

JavaTM



HOW TO PROGRAM

NINTH EDITION

PAUL DEITEL
HARVEY DEITEL

VOLUME 2

Contents

Appendices M–Q are PDF documents posted online at the book's Companion Website (www.pearsonhighered.com/deitel/).

Preface	xxiii
----------------	--------------

Before You Begin	xxxiii
-------------------------	---------------

I	Introduction to Computers and Java	I
1.1	Introduction	2
1.2	Computers: Hardware and Software	5
1.3	Data Hierarchy	6
1.4	Computer Organization	8
1.5	Machine Languages, Assembly Languages and High-Level Languages	10
1.6	Introduction to Object Technology	11
1.7	Operating Systems	13
1.8	Programming Languages	16
1.9	Java and a Typical Java Development Environment	18
1.10	Test-Driving a Java Application	22
1.11	Web 2.0: Going Social	26
1.12	Software Technologies	29
1.13	Keeping Up-to-Date with Information Technologies	31
1.14	Wrap-Up	32
2	Introduction to Java Applications	37
2.1	Introduction	38
2.2	Your First Program in Java: Printing a Line of Text	38
2.3	Modifying Your First Java Program	44
2.4	Displaying Text with <code>printf</code>	46
2.5	Another Application: Adding Integers	47
2.6	Memory Concepts	52

23.4	Applet Life-Cycle Methods	951
23.5	Initialization with Method <code>init</code>	952
23.6	Sandbox Security Model	954
23.7	Java Web Start and the Java Network Launch Protocol (JNLP)	956
23.7.1	Packaging the DrawTest Applet for Use with Java Web Start	956
23.7.2	JNLP Document for the DrawTest Applet	957
23.8	Wrap-Up	961

24 Multimedia: Applets and Applications 967

24.1	Introduction	968
24.2	Loading, Displaying and Scaling Images	969
24.3	Animating a Series of Images	975
24.4	Image Maps	982
24.5	Loading and Playing Audio Clips	985
24.6	Playing Video and Other Media with Java Media Framework	988
24.7	Wrap-Up	992
24.8	Web Resources	992

25 GUI Components: Part 2 1000

25.1	Introduction	1001
25.2	<code>JSlider</code>	1001
25.3	Windows: Additional Notes	1005
25.4	Using Menus with Frames	1006
25.5	<code>JPopupMenu</code>	1014
25.6	Pluggable Look-and-Feel	1017
25.7	<code>JDesktopPane</code> and <code>JInternalFrame</code>	1022
25.8	<code>JTabbedPane</code>	1026
25.9	Layout Managers: <code>BoxLayout</code> and <code>GridBagLayout</code>	1028
25.10	Wrap-Up	1040

26 Multithreading 1045

26.1	Introduction	1046
26.2	Thread States: Life Cycle of a Thread	1048
26.3	Creating and Executing Threads with Executor Framework	1051
26.4	Thread Synchronization	1054
26.4.1	Unsynchronized Data Sharing	1055
26.4.2	Synchronized Data Sharing—Making Operations Atomic	1059
26.5	Producer/Consumer Relationship without Synchronization	1062
26.6	Producer/Consumer Relationship: <code>ArrayBlockingQueue</code>	1070
26.7	Producer/Consumer Relationship with Synchronization	1073
26.8	Producer/Consumer Relationship: Bounded Buffers	1079
26.9	Producer/Consumer Relationship: The Lock and Condition Interfaces	1086
26.10	Concurrent Collections Overview	1093
26.11	Multithreading with GUI	1095

26.11.1	Performing Computations in a Worker Thread	1096
26.11.2	Processing Intermediate Results with <code>SwingWorker</code>	1102
26.12	Interfaces <code>Callable</code> and <code>Future</code>	1109
26.13	Java SE 7: Fork/Join Framework	1109
26.14	Wrap-Up	1110

27 Networking 1118

27.1	Introduction	1119
27.2	Manipulating URLs	1120
27.3	Reading a File on a Web Server	1125
27.4	Establishing a Simple Server Using Stream Sockets	1128
27.5	Establishing a Simple Client Using Stream Sockets	1130
27.6	Client/Server Interaction with Stream Socket Connections	1130
27.7	Datagrams: Connectionless Client/Server Interaction	1142
27.8	Client/Server Tic-Tac-Toe Using a Multithreaded Server	1150
27.9	[Web Bonus] Case Study: <code>DeitelMessenger</code>	1165
27.10	Wrap-Up	1165

