

Ivor Horton's Beginning Java™ 2, JDK™ 5 Edition

Ivor Horton



WILEY

Wiley Publishing, Inc.

Ivor Horton's Beginning Java™ 2, JDK™ 5 Edition

Ivor Horton's Beginning Java™ 2, JDK™ 5 Edition

Ivor Horton



WILEY

Wiley Publishing, Inc.

Ivor Horton's Beginning Java™ 2, JDK™ 5 Edition

Published by

Wiley Publishing, Inc.

10475 Crosspoint Boulevard

Indianapolis, IN 46256

www.wiley.com

Copyright © 2005 by Ivor Horton

Published by Wiley Publishing, Inc., Indianapolis, Indiana

Published simultaneously in Canada

ISBN: 0-7645-6874-4

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

5B/RU/RS/QU/IN

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600. Requests to the Publisher for permission should be addressed to the Legal Department, Wiley Publishing, Inc., 10475 Crosspoint Blvd., Indianapolis, IN 46256, (317) 572-3447, fax (317) 572-4355, e-mail: brandreview@wiley.com.

LIMIT OF LIABILITY/DISCLAIMER OF WARRANTY: THE PUBLISHER AND THE AUTHOR MAKE NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE ACCURACY OR COMPLETENESS OF THE CONTENTS OF THIS WORK AND SPECIFICALLY DISCLAIM ALL WARRANTIES, INCLUDING WITHOUT LIMITATION WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE. NO WARRANTY MAY BE CREATED OR EXTENDED BY SALES OR PROMOTIONAL MATERIALS. THE ADVICE AND STRATEGIES CONTAINED HEREIN MAY NOT BE SUITABLE FOR EVERY SITUATION. THIS WORK IS SOLD WITH THE UNDERSTANDING THAT THE PUBLISHER IS NOT ENGAGED IN RENDERING LEGAL, ACCOUNTING, OR OTHER PROFESSIONAL SERVICES. IF PROFESSIONAL ASSISTANCE IS REQUIRED, THE SERVICES OF A COMPETENT PROFESSIONAL PERSON SHOULD BE SOUGHT. NEITHER THE PUBLISHER NOR THE AUTHOR SHALL BE LIABLE FOR DAMAGES ARISING HEREFROM. THE FACT THAT AN ORGANIZATION OR WEBSITE IS REFERRED TO IN THIS WORK AS A CITATION AND/OR A POTENTIAL SOURCE OF FURTHER INFORMATION DOES NOT MEAN THAT THE AUTHOR OR THE PUBLISHER ENDORSES THE INFORMATION THE ORGANIZATION OR WEBSITE MAY PROVIDE OR RECOMMENDATIONS IT MAY MAKE. FURTHER, READERS SHOULD BE AWARE THAT INTERNET WEBSITES LISTED IN THIS WORK MAY HAVE CHANGED OR DISAPPEARED BETWEEN WHEN THIS WORK WAS WRITTEN AND WHEN IT IS READ.

For general information on our other products and services or to obtain technical support, please contact our Customer Care Department within the U.S. at (800) 762-2974, outside the U.S. at (317) 572-3993 or fax (317) 572-4002.

For technical support, please visit www.wiley.com/techsupport.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

Library of Congress Cataloging-in-Publication Data

Horton, Ivor.

Ivor Horton's Beginning Java 2, JDK 5 Edition / Ivor Horton.

p. cm.

Includes index.

ISBN 0-7645-6874-4 (paper/website)

1. Java (Computer program language) I. Title: *Ivor Horton's Beginning Java 2, JDK 5 Edition*. II. Title.

QA76.73.J38H6758 2004

005.13'3—dc22

2004017036

Trademarks: Wiley, the Wiley Publishing logo, Wrox, the Wrox logo, Programmer to Programmer, and related trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries, and may not be used without written permission. Java and JDK are trademarks of Sun Microsystems, Inc. All other trademarks are the property of their respective owners. Wiley Publishing, Inc., is not associated with any product or vendor mentioned in this book.

