Output: Exploring java.io .....	555
<b>20</b>	Networking .....	599
<b>21</b>	The Applet Class .....	617
<b>22</b>	Event Handling .....	637
<b>23</b>	Introducing the AWT: Working with Windows, Graphics, and Text ..	663
<b>24</b>	Using AWT Controls, Layout Managers, and Menus .....	701
<b>25</b>	Images .....	755
<b>26</b>	The Concurrency Utilities .....	787
<b>27</b>	NIO, Regular Expressions, and Other Packages .....	813

---

**Part III   Software Development Using Java**

<b>28</b>	Java Beans .....	847
<b>29</b>	Introducing Swing .....	859
<b>30</b>	Exploring Swing .....	879
<b>31</b>	Servlets .....	907

---

**Part IV   Applying Java**

<b>32</b>	Financial Applets and Servlets .....	931
<b>33</b>	Creating a Download Manager in Java .....	965
<b>A</b>	Using Java's Documentation Comments .....	991
	Index .....	997

---

# Contents

Preface .....	xxix
---------------	------

---

## Part I The Java Language

<b>1 The History and Evolution of Java .....</b>	<b>3</b>
Java's Lineage .....	3
The Birth of Modern Programming: C .....	4
C++: The Next Step .....	5
The Stage Is Set for Java .....	6
The Creation of Java .....	6
The C# Connection .....	8
How Java Changed the Internet .....	8
Java Applets .....	8
Security .....	9
Portability .....	9
Java's Magic: The Bytecode .....	9
Servlets: Java on the Server Side .....	10
The Java Buzzwords .....	10
Simple .....	11
Object-Oriented .....	11
Robust .....	11
Multithreaded .....	12
Architecture-Neutral .....	12
Interpreted and High Performance .....	12
Distributed .....	12
Dynamic .....	13
The Evolution of Java .....	13
Java SE 6 .....	14
A Culture of Innovation .....	14
<b>2 An Overview of Java .....</b>	<b>15</b>
Object-Oriented Programming .....	15
Two Paradigms .....	15
Abstraction .....	16
The Three OOP Principles .....	16
A First Simple Program .....	21
Entering the Program .....	21
Compiling the Program .....	22
A Closer Look at the First Sample Program .....	22



A Second Short Program .....	24
Two Control Statements .....	26
The if Statement .....	26
The for Loop .....	27
Using Blocks of Code .....	29
Lexical Issues .....	30
Whitespace .....	30
Identifiers .....	30
Literals .....	31
Comments .....	31
Separators .....	31
The Java Keywords .....	31
The Java Class Libraries .....	32
<b>3 Data Types, Variables, and Arrays .....</b>	<b>33</b>
Java Is a Strongly Typed Language .....	33
The Primitive Types .....	33
Integers .....	34
byte .....	35
short .....	35
int .....	35
long .....	35
Floating-Point Types .....	36
float .....	36
double .....	36
Characters .....	37
Booleans .....	38
A Closer Look at Literals .....	39
Integer Literals .....	39
Floating-Point Literals .....	40
Boolean Literals .....	40
Character Literals .....	40
String Literals .....	40
Variables .....	41
Declaring a Variable .....	41
Dynamic Initialization .....	42
The Scope and Lifetime of Variables .....	42
Type Conversion and Casting .....	45
Java's Automatic Conversions .....	45
Casting Incompatible Types .....	45
Automatic Type Promotion in Expressions .....	47
The Type Promotion Rules .....	47
Arrays .....	48
One-Dimensional Arrays .....	48
Multidimensional Arrays .....	51
Alternative Array Declaration Syntax .....	55