28 Accessing Databases with JDBC 1171

28.1	Introduction	1172
28.2	Relational Databases	1173
28.3	Relational Database Overview: The books Database	1174
28.4	SQL	1177
28.4.1	Basic SELECT Query	1178
28.4.2	WHERE Clause	1179
28.4.3	ORDER BY Clause	1181
28.4.4	Merging Data from Multiple Tables: INNER JOIN	1182
28.4.5	INSERT Statement	1184
28.4.6	UPDATE Statement	1185
28.4.7	DELETE Statement	1186
28.5	Instructions for Installing MySQL and MySQL Connector/J	1186
28.6	Instructions for Setting Up a MySQL User Account	1187
28.7	Creating Database books in MySQL	1188
28.8	Manipulating Databases with JDBC	1189
28.8.1	Connecting to and Querying a Database	1189
28.8.2	Querying the books Database	1194
28.9	RowSet Interface	1207
28.10	Java DB/Apache Derby	1209
28.11	PreparedStatement	1211
28.12	Stored Procedures	1226
28.13	Transaction Processing	1227
28.14	Wrap-Up	1227
28.15	Web Resources	1228

29	JavaServer™ Faces Web Apps: Part 1	1235
29.1	Introduction	1236
29.2	HyperText Transfer Protocol (HTTP) Transactions	1237
29.3	Multitier Application Architecture	1240
29.4	Your First JSF Web App	1241
29.4.1	The Default <code>index.xhtml</code> Document: Introducing Facelets	1242
29.4.2	Examining the <code>WebTimeBean</code> Class	1244
29.4.3	Building the <code>webTime</code> JSF Web App in NetBeans	1246
29.5	Model-View-Controller Architecture of JSF Apps	1250
29.6	Common JSF Components	1250
29.7	Validation Using JSF Standard Validators	1254
29.8	Session Tracking	1261
29.8.1	Cookies	1262
29.8.2	Session Tracking with <code>@SessionScoped</code> Beans	1263
29.9	Wrap-Up	1269
30	JavaServer™ Faces Web Apps: Part 2	1276
30.1	Introduction	1277
30.2	Accessing Databases in Web Apps	1277
30.2.1	Setting Up the Database	1279
30.2.2	<code>@ManagedBean</code> Class <code>AddressBean</code>	1282
30.2.3	<code>index.xhtml</code> Facelets Page	1286
30.2.4	<code>addentry.xhtml</code> Facelets Page	1288
30.3	Ajax	1290
30.4	Adding Ajax Functionality to the Validation App	1292
30.5	Wrap-Up	1295
31	Web Services	1299
31.1	Introduction	1300
31.2	Web Service Basics	1302
31.3	Simple Object Access Protocol (SOAP)	1302
31.4	Representational State Transfer (REST)	1302
31.5	JavaScript Object Notation (JSON)	1303
31.6	Publishing and Consuming SOAP-Based Web Services	1303
31.6.1	Creating a Web Application Project and Adding a Web Service Class in NetBeans	1303
31.6.2	Defining the <code>WelComeSOAP</code> Web Service in NetBeans	1304
31.6.3	Publishing the <code>WelComeSOAP</code> Web Service from NetBeans	1307
31.6.4	Testing the <code>WelComeSOAP</code> Web Service with GlassFish Application Server's Tester Web Page	1308
31.6.5	Describing a Web Service with the Web Service Description Language (WSDL)	1309
31.6.6	Creating a Client to Consume the <code>WelComeSOAP</code> Web Service	1310
31.6.7	Consuming the <code>WelComeSOAP</code> Web Service	1312

31.7	Publishing and Consuming REST-Based XML Web Services	1315
31.7.1	Creating a REST-Based XML Web Service	1315
31.7.2	Consuming a REST-Based XML Web Service	1318
31.8	Publishing and Consuming REST-Based JSON Web Services	1320
31.8.1	Creating a REST-Based JSON Web Service	1320
31.8.2	Consuming a REST-Based JSON Web Service	1322
31.9	Session Tracking in a SOAP Web Service	1324
31.9.1	Creating a Blackjack Web Service	1325
31.9.2	Consuming the Blackjack Web Service	1328
31.10	Consuming a Database-Driven SOAP Web Service	1339
31.10.1	Creating the Reservation Database	1340
31.10.2	Creating a Web Application to Interact with the Reservation Service	1343
31.11	Equation Generator: Returning User-Defined Types	1346
31.11.1	Creating the EquationGeneratorXML Web Service	1349
31.11.2	Consuming the EquationGeneratorXML Web Service	1350
31.11.3	Creating the EquationGeneratorJSON Web Service	1354
31.11.4	Consuming the EquationGeneratorJSON Web Service	1354
31.12	Wrap-Up	1357