About the Author

Ivor Horton started out as a mathematician, but shortly after graduating, he was lured into messing about with computers by a well-known manufacturer. He has spent many happy years programming occasionally useful applications in a variety of languages as well as teaching mainly scientists and engineers to do likewise. He has extensive experience in applying computers to problems in engineering design and to manufacturing operations in a wide range of industries. He is the author of a number of tutorial books on programming in C, C++, and Java. When not writing programming books or providing advice to others, he leads a life of leisure.

Credits

Executive Editor

Robert Elliott

Senior Development Editor

Kevin Kent

Technical Editors

Calvin Austin, J2SE 5.0 Specification Lead, Sun Microsystems

Wiley-Dreamtech India Pvt Ltd

Production Editor

William A. Barton

Copy Editor

Luann Rouff

Editorial Manager

Mary Beth Wakefield

Vice President & Executive Group Publisher

Richard Swadley

Vice President and Publisher

Joseph B. Wikert

Project Coordinator

Erin Smith

Graphics and Production Specialists

Karl Brandt

Jonelle Burns

Kelly Emkow

Carrie Foster

Lauren Goddard

Denny Hager

Joyce Haughey

Jennifer Heleine

Ron Terry

Quality Control Technicians

Joe Niesen

Susan Moritz

Brian H. Walls

Media Development Specialist

Travis Silvers

Proofreading and Indexing

TECHBOOKS Production Services

Cover Photograph

© Ian Capener

Foreword

You are probably reading this foreword with one of several things in mind. First, is this the right book for me, is the material current, and does the text reflect the final API? Second, what should I expect to learn and where should I start reading a book of this length?

Many of the forewords I have seen will lead you through an amusing anecdote or story and then mention a little about the author, but then fail to leave you any wiser about the answer to those questions. So, to get straight to the point and to answer the second question first, this is a book that you can start from page one and read right through to the end. If you haven't read any of Ivor Horton's books before, you are in for a pleasant surprise. Ivor's style is very accessible, which makes the content easy to follow and understand. I know, because I have read this book from cover to cover.

This edition of *Ivor Horton's Beginning Java 2, JDK 5 Edition* is based on the J2SE 5.0 platform. The J2SE 5.0 release is one of the most significant updates to the Java platform in many years and has been three years in the making. The release involved 160 experts worldwide, all working through the Java Community Process and focused on making the platform better for all developers. I have been involved with the project since day one as the Specification Lead for JSR 176, which defines the contents of J2SE 5.0. As such, I had a great interest in making sure that this book is accurate and matches the final API set. I've even compiled and run every code example twice, and there are a lot of great examples, as you will find out.

So what can you expect to learn from this new edition? First, Ivor covers the basic programming blocks and gets you started with your first Java program. Ivor then introduces the Java language changes step by step, including the new generic types, the enhanced `for` loop, enumerated types, and many others. You will also get to use the new language changes in later chapters and learn some of the other non-language features, such as XML DOM3 updates. So whether you are a new developer or already have some Java programming experience, you will gain the skills needed to work with the latest Java release.

In closing, I encourage you to read and enjoy what JDK 5.0 has to offer and find out how easy using J2SE 5.0 really is.

Calvin Austin
J2SE 5.0 Specification Lead
Sun Microsystems

Acknowledgments

While a book is usually attributed to the author, a book — particularly a large book such as this — is always the result of the efforts of a sizeable team of people. I'd therefore like to thank all the editorial and production staff at Wiley who worked so hard to produce this fifth edition of my Java tutorial from my initial draft.

I'd especially like to thank Calvin Austin of Sun Microsystems for his extensive technical input. He somehow found the time to go through the complete text and try out all the examples — twice — in spite of the considerable demands of his day job. Calvin's suggestions for improvements were invaluable, as was his ability to spot my mistakes, and I'm particularly grateful for his indications of where I'd missed some of the inevitable changes arising from the evolution of the J2SE 5.0 API during the beta phases. Any errors that remain are, of course, my own, but there are surely fewer of them as a consequence of Calvin's efforts.