A Few Words About Strings .....	55
A Note to C/C++ Programmers About Pointers .....	56
<b>4 Operators .....</b>	<b>57</b>
Arithmetic Operators .....	57
The Basic Arithmetic Operators .....	58
The Modulus Operator .....	59
Arithmetic Compound Assignment Operators .....	59
Increment and Decrement .....	60
The Bitwise Operators .....	62
The Bitwise Logical Operators .....	63
The Left Shift .....	65
The Right Shift .....	66
The Unsigned Right Shift .....	68
Bitwise Operator Compound Assignments .....	69
Relational Operators .....	70
Boolean Logical Operators .....	71
Short-Circuit Logical Operators .....	72
The Assignment Operator .....	73
The ? Operator .....	73
Operator Precedence .....	74
Using Parentheses .....	74
<b>5 Control Statements .....</b>	<b>77</b>
Java's Selection Statements .....	77
if .....	77
switch .....	80
Iteration Statements .....	84
while .....	84
do-while .....	86
for .....	88
The For-Each Version of the for Loop .....	92
Nested Loops .....	97
Jump Statements .....	98
Using break .....	98
Using continue .....	102
return .....	103
<b>6 Introducing Classes .....</b>	<b>105</b>
Class Fundamentals .....	105
The General Form of a Class .....	105
A Simple Class .....	106
Declaring Objects .....	109
A Closer Look at new .....	109
Assigning Object Reference Variables .....	111
Introducing Methods .....	111
Adding a Method to the Box Class .....	112

Returning a Value .....	114
Adding a Method That Takes Parameters .....	115
Constructors .....	117
Parameterized Constructors .....	119
The this Keyword .....	120
Instance Variable Hiding .....	121
Garbage Collection .....	121
The finalize( ) Method .....	121
A Stack Class .....	122
<b>7 A Closer Look at Methods and Classes .....</b>	<b>125</b>
Overloading Methods .....	125
Overloading Constructors .....	128
Using Objects as Parameters .....	130
A Closer Look at Argument Passing .....	132
Returning Objects .....	134
Recursion .....	135
Introducing Access Control .....	138
Understanding static .....	141
Introducing final .....	143
Arrays Revisited .....	143
Introducing Nested and Inner Classes .....	145
Exploring the String Class .....	148
Using Command-Line Arguments .....	150
Varargs: Variable-Length Arguments .....	151
Overloading Vararg Methods .....	154
Varargs and Ambiguity .....	155
<b>8 Inheritance .....</b>	<b>157</b>
Inheritance Basics .....	157
Member Access and Inheritance .....	159
A More Practical Example .....	160
A Superclass Variable Can Reference a Subclass Object .....	162
Using super .....	163
Using super to Call Superclass Constructors .....	163
A Second Use for super .....	166
Creating a Multilevel Hierarchy .....	167
When Constructors Are Called .....	170
Method Overriding .....	171
Dynamic Method Dispatch .....	174
Why Overridden Methods? .....	175
Applying Method Overriding .....	176
Using Abstract Classes .....	177
Using final with Inheritance .....	180
Using final to Prevent Overriding .....	180
Using final to Prevent Inheritance .....	181
The Object Class .....	181

<b>9 Packages and Interfaces</b>	<b>183</b>
Packages	183
Defining a Package	184
Finding Packages and CLASSPATH	184
A Short Package Example	185
Access Protection	186
An Access Example	187
Importing Packages	190
Interfaces	192
Defining an Interface	193
Implementing Interfaces	194
Nested Interfaces	196
Applying Interfaces	197
Variables in Interfaces	200
Interfaces Can Be Extended	202
<b>10 Exception Handling</b>	<b>205</b>
Exception-Handling Fundamentals	205
Exception Types	206
Uncaught Exceptions	206
Using try and catch	207
Displaying a Description of an Exception	209
Multiple catch Clauses	209
Nested try Statements	211
throw	213
throws	214
finally	216
Java's Built-in Exceptions	217
Creating Your Own Exception Subclasses	219
Chained Exceptions	221
Using Exceptions	222
<b>11 Multithreaded Programming</b>	<b>223</b>
The Java Thread Model	224
Thread Priorities	224
Synchronization	225
Messaging	225
The Thread Class and the Runnable Interface	226
The Main Thread	226
Creating a Thread	228
Implementing Runnable	228
Extending Thread	230
Choosing an Approach	232
Creating Multiple Threads	232
Using <code>isAlive()</code> and <code>join()</code>	233
Thread Priorities	236