A Operator Precedence Chart **1365**

B ASCII Character Set **1367**

C Keywords and Reserved Words **1368**

D Primitive Types **1369**

E Using the Java API Documentation **1370**

E.1	Introduction	1370
E.2	Navigating the Java API	1370

F Using the Debugger **1378**

F.1	Introduction	1379
F.2	Breakpoints and the run, stop, cont and print Commands	1379
F.3	The print and set Commands	1383
F.4	Controlling Execution Using the step, step up and next Commands	1385
F.5	The watch Command	1388
F.6	The clear Command	1391
F.7	Wrap-Up	1393

G Formatted Output **1395**

G.1	Introduction	1396
-----	--------------	------

2.7	Arithmetic	53
2.8	Decision Making: Equality and Relational Operators	56
2.9	Wrap-Up	60

3 Introduction to Classes, Objects, Methods and Strings 71

3.1	Introduction	72
3.2	Declaring a Class with a Method and Instantiating an Object of a Class	72
3.3	Declaring a Method with a Parameter	76
3.4	Instance Variables, <i>set</i> Methods and <i>get</i> Methods	79
3.5	Primitive Types vs. Reference Types	84
3.6	Initializing Objects with Constructors	85
3.7	Floating-Point Numbers and Type <code>double</code>	88
3.8	(Optional) GUI and Graphics Case Study: Using Dialog Boxes	92
3.9	Wrap-Up	95

4 Control Statements: Part I 102

4.1	Introduction	103
4.2	Algorithms	103
4.3	Pseudocode	104
4.4	Control Structures	104
4.5	<code>if</code> Single-Selection Statement	107
4.6	<code>if...else</code> Double-Selection Statement	107
4.7	<code>while</code> Repetition Statement	112
4.8	Formulating Algorithms: Counter-Controlled Repetition	113
4.9	Formulating Algorithms: Sentinel-Controlled Repetition	118
4.10	Formulating Algorithms: Nested Control Statements	125
4.11	Compound Assignment Operators	130
4.12	Increment and Decrement Operators	130
4.13	Primitive Types	134
4.14	(Optional) GUI and Graphics Case Study: Creating Simple Drawings	134
4.15	Wrap-Up	138

5 Control Statements: Part 2 151

5.1	Introduction	152
5.2	Essentials of Counter-Controlled Repetition	152
5.3	<code>for</code> Repetition Statement	154
5.4	Examples Using the <code>for</code> Statement	158
5.5	<code>do...while</code> Repetition Statement	162
5.6	<code>switch</code> Multiple-Selection Statement	164
5.7	<code>break</code> and <code>continue</code> Statements	172
5.8	Logical Operators	173
5.9	Structured Programming Summary	179

5.10	(Optional) GUI and Graphics Case Study: Drawing Rectangles and Ovals	184
5.11	Wrap-Up	187

6 Methods: A Deeper Look 197

6.1	Introduction	198
6.2	Program Modules in Java	198
6.3	static Methods, static Fields and Class Math	200
6.4	Declaring Methods with Multiple Parameters	202
6.5	Notes on Declaring and Using Methods	205
6.6	Method-Call Stack and Activation Records	206
6.7	Argument Promotion and Casting	207
6.8	Java API Packages	208
6.9	Case Study: Random-Number Generation	210
6.9.1	Generalized Scaling and Shifting of Random Numbers	214
6.9.2	Random-Number Repeatability for Testing and Debugging	214
6.10	Case Study: A Game of Chance; Introducing Enumerations	215
6.11	Scope of Declarations	219
6.12	Method Overloading	222
6.13	(Optional) GUI and Graphics Case Study: Colors and Filled Shapes	224
6.14	Wrap-Up	227