I'd also like to thank readers of past editions of *Ivor Horton's Beginning Java* for their numerous suggestions for corrections and improvements. In addition to the many changes that I made in response to these, I also updated and reintroduced the chapters on using JDBC that were omitted from the previous edition, in response to requests from a number of readers. The greatly increased page count of this edition over the previous edition is only in part a consequence of restoring the JDBC tutorial. The bulk of the additional page count is attributable to new material relating to the features introduced by J2SE 5.0 that deliver exciting new capabilities for every Java programmer. The J2SE 5.0 release is truly a major step forward that encompasses important extensions to the Java language as well as major additions to the class libraries.

Finally I'd like to thank my wife, Eve, who provides unstinting support for whatever I happen to be doing and cheerfully accepts my complaints about the workload that I freely elected to undertake. She always manages to be on hand whenever I need sustenance or sympathy, or both, and undoubtedly I would never have finished this book without her.

Ivor Horton

Contents

About the Author	v
Foreword	ix
Acknowledgments	xi
Introduction	xxxvii
Chapter 1: Introducing Java	1
What Is Java All About?	1
Features of the Java Language	2
Learning Java	3
Java Programs	3
Learning Java — The Road Ahead	3
The Java Environment	4
Java Program Development	5
Installing the JDK	6
Compiling a Java Program	8
Executing a Java Application	9
Executing an Applet	10
Object-Oriented Programming in Java	12
So What Are Objects?	13
What Defines a Class of Objects?	14
Operating on Objects	17
Java Program Statements	19
Encapsulation	20
Classes and Data Types	20
Classes and Subclasses	21
Advantages of Using Objects	21
Java Program Structure	21
Java's Class Library	22
Java Applications	24
Java and Unicode	27
Summary	27
Resources	28

Chapter 2: Programs, Data, Variables, and Calculation	29
Data and Variables	29
Naming Your Variables	30
Variable Names and Unicode	31
Variables and Types	31
Integer Data Types	31
Integer Literals	33
Declaring Integer Variables	34
Floating-Point Data Types	36
Floating-Point Literals	36
Declaring Floating-Point Variables	37
Fixing the Value of a Variable	37
Arithmetic Calculations	38
Integer Calculations	39
Producing Output	44
Integer Division and Remainders	45
The Increment and Decrement Operators	46
Computation with Shorter Integer Types	48
Errors in Integer Arithmetic	49
Floating-Point Calculations	49
Other Floating-Point Arithmetic Operators	50
Error Conditions in Floating-Point Arithmetic	51
Mixed Arithmetic Expressions	51
Explicit Casting	52
Automatic Type Conversions in Assignments	52
The op= Operators	53
Mathematical Functions and Constants	54
Importing the Math Class Methods	59
Storing Characters	60
Character Escape Sequences	60
Character Arithmetic	61
Bitwise Operations	63
Using the AND and OR Operators	65
Using the Exclusive OR Operator	68
Shift Operations	70
Methods for Bitwise Operations	74
Variables with a Fixed Set of Integer Values	77
Boolean Variables	79
Operator Precedence	80
Program Comments	81
Documentation Comments	82

Summary	83
Exercises	84
Chapter 3: Loops and Logic	85
Making Decisions	85
Making Comparisons	86
The if Statement	87
Statement Blocks	88
The else Clause	90
Nested if Statements	91
Comparing Enumeration Values	94
Logical Operators	95
Logical AND Operations	95
&& versus &	97
Logical OR Operations	98
Boolean NOT Operations	98
Character Testing Using Standard Library Methods	99
The Conditional Operator	100
The switch Statement	102
The General Case of the switch Statement	104
Variable Scope	108
Loops	111
Varieties of Loop	112
Counting Using Floating-Point Values	117
Nested Loops	121
The continue Statement	123
The Labeled continue Statement	123
Using the break Statement in a Loop	124
Breaking Indefinite Loops	127
The Labeled break Statement	128
Assertions	130
More Complex Assertions	132
Summary	133
Exercises	134
Chapter 4: Arrays and Strings	135
Arrays	135
Array Variables	136
Defining an Array	136
The Length of an Array	137