Synchronization .....	238
Using Synchronized Methods .....	239
The synchronized Statement .....	241
Interthread Communication .....	242
Deadlock .....	247
Suspending, Resuming, and Stopping Threads .....	249
Suspending, Resuming, and Stopping Threads	
Using Java 1.1 and Earlier .....	249
The Modern Way of Suspending, Resuming,	
and Stopping Threads .....	251
Using Multithreading .....	254
<b>12 Enumerations, Autoboxing, and Annotations (Metadata) .....</b>	<b>255</b>
Enumerations .....	255
Enumeration Fundamentals .....	255
The values() and valueOf() Methods .....	258
Java Enumerations Are Class Types .....	259
Enumerations Inherit Enum .....	261
Another Enumeration Example .....	263
Type Wrappers .....	264
Autoboxing .....	266
Autoboxing and Methods .....	267
Autoboxing/Unboxing Occurs in Expressions .....	268
Autoboxing/Unboxing Boolean and Character Values .....	270
Autoboxing/Unboxing Helps Prevent Errors .....	271
A Word of Warning .....	271
Annotations (Metadata) .....	272
Annotation Basics .....	272
Specifying a Retention Policy .....	273
Obtaining Annotations at Run Time by Use of Reflection ...	273
The AnnotatedElement Interface .....	278
Using Default Values .....	279
Marker Annotations .....	280
Single-Member Annotations .....	281
The Built-In Annotations .....	282
Some Restrictions .....	284
<b>13 I/O, Applets, and Other Topics .....</b>	<b>285</b>
I/O Basics .....	285
Streams .....	286
Byte Streams and Character Streams .....	286
The Predefined Streams .....	288
Reading Console Input .....	288
Reading Characters .....	289
Reading Strings .....	290

Writing Console Output .....	292
The PrintWriter Class .....	292
Reading and Writing Files .....	293
Applet Fundamentals .....	296
The transient and volatile Modifiers .....	299
Using instanceof .....	300
strictfp .....	302
Native Methods .....	302
Problems with Native Methods .....	306
Using assert .....	306
Assertion Enabling and Disabling Options .....	309
Static Import .....	309
Invoking Overloaded Constructors Through this() .....	312
<b>14 Generics .....</b>	<b>315</b>
What Are Generics? .....	316
A Simple Generics Example .....	316
Generics Work Only with Objects .....	320
Generic Types Differ Based on Their Type Arguments .....	320
How Generics Improve Type Safety .....	320
A Generic Class with Two Type Parameters .....	322
The General Form of a Generic Class .....	324
Bounded Types .....	324
Using Wildcard Arguments .....	327
Bounded Wildcards .....	329
Creating a Generic Method .....	334
Generic Constructors .....	336
Generic Interfaces .....	337
Raw Types and Legacy Code .....	339
Generic Class Hierarchies .....	342
Using a Generic Superclass .....	342
A Generic Subclass .....	344
Run-Time Type Comparisons Within a Generic Hierarchy .....	345
Casting .....	348
Overriding Methods in a Generic Class .....	348
Erasure .....	349
Bridge Methods .....	351
Ambiguity Errors .....	353
Some Generic Restrictions .....	354
Type Parameters Can't Be Instantiated .....	354
Restrictions on Static Members .....	354
Generic Array Restrictions .....	355
Generic Exception Restriction .....	356
Final Thoughts on Generics .....	356

---

**Part II   The Java Library**

<b>15</b>	<b>String Handling</b> .....	<b>359</b>
	The String Constructors .....	359
	String Length .....	362
	Special String Operations .....	362
	String Literals .....	362
	String Concatenation .....	362
	String Concatenation with Other Data Types .....	363
	String Conversion and toString() .....	364
	Character Extraction .....	365
	charAt() .....	365
	getChars() .....	365
	getBytes() .....	366
	toCharArray() .....	366
	String Comparison .....	366
	equals() and equalsIgnoreCase() .....	366
	regionMatches() .....	367
	startsWith() and endsWith() .....	368
	equals() Versus == .....	368
	compareTo() .....	369
	Searching Strings .....	370
	Modifying a String .....	372
	substring() .....	372
	concat() .....	373
	replace() .....	373
	trim() .....	373
	Data Conversion Using valueOf() .....	374
	Changing the Case of Characters Within a String .....	375
	Additional String Methods .....	376
	StringBuffer .....	377
	StringBuffer Constructors .....	377
	length() and capacity() .....	378
	ensureCapacity() .....	378
	setLength() .....	378
	charAt() and setCharAt() .....	379
	getChars() .....	379
	append() .....	380
	insert() .....	381
	reverse() .....	381
	delete() and deleteCharAt() .....	382
	replace() .....	382
	substring() .....	383
	Additional StringBuffer Methods .....	383
	StringBuilder .....	384

