



Introduction to Programming with Java™

Outsource Laboratories Press™

Copyright © 2002 - 2003 Outsource Laboratories Press

Printed in the United States of America

ISBN 0-9725199-3-9 Hard Cover

1 2 3 4 5 6 7 8 9 10 09 08 07 06 05 04 03

OUTSOURCE LABORATORIES PRESS
PO Box 187, Matawan, NJ 07747-0187 USA

For more information about Outsource Laboratories Press' textbooks and services, contact us:

Toll free: 888-GO-OLABS (888-466-5227)

Email: training@olabs.com

<http://www.olabs.com>

Technical Support: Outsource Laboratories will offer limited technical support to instructors who have purchased classroom sets and have registered with Outsource Laboratories. Contact your representative for more information.

All rights reserved. This product and related documentation is protected by copyright and distributed under license restricting its use, copying, distribution, and decompilation. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without prior written authorization from the publisher.

The authors and publisher have taken care in preparation of this book, but make no expressed or implied warranty of any kind and assume no responsibility for any errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the information contained in this book.

All other products referenced herein are trademarks of their respective holders.

Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

5203

Table of Contents

About Olabs' Text xvii

1 Basics of Programming

Introduction	1-1
What is Programming?	1-1
Steps for Developing a Program	1-2
Step 1: Identify a Task or Problem	1-3
Step 2: Outline the Solution by Writing an Algorithm	1-4
Step 3: Implement the Algorithm using a Programming Language	1-4
Step 4: Compile the Program	1-5
Step 5: Run the Program	1-5
Step 6: Maintain the Program	1-5
Writing Algorithms with Pseudocode	1-5
Basic Actions Used in Programs	1-6
Taking Input	1-6
Providing Output	1-7
Performing Arithmetic	1-7
Storing a Value	1-7
Choosing between Alternative Actions	1-7
Repeating Actions	1-8
Example Programs in Pseudocode	1-9
Example 1: Conversion of Fahrenheit to Celsius	1-9
Example 2: Calculate Total Cost Including Sales Tax	1-9
Example 3: Provide the Name of the Month	1-10
Compiling and Running a Java Program	1-11
Using the DOS Prompt	1-11
Starting a Command Prompt Window	1-11
Changing into a Different Folder	1-12
Listing All Files in a Folder	1-12
A few Words About Files and Filenames	1-14
Compiling a Java Program	1-14
Running a Java Program	1-15
Exercise 1-1: Compiling and Running Java Programs	1-16
Exercise 1-2: Create a Simple Java Application	1-17
Review Questions	1-18

2 Java Programming 2-1

Introduction	2-1
Facts about the Java Language	2-2
How Does a Computer Work?	2-3

Data	2-3
Operations	2-4
Assembly Language	2-5
Statements	2-6
Compilers	2-7
Java Virtual Machines	2-8
Java Development Tools	2-9
Java Compiler: javac	2-9
Java Interpreter: java	2-10
Exercise 2-1: Compile and Execute a Java Program	2-12
Java Applications, Applets and Servlets	2-12
Applications	2-13
Console Applications	2-13
GUI Applications	2-14
Embedded Applications	2-14
Applets	2-15
Servlets	2-16
JVM and Applications, Applets, Servlets	2-17
The Concept of Modularity	2-18
Basic Structure of a Java Program	2-19
Exercise 2-2: The main Method	2-20
Comments in Java	2-20
Exercise 2-3: Using Comments	2-22
Exercise 2-4: Create Another Simple Java Application	2-22
Review Questions	2-23

3 Variables, Data Types and Operators 3-1

Introduction	3-1
The Add2Numbers Program	3-2
Running the Program	3-3
Statements and Expressions	3-4
Variables	3-5
Declaring Variables	3-6
Assigning Values to Variables	3-7
Variable Names	3-8
Exercise 3-1: Find Valid Variable Names	3-9
Data Types	3-9
Primitive vs. Reference Data Types	3-10
Why Different Data Types?	3-11
Java's Main Data Types	3-12
The Data Type 'int'	3-12
Exercise 3-2: Declare a Variable of Type int	3-13
Exercise 3-3: Assigning Values to Variables of Type int	3-14
The LearningIO Class	3-15
Packages	3-15
Setting LearningIO in the Classpath	3-17