Contents

Accessing Array Elements	138
Reusing Array Variables	138
Initializing Arrays	139
Using a Utility Method to Initialize an Array	140
Initializing an Array Variable	141
Using Arrays	142
Using the Collection-Based for Loop with an Array	143
Arrays of Arrays	145
Arrays of Arrays of Varying Length	149
Multidimensional Arrays	151
Arrays of Characters	152
Strings	152
String Literals	153
Creating String Objects	153
Arrays of Strings	155
Operations on Strings	157
Joining Strings	157
Comparing Strings	161
Comparing Strings for Equality	163
String Interning	166
Checking the Start and End of a String	167
Sequencing Strings	167
Accessing String Characters	169
Extracting String Characters	170
Searching Strings for Characters	172
Searching for Substrings	173
Extracting Substrings	177
Tokenizing a String	179
Modified Versions of String Objects	182
Creating Character Arrays from String Objects	182
Using the Collection-Based for Loop with a String	183
Obtaining the Characters in a String as an Array of Bytes	183
Creating String Objects from Character Arrays	184
Mutable Strings	184
Creating StringBuffer Objects	185
The Capacity of a StringBuffer Object	186
Changing the String Length for a StringBuffer Object	188
Adding to a StringBuffer Object	189
Appending a Substring	189
Appending Basic Types	190
Finding the Position of a Substring	191
Replacing a Substring in the Buffer	192
Inserting Strings	192

Extracting Characters from a Mutable String	193
Other Mutable String Operations	193
Creating a String Object from a StringBuffer Object	194
Summary	196
Exercises	197
 Chapter 5: Defining Classes	 199
What Is a Class?	200
Fields in a Class Definition	200
Methods in a Class Definition	202
Accessing Variables and Methods	203
Defining Classes	204
Defining Methods	205
Returning from a Method	206
The Parameter List	206
How Argument Values Are Passed to a Method	208
Final Parameters	209
Defining Class Methods	209
Accessing Class Data Members in a Method	209
The Variable this	210
Initializing Data Members	211
Using Initialization Blocks	212
Constructors	215
The Default Constructor	216
Creating Objects of a Class	217
Passing Objects to a Method	218
The Lifetime of an Object	219
Defining and Using a Class	220
Method Overloading	222
Multiple Constructors	223
Calling a Constructor from a Constructor	225
Duplicating Objects Using a Constructor	226
Using Objects	227
Creating a Point from Two Lines	230
Recursion	233
Understanding Packages	236
Packaging Up Your Classes	237
Packages and the Directory Structure	237
Compiling a Package	238
Accessing a Package	239
Using Extensions	240
Adding Classes from a Package to Your Program	241

Contents

Packages and Names in Your Programs	241
Importing Static Class Members	242
Standard Packages	243
Standard Classes Encapsulating the Primitive Data Types	244
Controlling Access to Class Members	246
Using Access Attributes	246
Specifying Access Attributes	248
Choosing Access Attributes	250
Using Package and Access Attributes	251
Nested Classes	256
Static Nested Classes	257
Using a Non-Static Nested Class	262
Using a Nested Class Outside the Top-Level Class	263
Local Nested Classes	264
The finalize() Method	265
Native Methods	266
Summary	266
Exercises	267
Chapter 6: Extending Classes and Inheritance	269
Using Existing Classes	269
Class Inheritance	271
Inheriting Data Members	272
Hidden Data Members	273
Inherited Methods	273
Objects of a Derived Class	274
Deriving a Class	275
Derived Class Constructors	275
Calling the Base Class Constructor	276
Overriding a Base Class Method	277
Choosing Base Class Access Attributes	279
Polymorphism	279
Using Polymorphism	282
Multiple Levels of Inheritance	286
Abstract Classes	287
The Universal Superclass	288
The toString() Method	289
Determining the Type of an Object	289
Copying Objects	291
Methods Accepting a Variable Number of Arguments	295
Limiting the Types in a Variable Argument List	297