<b>16 Exploring java.lang</b> .....	<b>385</b>
Primitive Type Wrappers .....	386
Number .....	386
Double and Float .....	386
Byte, Short, Integer, and Long .....	390
Character .....	398
Recent Additions to Character for Unicode Code Point Support .....	401
Boolean .....	402
Void .....	403
Process .....	403
Runtime .....	404
Memory Management .....	405
Executing Other Programs .....	406
ProcessBuilder .....	407
System .....	409
Using currentTimeMillis( ) to Time Program Execution .....	410
Using arraycopy( ) .....	411
Environment Properties .....	412
Object .....	412
Using clone( ) and the Cloneable Interface .....	413
Class .....	415
ClassLoader .....	418
Math .....	418
Transcendental Functions .....	418
Exponential Functions .....	419
Rounding Functions .....	419
Miscellaneous Math Methods .....	420
StrictMath .....	422
Compiler .....	422
Thread, ThreadGroup, and Runnable .....	422
The Runnable Interface .....	422
Thread .....	422
ThreadGroup .....	424
ThreadLocal and InheritableThreadLocal .....	429
Package .....	429
RuntimePermission .....	431
Throwable .....	431
SecurityManager .....	431
StackTraceElement .....	431
Enum .....	432
The CharSequence Interface .....	433
The Comparable Interface .....	433
The Appendable Interface .....	434



The Iterable Interface .....	434
The Readable Interface .....	434
The java.lang Subpackages .....	435
java.lang.annotation .....	435
java.lang.instrument .....	435
java.lang.management .....	435
java.lang.ref .....	435
java.lang.reflect .....	436
<b>17 java.util Part 1: The Collections Framework .....</b>	<b>437</b>
Collections Overview .....	438
Recent Changes to Collections .....	439
Generics Fundamentally Change the Collections Framework .....	439
Autoboxing Facilitates the Use of Primitive Types .....	439
The For-Each Style for Loop .....	440
The Collection Interfaces .....	440
The Collection Interface .....	441
The List Interface .....	441
The Set Interface .....	443
The SortedSet Interface .....	444
The NavigableSet Interface .....	444
The Queue Interface .....	445
The Deque Interface .....	446
The Collection Classes .....	448
The ArrayList Class .....	448
The LinkedList Class .....	451
The HashSet Class .....	453
The LinkedHashSet Class .....	454
The TreeSet Class .....	455
The PriorityQueue Class .....	456
The ArrayDeque Class .....	457
The EnumSet Class .....	458
Accessing a Collection via an Iterator .....	458
Using an Iterator .....	459
The For-Each Alternative to Iterators .....	461
Storing User-Defined Classes in Collections .....	462
The RandomAccess Interface .....	463
Working with Maps .....	464
The Map Interfaces .....	464
The NavigableMap Interface .....	466
The Map Classes .....	468
Comparators .....	472
Using a Comparator .....	473
The Collection Algorithms .....	475

Arrays .....	480
Why Generic Collections? .....	484
The Legacy Classes and Interfaces .....	487
The Enumeration Interface .....	487
Vector .....	487
Stack .....	491
Dictionary .....	493
Hashtable .....	494
Properties .....	497
Using store() and load() .....	500
Parting Thoughts on Collections .....	501
<b>18 java.util Part 2: More Utility Classes .....</b>	<b>503</b>
StringTokenizer .....	503
BitSet .....	505
Date .....	507
Calendar .....	509
GregorianCalendar .....	512
TimeZone .....	513
SimpleTimeZone .....	514
Locale .....	515
Random .....	516
Observable .....	518
The Observer Interface .....	519
An Observer Example .....	519
Timer and TimerTask .....	522
Currency .....	524
Formatter .....	525
The Formatter Constructors .....	526
The Formatter Methods .....	526
Formatting Basics .....	526
Formatting Strings and Characters .....	529
Formatting Numbers .....	529
Formatting Time and Date .....	530
The %n and %% Specifiers .....	532
Specifying a Minimum Field Width .....	533
Specifying Precision .....	534
Using the Format Flags .....	535
Justifying Output .....	535
The Space, +, 0, and ( Flags .....	536
The Comma Flag .....	537
The # Flag .....	537
The Uppercase Option .....	537
Using an Argument Index .....	538
The Java printf() Connection .....	539