7 Arrays and ArrayLists 240

7.1	Introduction	241
7.2	Arrays	242
7.3	Declaring and Creating Arrays	243
7.4	Examples Using Arrays	244
7.5	Case Study: Card Shuffling and Dealing Simulation	254
7.6	Enhanced for Statement	258
7.7	Passing Arrays to Methods	259
7.8	Case Study: Class GradeBook Using an Array to Store Grades	262
7.9	Multidimensional Arrays	268
7.10	Case Study: Class GradeBook Using a Two-Dimensional Array	271
7.11	Variable-Length Argument Lists	278
7.12	Using Command-Line Arguments	279
7.13	Class Arrays	281
7.14	Introduction to Collections and Class ArrayList	284
7.15	(Optional) GUI and Graphics Case Study: Drawing Arcs	286
7.16	Wrap-Up	289

8 Classes and Objects: A Deeper Look 311

8.1	Introduction	312
8.2	Time Class Case Study	312
8.3	Controlling Access to Members	316
8.4	Referring to the Current Object's Members with the this Reference	317
8.5	Time Class Case Study: Overloaded Constructors	320

8.6	Default and No-Argument Constructors	326
8.7	Notes on <i>Set</i> and <i>Get</i> Methods	326
8.8	Composition	328
8.9	Enumerations	331
8.10	Garbage Collection and Method <code>finalize</code>	333
8.11	<code>static</code> Class Members	334
8.12	<code>static</code> Import	338
8.13	<code>final</code> Instance Variables	339
8.14	Time Class Case Study: Creating Packages	340
8.15	Package Access	345
8.16	(Optional) GUI and Graphics Case Study: Using Objects with Graphics	347
8.17	Wrap-Up	351

9 Object-Oriented Programming: Inheritance 359

9.1	Introduction	360
9.2	Superclasses and Subclasses	361
9.3	<code>protected</code> Members	363
9.4	Relationship between Superclasses and Subclasses	364
9.4.1	Creating and Using a <code>CommissionEmployee</code> Class	364
9.4.2	Creating and Using a <code>BasePlusCommissionEmployee</code> Class	370
9.4.3	Creating a <code>CommissionEmployee–BasePlusCommissionEmployee</code> Inheritance Hierarchy	375
9.4.4	<code>CommissionEmployee–BasePlusCommissionEmployee</code> Inheritance Hierarchy Using <code>protected</code> Instance Variables	377
9.4.5	<code>CommissionEmployee–BasePlusCommissionEmployee</code> Inheritance Hierarchy Using <code>private</code> Instance Variables	380
9.5	Constructors in Subclasses	385
9.6	Software Engineering with Inheritance	386
9.7	Class Object	387
9.8	(Optional) GUI and Graphics Case Study: Displaying Text and Images Using Labels	388
9.9	Wrap-Up	391

10 Object-Oriented Programming: Polymorphism 394

10.1	Introduction	395
10.2	Polymorphism Examples	397
10.3	Demonstrating Polymorphic Behavior	398
10.4	Abstract Classes and Methods	400
10.5	Case Study: Payroll System Using Polymorphism	403
10.5.1	Abstract Superclass <code>Employee</code>	404
10.5.2	Concrete Subclass <code>SalariedEmployee</code>	407
10.5.3	Concrete Subclass <code>HourlyEmployee</code>	408
10.5.4	Concrete Subclass <code>CommissionEmployee</code>	410
10.5.5	Indirect Concrete Subclass <code>BasePlusCommissionEmployee</code>	412
10.5.6	Polymorphic Processing, Operator <code>instanceof</code> and Downcasting	413

10.5.7	Summary of the Allowed Assignments Between Superclass and Subclass Variables	418
10.6	final Methods and Classes	418
10.7	Case Study: Creating and Using Interfaces	419
10.7.1	Developing a Payable Hierarchy	421
10.7.2	Interface Payable	422
10.7.3	Class Invoice	422
10.7.4	Modifying Class Employee to Implement Interface Payable	425
10.7.5	Modifying Class SalariedEmployee for Use in the Payable Hierarchy	427
10.7.6	Using Interface Payable to Process Invoices and Employees Polymorphically	428
10.7.7	Common Interfaces of the Java API	430
10.8	(Optional) GUI and Graphics Case Study: Drawing with Polymorphism	431
10.9	Wrap-Up	433