Casting Objects	298
When to Cast Objects	300
Identifying Objects	301
More on Enumerations	302
Adding Members to an Enumeration Class	303
Designing Classes	307
A Classy Example	307
Designing the PolyLine Class	309
A General-Purpose Linked List	313
Using the final Modifier	317
Interfaces	318
Encapsulating Constants in a Program	319
Constants in an Interface	320
Constants Defined in a Class	321
Interfaces Declaring Methods	323
A Partial Interface Implementation	324
Extending Interfaces	325
Interfaces and Multiple Inheritance	326
Using Interfaces	326
Interfaces and Polymorphism	327
Using Multiple Interfaces	333
Method Parameters of Interface Types	333
Nesting Classes in an Interface Definition	334
Interfaces and the Real World	334
Anonymous Classes	335
Summary	335
Exercises	337
 Chapter 7: Exceptions	 339
The Idea Behind Exceptions	339
Types of Exceptions	340
Error Exceptions	341
RuntimeException Exceptions	342
Other Subclasses of Exception	343
Dealing with Exceptions	344
Specifying the Exceptions a Method Can Throw	344
Handling Exceptions	345
The try Block	345
The catch Block	345
try catch Bonding	347
Multiple catch Blocks	349
The finally Block	350

Contents

Structuring a Method	351
Execution Sequence	352
Normal Execution of a Method	355
Execution When an Exception Is Thrown	356
Execution When an Exception Is Not Caught	357
Nested try Blocks	358
Rethrowing Exceptions	359
Exception Objects	359
The Throwable Class	359
Standard Exceptions	362
Defining Your Own Exceptions	362
Defining an Exception Class	363
Throwing Your Own Exception	363
An Exception Handling Strategy	364
An Example of an Exception Class	364
Summary	368
Exercises	368
 Chapter 8: Understanding Streams	 371
 Streams and the New I/O Capability	 371
Understanding Streams	372
Input and Output Streams	373
Binary and Character Streams	374
The Classes for Input and Output	375
Basic Input Stream Operations	375
Basic Output Stream Operations	379
Stream Readers and Writers	379
Using Readers	381
Using Writers	382
The Standard Streams	384
Getting Data from the Keyboard	384
Tokenizing a Stream	385
Customizing a Stream Tokenizer	387
Writing to the Command Line	392
The printf() Method	392
Formatting Numerical Data	394
Specifying the Width and Precision	395
Formatting Characters and Strings	396
The Locale Class	397
Formatting Data into a String	398
Summary	399
Exercises	399

Chapter 9: Accessing Files and Directories	401
Working with File Objects	401
Creating File Objects	402
Portable Path Considerations	404
Absolute and Relative Paths	404
Accessing System Properties	405
Setting System Properties	407
Testing and Checking File Objects	408
Querying Files and Directories	409
Filtering a File List	414
Creating and Modifying Files and Directories	417
Creating File Output Streams	419
Ensuring a File Exists	421
Avoiding Overwriting a File	423
FileDescriptor Objects	424
Summary	425
Exercises	425
Chapter 10: Writing Files	427
File I/O Basics	427
File Input and Output	429
Channels	430
Channel Operations	431
File Channels	433
Buffers	434
Buffer Capacity	434
Buffer Position and Limit	435
Setting the Position and Limit	437
Creating Buffers	438
View Buffers	439
Duplicating and Slicing Buffers	441
Creating Buffers by Wrapping Arrays	443
Wrapping Strings	445
Marking a Buffer	446
Buffer Data Transfers	446
Transferring Data into a Buffer	447
Using View Buffers	449
Preparing a Buffer for Output to a File	449
Writing to a File	451
File Position	453
Using a View Buffer to Load Data into a Byte Buffer	458