Scanner .....	540
The Scanner Constructors .....	540
Scanning Basics .....	541
Some Scanner Examples .....	544
Setting Delimiters .....	547
Other Scanner Features .....	548
The ResourceBundle, ListResourceBundle, and PropertyResourceBundle Classes .....	549
Miscellaneous Utility Classes and Interfaces .....	553
The java.util Subpackages .....	554
java.util.concurrent, java.util.concurrent.atomic, and java.util.concurrent.locks .....	554
java.util.jar .....	554
java.util.logging .....	554
java.util.prefs .....	554
java.util.regex .....	554
java.util.spi .....	554
java.util.zip .....	554
<b>19 Input/Output: Exploring java.io .....</b>	<b>555</b>
The Java I/O Classes and Interfaces .....	555
File .....	556
Directories .....	559
Using FilenameFilter .....	560
The listFiles() Alternative .....	561
Creating Directories .....	561
The Closeable and Flushable Interfaces .....	561
The Stream Classes .....	562
The Byte Streams .....	562
InputStream .....	562
OutputStream .....	562
FileInputStream .....	564
FileOutputStream .....	565
ByteArrayInputStream .....	567
ByteArrayOutputStream .....	568
Filtered Byte Streams .....	569
Buffered Byte Streams .....	569
SequenceInputStream .....	573
PrintStream .....	574
DataOutputStream and DataInputStream .....	576
RandomAccessFile .....	578
The Character Streams .....	578
Reader .....	579
Writer .....	579
FileReader .....	579
FileWriter .....	579

CharArrayReader .....	582
CharArrayWriter .....	582
BufferedReader .....	583
BufferedWriter .....	585
PushbackReader .....	585
PrintWriter .....	586
The Console Class .....	587
Using Stream I/O .....	589
Improving wc( ) Using a StreamTokenizer .....	590
Serialization .....	592
Serializable .....	593
Externalizable .....	593
ObjectOutput .....	593
ObjectOutputStream .....	593
ObjectInput .....	595
ObjectInputStream .....	595
A Serialization Example .....	595
Stream Benefits .....	598
<b>20 Networking .....</b>	<b>599</b>
Networking Basics .....	599
The Networking Classes and Interfaces .....	600
InetAddress .....	601
Factory Methods .....	601
Instance Methods .....	602
Inet4Address and Inet6Address .....	603
TCP/IP Client Sockets .....	603
URL .....	605
URLConnection .....	607
HttpURLConnection .....	610
The URI Class .....	612
Cookies .....	612
TCP/IP Server Sockets .....	612
Datagrams .....	613
DatagramSocket .....	613
DatagramPacket .....	614
A Datagram Example .....	615
<b>21 The Applet Class .....</b>	<b>617</b>
Two Types of Applets .....	617
Applet Basics .....	617
The Applet Class .....	618
Applet Architecture .....	620
An Applet Skeleton .....	621
Applet Initialization and Termination .....	622
Overriding update( ) .....	623

Simple Applet Display Methods .....	623
Requesting Repainting .....	625
A Simple Banner Applet .....	626
Using the Status Window .....	628
The HTML APPLET Tag .....	629
Passing Parameters to Applets .....	630
Improving the Banner Applet .....	631
getDocumentBase( ) and getCodeBase( ) .....	633
AppletContext and showDocument( ) .....	634
The AudioClip Interface .....	635
The AppletStub Interface .....	635
Outputting to the Console .....	636
<b>22 Event Handling .....</b>	<b>637</b>
Two Event Handling Mechanisms .....	637
The Delegation Event Model .....	638
Events .....	638
Event Sources .....	638
Event Listeners .....	639
Event Classes .....	639
The ActionEvent Class .....	640
The AdjustmentEvent Class .....	641
The ComponentEvent Class .....	642
The ContainerEvent Class .....	642
The FocusEvent Class .....	643
The InputEvent Class .....	643
The ItemEvent Class .....	644
The KeyEvent Class .....	645
The MouseEvent Class .....	646
The MouseWheelEvent Class .....	647
The TextEvent Class .....	648
The WindowEvent Class .....	648
Sources of Events .....	649
Event Listener Interfaces .....	650
The ActionListener Interface .....	650
The AdjustmentListener Interface .....	651
The ComponentListener Interface .....	651
The ContainerListener Interface .....	651
The FocusListener Interface .....	651
The ItemListener Interface .....	651
The KeyListener Interface .....	651
The MouseListener Interface .....	652
The MouseMotionListener Interface .....	652
The MouseWheelListener Interface .....	652
The TextListener Interface .....	652
The WindowFocusListener Interface .....	652