11 Exception Handling: A Deeper Look 438

11.1	Introduction	439
11.2	Example: Divide by Zero without Exception Handling	439
11.3	Example: Handling ArithmeticExceptions and InputMismatchExceptions	442
11.4	When to Use Exception Handling	447
11.5	Java Exception Hierarchy	447
11.6	finally Block	450
11.7	Stack Unwinding and Obtaining Information from an Exception Object	454
11.8	Chained Exceptions	457
11.9	Declaring New Exception Types	459
11.10	Preconditions and Postconditions	460
11.11	Assertions	461
11.12	(New in Java SE 7) Multi-catch: Handling Multiple Exceptions in One catch	462
11.13	(New in Java SE 7) try-with-Resources: Automatic Resource Deallocation	463
11.14	Wrap-Up	463

12 ATM Case Study, Part I: Object-Oriented Design with the UML 469

12.1	Case Study Introduction	470
12.2	Examining the Requirements Document	470
12.3	Identifying the Classes in a Requirements Document	478
12.4	Identifying Class Attributes	484
12.5	Identifying Objects' States and Activities	489
12.6	Identifying Class Operations	493
12.7	Indicating Collaboration Among Objects	499
12.8	Wrap-Up	506

13 ATM Case Study Part 2: Implementing an Object-Oriented Design **510**

13.1	Introduction	511
13.2	Starting to Program the Classes of the ATM System	511
13.3	Incorporating Inheritance and Polymorphism into the ATM System	516
13.4	ATM Case Study Implementation	522
13.4.1	Class ATM	523
13.4.2	Class Screen	528
13.4.3	Class Keypad	529
13.4.4	Class CashDispenser	530
13.4.5	Class DepositSlot	531
13.4.6	Class Account	532
13.4.7	Class BankDatabase	534
13.4.8	Class Transaction	537
13.4.9	Class BalanceInquiry	538
13.4.10	Class Withdrawal	539
13.4.11	Class Deposit	543
13.4.12	Class ATMCASEStudy	546
13.5	Wrap-Up	546

14 GUI Components: Part I **549**

14.1	Introduction	550
14.2	Java's New Nimbus Look-and-Feel	551
14.3	Simple GUI-Based Input/Output with JOptionPane	552
14.4	Overview of Swing Components	555
14.5	Displaying Text and Images in a Window	557
14.6	Text Fields and an Introduction to Event Handling with Nested Classes	561
14.7	Common GUI Event Types and Listener Interfaces	567
14.8	How Event Handling Works	569
14.9	JButton	571
14.10	Buttons That Maintain State	574
14.10.1	JCheckBox	574
14.10.2	JRadioButton	577
14.11	JComboBox; Using an Anonymous Inner Class for Event Handling	580
14.12	JList	584
14.13	Multiple-Selection Lists	586
14.14	Mouse Event Handling	589
14.15	Adapter Classes	594
14.16	JPanel Subclass for Drawing with the Mouse	597
14.17	Key Event Handling	601
14.18	Introduction to Layout Managers	604
14.18.1	FlowLayout	605
14.18.2	BorderLayout	608
14.18.3	GridLayout	611
14.19	Using Panels to Manage More Complex Layouts	613

14.20	JTextArea	615
14.21	Wrap-Up	618

15 Graphics and Java 2D 631

15.1	Introduction	632
15.2	Graphics Contexts and Graphics Objects	634
15.3	Color Control	635
15.4	Manipulating Fonts	642
15.5	Drawing Lines, Rectangles and Ovals	647
15.6	Drawing Arcs	651
15.7	Drawing Polygons and Polylines	654
15.8	Java 2D API	657
15.9	Wrap-Up	664

16 Strings, Characters and Regular Expressions 672

16.1	Introduction	673
16.2	Fundamentals of Characters and Strings	673
16.3	Class String	674
16.3.1	String Constructors	674
16.3.2	String Methods length, charAt and getChars	675
16.3.3	Comparing Strings	676
16.3.4	Locating Characters and Substrings in Strings	681
16.3.5	Extracting Substrings from Strings	683
16.3.6	Concatenating Strings	684
16.3.7	Miscellaneous String Methods	684
16.3.8	String Method valueOf	686
16.4	Class StringBuilder	687
16.4.1	StringBuilder Constructors	688
16.4.2	StringBuilder Methods length, capacity, setLength and ensureCapacity	688
16.4.3	StringBuilder Methods charAt, setCharAt, getChars and reverse	690
16.4.4	StringBuilder append Methods	691
16.4.5	StringBuilder Insertion and Deletion Methods	693
16.5	Class Character	694
16.6	Tokenizing Strings	699
16.7	Regular Expressions, Class Pattern and Class Matcher	700
16.8	Wrap-Up	708