Contents

Writing Varying Length Strings to a File	460
Using a Formatter Object to Load a Buffer	462
Direct and Indirect Buffers	466
Writing Numerical Data to a File	467
Writing Mixed Data to a File	471
Gathering-Write Operations	477
Summary	481
Exercises	482
 Chapter 11: Reading Files	 483
 File Read Operations	 483
Creating File Input Streams	484
File Channel Read Operations	485
Reading a Text File	488
Getting Data from the Buffer	489
Reading Binary Data	491
Reading Mixed Data	496
Compacting a Buffer	499
Copying Files	502
Random Access to a File	507
Read/Write Operations with a Single File Channel	512
Memory-Mapped Files	513
Locking a File	517
Locking Part of a File	519
Practical File Locking Considerations	519
Summary	523
Exercises	524
 Chapter 12: Serializing Objects	 525
 Storing Objects in a File	 525
Writing an Object to a File	526
Writing Basic Data Types to an Object Stream	528
Implementing the Serializable Interface	529
Conditions for Serialization	532
Transient Data Members of a Class	533
Reading an Object from a File	533
Determining the Class of a Deserialized Object	537
Reading Basic Data from an Object Stream	538
Using Object Serialization	538
Serializing Classes Yourself	541

Serialization Problems and Complications	542
Resetting an Object Output Stream	544
Summary	545
Exercises	546
Chapter 13: Generic Class Types	547
What Are Generic Types?	547
Defining a Generic Class Type	548
Implementing a Generic Type	550
Instantiating a Generic Type	551
Using Primitive Type Wrapper Class Types as Arguments	555
The Runtime Type of Generic Type Instances	557
Relationships between Generic Type Instances	559
Multiple Type Parameters	559
Type Parameter Scope	560
Static Fields in a Generic Type	560
Type Parameter Bounds	561
Generic Types and Generic Interfaces	565
Enabling the Collection-Based for Loop	565
Implementing an Iterator Capability	567
A Parameterized Type for Binary Trees	569
Defining the Generic Type	571
Hidden Constraints in the BinaryTree<> Type	579
Variables of a Raw Type	580
Using Wildcards as Type Parameter Arguments	582
Constraints on a Wildcard	584
More on the Class Class	587
Arrays and Parameterized Types	588
Parameterized Methods	592
Generic Constructors	595
Parameterized Types and Inheritance	598
Summary	599
Exercises	600
Chapter 14: The Collections Framework	601
Understanding the Collections Framework	601
Collections of Objects	602
Sets	603
Sequences	604
Maps	605
Hashing	606

Contents

Iterators	606
List Iterators	608
Collection Classes	610
Collection Interfaces	614
Using Vectors	615
Creating a Vector	616
The Capacity and Size of a Vector	618
Storing Objects in a Vector	620
Retrieving Objects from a Vector	621
Accessing Elements in a Vector through a List Iterator	621
Extracting All the Elements from a Vector	622
Removing Objects from a Vector	623
Searching a Vector	625
Applying Vectors	626
Sorting a Collection	630
Stack Storage	632
Linked Lists	638
Using Maps	640
The Hashing Process	640
Using Your Own Class Objects as Keys	642
Generating Hashcodes	642
Creating a HashMap Container	643
Storing, Retrieving, and Removing Objects	644
Processing all the Elements in a Map	646
Summary	657
Exercises	658
 Chapter 15: A Collection of Useful Classes	 659
 Utility Methods for Arrays	 659
Filling an Array	660
Comparing Arrays	661
Sorting Arrays	662
Searching Arrays	666
Observable and Observer Objects	670
Defining Classes of Observable Objects	671
Observable Class Methods	671
Generating Random Numbers	675
Random Operations	676
Dates and Times	678
The Date Class	679
Interpreting Date Objects	679
Obtaining a Date Object from a String	684

Gregorian Calendars	684
Setting the Date and Time	686
Getting Date and Time Information	687
Modifying Dates and Times	688
Comparing Calendars	688
Regular Expressions	691
Defining Regular Expressions	691
Creating a Pattern	692
Creating a Matcher	693
Searching a String	694
Matching an Entire String	696
Defining Sets of Characters	697
Matching Boundaries	700
Using Quantifiers	701
Tokenizing a String	703
Search and Replace Operations	705
Using Capturing Groups	708
Juggling Captured Text	710
Using a Scanner	714
Creating Scanner Objects	714
Getting Input from a Scanner	715
Testing for Tokens	717
Defining Your Own Patterns for Tokens	718
Summary	720
Exercises	721
 Chapter 16: Threads	 723
Understanding Threads	723
Creating Threads	726
Stopping a Thread	731
Connecting Threads	733
Thread Scheduling	733
Implementing the Runnable Interface	734
Managing Threads	736
Synchronization	737
Synchronized Methods	737
Synchronizing Statement Blocks	749
Deadlocks	755
Communicating between Threads	756
Using wait() and notifyAll() in the Bank Program	758
Thread Priorities	761
Using Thread Priorities	762