The WindowListener Interface .....	653
Using the Delegation Event Model .....	653
Handling Mouse Events .....	653
Handling Keyboard Events .....	656
Adapter Classes .....	659
Inner Classes .....	660
Anonymous Inner Classes .....	662
<b>23 Introducing the AWT: Working with Windows, Graphics,     and Text .....</b>	<b>663</b>
AWT Classes .....	664
Window Fundamentals .....	666
Component .....	666
Container .....	666
Panel .....	667
Window .....	667
Frame .....	667
Canvas .....	667
Working with Frame Windows .....	667
Setting the Window's Dimensions .....	668
Hiding and Showing a Window .....	668
Setting a Window's Title .....	668
Closing a Frame Window .....	668
Creating a Frame Window in an Applet .....	668
Handling Events in a Frame Window .....	670
Creating a Windowed Program .....	674
Displaying Information Within a Window .....	676
Working with Graphics .....	676
Drawing Lines .....	677
Drawing Rectangles .....	677
Drawing Ellipses and Circles .....	678
Drawing Arcs .....	679
Drawing Polygons .....	680
Sizing Graphics .....	681
Working with Color .....	682
Color Methods .....	683
Setting the Current Graphics Color .....	684
A Color Demonstration Applet .....	684
Setting the Paint Mode .....	685
Working with Fonts .....	686
Determining the Available Fonts .....	687
Creating and Selecting a Font .....	689
Obtaining Font Information .....	690
Managing Text Output Using FontMetrics .....	691
Displaying Multiple Lines of Text .....	693

Centering Text .....	694
Multiline Text Alignment .....	695
<b>24 Using AWT Controls, Layout Managers, and Menus .....</b>	<b>701</b>
Control Fundamentals .....	701
Adding and Removing Controls .....	702
Responding to Controls .....	702
The HeadlessException .....	702
Labels .....	702
Using Buttons .....	704
Handling Buttons .....	704
Applying Check Boxes .....	707
Handling Check Boxes .....	707
CheckboxGroup .....	709
Choice Controls .....	711
Handling Choice Lists .....	711
Using Lists .....	713
Handling Lists .....	714
Managing Scroll Bars .....	716
Handling Scroll Bars .....	717
Using a TextField .....	719
Handling a TextField .....	720
Using a TextArea .....	721
Understanding Layout Managers .....	723
FlowLayout .....	724
BorderLayout .....	725
Using Insets .....	727
GridLayout .....	728
CardLayout .....	730
GridBagLayout .....	732
Menu Bars and Menus .....	737
Dialog Boxes .....	742
FileDialog .....	747
Handling Events by Extending AWT Components .....	748
Extending Button .....	749
Extending Checkbox .....	750
Extending a Check Box Group .....	751
Extending Choice .....	752
Extending List .....	752
Extending Scrollbar .....	753
<b>25 Images .....</b>	<b>755</b>
File Formats .....	755
Image Fundamentals: Creating, Loading, and Displaying .....	756
Creating an Image Object .....	756

Loading an Image .....	756
Displaying an Image .....	757
ImageObserver .....	758
Double Buffering .....	759
MediaTracker .....	762
ImageProducer .....	765
MemoryImageSource .....	766
ImageConsumer .....	767
PixelGrabber .....	767
ImageFilter .....	770
CropImageFilter .....	770
RGBImageFilter .....	772
Cell Animation .....	783
Additional Imaging Classes .....	786
<b>26 The Concurrency Utilities .....</b>	<b>787</b>
The Concurrent API Packages .....	788
java.util.concurrent .....	788
java.util.concurrent.atomic .....	789
java.util.concurrent.locks .....	789
Using Synchronization Objects .....	789
Semaphore .....	789
CountDownLatch .....	795
CyclicBarrier .....	796
Exchanger .....	799
Using an Executor .....	801
A Simple Executor Example .....	802
Using Callable and Future .....	804
The TimeUnit Enumeration .....	806
The Concurrent Collections .....	808
Locks .....	808
Atomic Operations .....	811
The Concurrency Utilities Versus Java's Traditional Approach ....	812
<b>27 NIO, Regular Expressions, and Other Packages .....</b>	<b>813</b>
The Core Java API Packages .....	813
NIO .....	815
NIO Fundamentals .....	815
Charsets and Selectors .....	818
Using the NIO System .....	819
Is NIO the Future of I/O Handling? .....	825
Regular Expression Processing .....	825
Pattern .....	825
Matcher .....	826
Regular Expression Syntax .....	827
Demonstrating Pattern Matching .....	827