Directory Structure of LearningIO	3-18
Using Standard Java I/O	3-19
Partial Member Listing for the LearningIO Class	3-20
Exercise 3-4: Using Learning IO to Read Numbers	3-21
Exercise 3-5: Using Learning IO to Read Strings	3-21
The Data Type ‘double’	3-22
Exercise 3-6: Declare a Variable of Type double	3-24
Exercise 3-7: Assigning Values to Variables of Type double	3-25
Assigning a Value When Declaring a Variable	3-26
The Data Type ‘char’	3-27
Exercise 3-8: Declare a Variable of Type char	3-29
Exercise 3-9: Assigning Values to Variables of Type char	3-30
Exercise 3-10: Identify Values of Type char	3-31
Characters and Their Associated Number Code	3-32
The Data Type ‘String’	3-32
Printing Strings	3-33
Shortcut for Printing Strings	3-34
Exercise 3-11: Using String Variables	3-35
Exercise 3-12: Writing a Program	3-36
Exercise 3-13: Changing the String Value	3-37
The Data Type ‘boolean’	3-37
Exercise 3-14: Declare a Variable of Type boolean	3-39
Exercise 3-15: Assigning Values to Variables of Type boolean	3-40
Exercise 3-16: Map Values to Variables	3-40
Literals	3-41
Type Casting	3-43
Type Conversion	3-43
Constants	3-44
Arithmetic Operators	3-45
Addition Operator (+)	3-46
Subtraction Operator (-)	3-48
Multiplication Operator (*)	3-50
Division Operator (/)	3-52
Modulo Operator (%)	3-54
Exercise 3-17: Using Operators	3-56
Exercise 3-18: Bills Calculator Program	3-58
Exercise 3-19: Calculating the Average Mark	3-59
Exercise 3-20: Calculate the Weekly Salary	3-60
Arithmetic/Assignment Operators	3-61
Exercise 3-21: Using Arithmetic/Assignment Operators	3-62
Increment and Decrement Operators	3-63
Exercise 3-22: Side Effects of using the Decrement Operator	3-64
Order of Operations - Precedence of Operators	3-65
Exercise 3-23: Change the Order of Operations	3-66
Review Questions	3-67

4	Flow Control - Selection	4-1
	Introduction	4-1
	What is Flow Control?	4-2
	Selection Statements	4-2
	Comparison Operators	4-4
	Exercise 4-1: Using Comparison Operators	4-5
	Logical Operators	4-6
	AND Operator	4-6
	OR Operator	4-6
	NOT Operator	4-7
	Exercise 4-2: Using Logical Operators	4-8
	Types of if-statements	4-9
	The if/else statement	4-9
	If-statement without else	4-14
	Exercise 4-3: Using an If/Else-Statement	4-17
	Exercise 4-4: Reef Cruise Booking System	4-19
	Multiple if-statements	4-20
	Exercise 4-5: Understanding Multiple If-Statements	4-23
	Exercise 4-6: Fastfood Ordering System	4-24
	The Switch Statement	4-27
	Exercise 4-7: Using a Switch Statement	4-32
	Review Questions	4-33
5	Flow Control - Repetition	5-1
	Introduction	5-1
	Repetition Statements	5-2
	Counter v. Sentinel- Controlled Loops	5-3
	The While Loop	5-4
	Exercise 5-1: Keeping Track of Variables In a While Loop	5-7
	Exercise 5-2: Using a While Loop to Count Integer Values	5-8
	Exercise 5-3: Debt Payment Program	5-9
	The Do-While Loop	5-11
	Exercise 5-4: Average Marks Revisited	5-14
	The For Loop	5-15
	Exercise 5-5: Using For Loop to Sum Numbers	5-18
	Exercise 5-6: Selecting the Appropriate Loop	5-19
	Review Questions	5-20
6	Methods	6-1
	Introduction	6-1
	Modularity: The Reason Behind Methods	6-2
	Basic Structure of a Method	6-6
	Exercise 6-1: Modularizing a Program	6-7