Contents

Summary	765
Exercises	766
 Chapter 17: Creating Windows	 767
Graphical User Interfaces in Java	767
Model-View-Controller (MVC) Architecture	768
Creating a Window	770
Components and Containers	775
Window and Frame Components	776
Window Panes	777
Basics of Components	779
Component Attributes	779
The Size and Position of a Component	780
Points and Rectangles	784
Point Objects	784
Rectangle Objects	785
Visual Characteristics of a Component	788
Defining Color	789
System Colors	791
Creating Cursors	791
Selecting Fonts	792
Swing Components	797
Buttons	798
Menus	799
Text Components	800
Other Swing Components	800
Using Containers	801
Adding Components to a Container	802
Container Layout Managers	803
The Flow Layout Manager	805
Changing the Gap	807
Using a Border Layout Manager	811
Using a Card Layout Manager	813
Using a Grid Layout Manager	815
Using a BoxLayout Manager	817
Struts and Glue	820
Using a GridBagLayout Manager	825
GridBagConstraints Instance Variables	826
Using a SpringLayout Manager	834
Understanding Constraints	835
Defining Constraints	836
Setting Constraints for a Component	837

Adding a Menu to a Window	843
Creating JMenu and JMenuitem	843
Creating a Menu	844
Adding Menu Items to a Menu	847
Adding a Shortcut for a Menu Item	852
More on Applets	854
Converting an Application to an Applet	856
Summary	857
Exercises	858
 Chapter 18: Handling Events	 861
Window-Based Java Programs	861
Event-Driven Programs	862
The Event-Handling Process	863
Avoiding Deadlocks in GUI Code	865
Event Classes	867
Low-Level Event Classes	868
Making a Window Handle Its Own Events	870
Enabling Other Low-level Events	873
Low-Level Event Listeners	874
The WindowListener Interface	875
The WindowFocusListener Interface	875
The WindowStateListener Interface	875
The MouseListener Interface	876
The MouseMotionListener Interface	876
The MouseWheelListener Interface	876
The KeyListener Interface	876
The FocusListener Interface	877
Using Adapter Classes	879
Semantic Events	882
Semantic Event Listeners	883
Semantic Event Handling in Applets	884
Alternative Event-Handling Approaches	893
Handling Low-Level and Semantic Events	895
Semantic Event Listeners in an Application	896
Listening to Menu Items	896
Fixing the Color Menu Check Marks	902
Using Actions	902
The Action Interface	903
Using Actions as Menu Items	905
Defining Action Classes	906

Contents

Adding a Toolbar	911
Adding Buttons to a Toolbar	912
Adding Icons	914
Fixing the Menus	918
Adding Tooltips	920
Disabling Actions	922
Summary	924
Exercises	924
Chapter 19: Drawing in a Window	927
Using the Model/View Architecture	927
Coordinate Systems in Components	931
Drawing on a Component	933
Graphics Contexts	934
The Drawing Process	937
Rendering Operations	938
Shapes	939
Classes Defining Points	939
Lines and Rectangles	941
Combining Rectangles	943
Testing Rectangles	944
Arcs and Ellipses	947
Curves	950
Complex Paths	960
Filling Shapes	966
Gradient Fill	968
Managing Shapes	972
Storing Shapes in the Model	974
Drawing Shapes	975
Drawing Using the Mouse	976
Handling Mouse Events	977
Handling Mouse Button Press Events	979
Using XOR Mode	980
Handling Mouse Dragging Events	981
Handling Button Release Events	983
Locating the Mouse Cursor Using MouseEvent Class Methods	985
Defining Your Own Shape Classes	985
Defining Lines	986
Defining Rectangles	988
Defining Circles	990
Drawing Curves	993