Two Pattern-Matching Options .....	833
Exploring Regular Expressions .....	833
Reflection .....	833
Remote Method Invocation (RMI) .....	837
A Simple Client/Server Application Using RMI .....	837
Text Formatting .....	840
DateFormat Class .....	840
SimpleDateFormat Class .....	842

---

## **Part III   Software Development Using Java**

<b>28   Java Beans</b> .....	<b>847</b>
What Is a Java Bean? .....	847
Advantages of Java Beans .....	848
Introspection .....	848
Design Patterns for Properties .....	848
Design Patterns for Events .....	849
Methods and Design Patterns .....	850
Using the BeanInfo Interface .....	850
Bound and Constrained Properties .....	850
Persistence .....	851
Customizers .....	851
The Java Beans API .....	851
Introspector .....	853
PropertyDescriptor .....	854
EventSetDescriptor .....	854
MethodDescriptor .....	854
A Bean Example .....	854
<b>29   Introducing Swing</b> .....	<b>859</b>
The Origins of Swing .....	859
Swing Is Built on the AWT .....	860
Two Key Swing Features .....	860
Swing Components Are Lightweight .....	860
Swing Supports a Pluggable Look and Feel .....	860
The MVC Connection .....	861
Components and Containers .....	862
Components .....	862
Containers .....	863
The Top-Level Container Panes .....	863
The Swing Packages .....	863
A Simple Swing Application .....	864
Event Handling .....	868
Create a Swing Applet .....	871
Painting in Swing .....	873

	Painting Fundamentals .....	874
	Compute the Paintable Area .....	875
	A Paint Example .....	875
<b>30</b>	<b>Exploring Swing .....</b>	<b>879</b>
	JLabel and ImageIcon .....	879
	JTextField .....	881
	The Swing Buttons .....	883
	JButton .....	883
	JToggleButton .....	885
	Check Boxes .....	887
	Radio Buttons .....	889
	JTabbedPane .....	891
	JScrollPane .....	893
	JList .....	895
	JComboBox .....	898
	Trees .....	900
	JTable .....	904
	Continuing Your Exploration of Swing .....	906
<b>31</b>	<b>Servlets .....</b>	<b>907</b>
	Background .....	907
	The Life Cycle of a Servlet .....	908
	Using Tomcat for Servlet Development .....	908
	A Simple Servlet .....	910
	Create and Compile the Servlet Source Code .....	910
	Start Tomcat .....	911
	Start a Web Browser and Request the Servlet .....	911
	The Servlet API .....	911
	The javax.servlet Package .....	911
	The Servlet Interface .....	912
	The ServletConfig Interface .....	912
	The ServletContext Interface .....	912
	The ServletRequest Interface .....	913
	The ServletResponse Interface .....	913
	The GenericServlet Class .....	914
	The ServletInputStream Class .....	915
	The ServletOutputStream Class .....	915
	The Servlet Exception Classes .....	915
	Reading Servlet Parameters .....	915
	The javax.servlet.http Package .....	917
	The HttpServletRequest Interface .....	917
	The HttpServletResponse Interface .....	917
	The HttpSession Interface .....	917
	The HttpSessionBindingListener Interface .....	919
	The Cookie Class .....	919

The HttpServlet Class .....	921
The HttpSessionEvent Class .....	921
The HttpSessionBindingEvent Class .....	922
Handling HTTP Requests and Responses .....	922
Handling HTTP GET Requests .....	922
Handling HTTP POST Requests .....	924
Using Cookies .....	925
Session Tracking .....	927