17 Files, Streams and Object Serialization 719

17.1	Introduction	720
17.2	Files and Streams	720
17.3	Class File	722
17.4	Sequential-Access Text Files	726
17.4.1	Creating a Sequential-Access Text File	726

17.4.2	Reading Data from a Sequential-Access Text File	733
17.4.3	Case Study: A Credit-Inquiry Program	736
17.4.4	Updating Sequential-Access Files	741
17.5	Object Serialization	742
17.5.1	Creating a Sequential-Access File Using Object Serialization	743
17.5.2	Reading and Deserializing Data from a Sequential-Access File	749
17.6	Additional <code>java.io</code> Classes	751
17.6.1	Interfaces and Classes for Byte-Based Input and Output	751
17.6.2	Interfaces and Classes for Character-Based Input and Output	753
17.7	Opening Files with <code>JFileChooser</code>	754
17.8	Wrap-Up	757

18 Recursion **765**

18.1	Introduction	766
18.2	Recursion Concepts	767
18.3	Example Using Recursion: Factorials	768
18.4	Example Using Recursion: Fibonacci Series	771
18.5	Recursion and the Method-Call Stack	774
18.6	Recursion vs. Iteration	776
18.7	Towers of Hanoi	777
18.8	Fractals	779
18.9	Recursive Backtracking	790
18.10	Wrap-Up	790

19 Searching, Sorting and Big O **798**

19.1	Introduction	799
19.2	Searching Algorithms	800
19.2.1	Linear Search	800
19.2.2	Binary Search	804
19.3	Sorting Algorithms	809
19.3.1	Selection Sort	810
19.3.2	Insertion Sort	814
19.3.3	Merge Sort	817
19.4	Wrap-Up	824

20 Generic Collections **829**

20.1	Introduction	830
20.2	Collections Overview	830
20.3	Type-Wrapper Classes for Primitive Types	831
20.4	Autoboxing and Auto-Unboxing	832
20.5	Interface <code>Collection</code> and Class <code>Collections</code>	832
20.6	Lists	833
20.6.1	<code>ArrayList</code> and <code>Iterator</code>	834
20.6.2	<code>LinkedList</code>	836

20.7	Collections Methods	841
20.7.1	Method <code>sort</code>	842
20.7.2	Method <code>shuffle</code>	845
20.7.3	Methods <code>reverse</code> , <code>fill</code> , <code>copy</code> , <code>max</code> and <code>min</code>	847
20.7.4	Method <code>binarySearch</code>	849
20.7.5	Methods <code>addAll</code> , <code>frequency</code> and <code>disjoint</code>	851
20.8	Stack Class of Package <code>java.util</code>	853
20.9	Class <code>PriorityQueue</code> and Interface <code>Queue</code>	855
20.10	Sets	856
20.11	Maps	859
20.12	<code>Properties</code> Class	863
20.13	Synchronized Collections	866
20.14	Unmodifiable Collections	866
20.15	Abstract Implementations	867
20.16	Wrap-Up	867

21 Generic Classes and Methods 873

21.1	Introduction	874
21.2	Motivation for Generic Methods	874
21.3	Generic Methods: Implementation and Compile-Time Translation	877
21.4	Additional Compile-Time Translation Issues: Methods That Use a Type Parameter as the Return Type	880
21.5	Overloading Generic Methods	883
21.6	Generic Classes	883
21.7	Raw Types	891
21.8	Wildcards in Methods That Accept Type Parameters	895
21.9	Generics and Inheritance: Notes	899
21.10	Wrap-Up	900

22 Custom Generic Data Structures 904

22.1	Introduction	905
22.2	Self-Referential Classes	905
22.3	Dynamic Memory Allocation	906
22.4	Linked Lists	907
22.5	Stacks	917
22.6	Queues	921
22.7	Trees	924
22.8	Wrap-Up	930

23 Applets and Java Web Start 941

23.1	Introduction	942
23.2	Sample Applets Provided with the JDK	943
23.3	Simple Java Applet: Drawing a String	947
23.3.1	Executing <code>WelcomeApplet</code> in the <code>appletviewer</code>	949
23.3.2	Executing an Applet in a Web Browser	951