## **Part IV   Applying Java**

<b>32   Financial Applets and Servlets .....</b>	<b>931</b>
Finding the Payments for a Loan .....	932
The RegPay Fields .....	935
The init( ) Method .....	936
The makeGUI( ) Method .....	936
The actionPerformed( ) Method .....	938
The compute( ) Method .....	939
Finding the Future Value of an Investment .....	940
Finding the Initial Investment Required to Achieve a Future Value ...	943
Finding the Initial Investment Needed for a Desired Annuity .....	947
Finding the Maximum Annuity for a Given Investment .....	951
Finding the Remaining Balance on a Loan .....	955
Creating Financial Servlets .....	959
Converting the RegPay Applet into a Servlet .....	960
The RegPayS Servlet .....	960
Some Things to Try .....	963
<b>33   Creating a Download Manager in Java .....</b>	<b>965</b>
Understanding Internet Downloads .....	966
An Overview of the Download Manager .....	966
The Download Class .....	967
The Download Variables .....	971
The Download Constructor .....	971
The download( ) Method .....	971
The run( ) Method .....	971
The stateChanged( ) Method .....	975
Action and Accessor Methods .....	975
The ProgressRenderer Class .....	975
The DownloadsTableModel Class .....	976
The addDownload( ) Method .....	978
The clearDownload( ) Method .....	979
The getColumnClass( ) Method .....	979
The getValueAt( ) Method .....	979
The update( ) Method .....	980

The DownloadManager Class .....	980
The DownloadManager Variables .....	986
The DownloadManager Constructor .....	986
The verifyUrl( ) Method .....	986
The tableSelectionChanged( ) Method .....	987
The updateButtons( ) Method .....	988
Handling Action Events .....	989
Compiling and Running the Download Manager .....	989
Enhancing the Download Manager .....	990
<b>A Using Java's Documentation Comments .....</b>	<b>991</b>
The javadoc Tags .....	991
@author .....	992
{@code} .....	992
@deprecated .....	992
{@docRoot} .....	993
@exception .....	993
{@inheritDoc} .....	993
{@link} .....	993
{@linkplain} .....	993
{@literal} .....	993
@param .....	993
@return .....	993
@see .....	994
@serial .....	994
@serialData .....	994
@serialField .....	994
@since .....	994
@throws .....	994
{@value} .....	995
@version .....	995
The General Form of a Documentation Comment .....	995
What javadoc Outputs .....	995
An Example that Uses Documentation Comments .....	995
<b>Index .....</b>	<b>997</b>

*This page intentionally left blank*

---

# Preface

As I write this, Java is just beginning its second decade. Unlike many other computer languages whose influence begins to wane over the years, Java's has grown stronger with the passage of time. Java leapt to the forefront of Internet programming with its first release. Each subsequent version has solidified that position. Today, Java is still the first and best choice for developing web-based applications.

One reason for Java's success is its agility. Java has rapidly adapted to changes in the programming environment and to changes in the way that programmers program. Most importantly, it has not just followed the trends, *it has helped create them*. Unlike some other languages that have a revision cycle of approximately 10 years, Java's release cycle averages about 1.5 years! Java's ability to accommodate the fast rate of change in the computing world is a crucial part of why it has stayed at the forefront of computer language design. With the release of Java SE 6, Java's leadership remains unchallenged. If you are programming for the Internet, you have chosen the right language. Java has been and continues to be the preeminent language of the Internet.

As many readers will know, this is the seventh edition of the book, which was first published in 1996. This edition has been updated for Java SE 6. It has also been expanded in several key areas. Here are two examples: it now includes twice as much coverage of Swing and a more detailed discussion of resource bundles. Throughout are many other additions and improvements. In all, dozens of pages of new material have been incorporated.

---

## A Book for All Programmers

This book is for all programmers, whether you are a novice or an experienced pro. The beginner will find its carefully paced discussions and many examples especially helpful. Its in-depth coverage of Java's more advanced features and libraries will appeal to the pro. For both, it offers a lasting resource and handy reference.

---

## What's Inside

This book is a comprehensive guide to the Java language, describing its syntax, keywords, and fundamental programming principles. Significant portions of the Java API library are also examined. The book is divided into four parts, each focusing on a different aspect of the Java programming environment.