	Passing Parameters to a Method	6-8
	What are Parameters?	6-8
	Passing Parameters to a Method	6-9
	Return Type of a Method	6-10
	Method with Return Type	6-10
	Exercise 6-2: Identifying Parts of a Method	6-12
	Exercise 6-3: Calling Methods I	6-13
	Using Methods Inside a Statement as an Operand	6-14
	Exercise 6-4: Reading Method Signatures	6-15
	Modularizing a Program	6-16
	Example 1 - CostCalculator	6-16
	Example 2 - Convert Inches to Centimeters	6-18
	Exercise 6-5: Calling Methods II	6-20
	Exercise 6-6: Converting Currencies	6-21
	Review Questions	6-22
7	Strings	7-1
	Introduction	7-1
	Strings	7-2
	Creating a String	7-2
	Escape Sequences	7-3
	Exercise 7-1: Escape Sequences	7-4
	Immutability of Strings	7-5
	String Methods	7-6
	Exercise 7-2: Using the java.lang.String methods	7-8
	Common String Manipulations	7-9
	Concatenation	7-9
	Exercise 7-3: Concatenation v. Addition	7-10
	Exercise 7-4: Concatenating Static and Dynamic Strings	7-11
	Substrings	7-12
	Exercise 7-5: Substrings	7-14
	Comparisons	7-15
	Exercise 7-6: Testing Strings for Equality	7-17
	String Buffers	7-17
	Declaring and Creating a String Buffer	7-17
	String Buffer Methods	7-19
	Common String Buffer Manipulations	7-20
	Exercise 7-7: Modifying String Buffers	7-21
	Review Questions	7-22
8	Using Arrays	8-1
	Introduction	8-1
	What is an Array?	8-1
	Declaring an Array Variable	8-4

Creating an Array	8-5
Array Values Known at Compile Time	8-5
Exercise 8-1: Creating an Array	8-6
Array Values Known at Run Time	8-7
Assigning Values to an Array	8-8
Assigning a Value to an Array Element	8-8
Variables/Expressions as Indices	8-9
Exercise 8-2: Identifying Array Elements	8-10
Retrieving Values from an Array	8-11
Retrieving an Array Element	8-11
Retrieving All Array Elements	8-12
The length Property	8-14
Common Problem: Array Index Out of Bounds	8-15
Example Array Scenarios	8-16
Scenario 1: Representing Rows of People in a Theater	8-16
Scenario 2: Weekday Translation Program	8-18
Exercise 8-3: The Weekdays Program	8-21
Scenario 3: Summing Up an Array of Prices	8-22
Scenario 4: Finding Lowest Price in Array of Prices	8-24
Scenario 5: Storing Student Names	8-26
Exercise 8-4: Storing Student Marks	8-29
Application Command Line Parameters	8-30
Exercise 8-5: Fibonacci Numbers	8-31
Multi-dimensional Arrays	8-32
Declaring a Multi-Dimensional Array Variable	8-33
Creating a Multi-Dimensional Array	8-33
Assigning Values to a Multi-Dimensional Array	8-36
Retrieving Values from a Multi-Dimensional Array	8-36
Exercise 8-6: Storing Team Scores	8-39
Review Questions	8-40

9 Introduction to Object Oriented Programming and UML	9-1
Introduction	9-1
What is OO Programming?	9-2
Classes	9-3
Defining a Class	9-5
Exercise 9-1: Interpreting UML Class Notation	9-7
Encapsulation	9-9
Exercise 9-2: Using Encapsulation to Define a Class	9-11
Inheritance	9-12
Exercise 9-3: Inheritance Concept	9-14
Abstract Classes	9-15
Interfaces	9-16
Code for our Class	9-17
Using a Class	9-18

Creating Objects from a Class	9-21
Exercise 9-4: Creating Objects from Classes	9-23
More UML Diagram Notation	9-24
Representing a Single Class	9-24
Exercise 9-5: Draw a Class Diagram for a Single Class	9-26
Representing Class-Object Relationship	9-28
Exercise 9-6: Drawing a Class-Object Relationship Diagram	9-29
Representing Class Relationships	9-30
Exercise 9-7: Representing Class Associations	9-33
Exercise 9-8: Representing Cardinality of Relationships	9-34
Inheritance Relationships	9-34
Exercise 9-9: Inheritance Relationship Diagrams	9-35
Review Questions	9-36

10 Using Objects 10-1

Introduction	10-1
More About Objects	10-2
Why Are Objects Important?	10-2
Where do Objects Come From?	10-3
A World of Objects	10-5
Exercise 10-1: Objects	10-6
Declaring Object Variables	10-7
Creating Objects	10-8
Accessing Object Attributes	10-9
Access Modifiers	10-9
Exercise 10-2: Creating and Using Objects	10-10
Invoking Object Methods	10-11
Member Signatures	10-12
The this Reference	10-13
Using the Java API Documentation	10-14
Initializing Objects with Constructors	10-15
Person Class Members	10-16
Method and Constructor Overloading	10-17
Exercise 10-3: Using the Person Class	10-19
Objects with Multiple Aliases (Shared Objects)	10-20
Side Effects of Multiple Aliases	10-21
Garbage Collection	10-23
The null Reference	10-25
Dereferencing a null Reference	10-25
The Database Class	10-27
Overview of the CD Database System Exercise	10-28
Exercise 10-4: The CD Database System (Part 1)	10-28
Exercise 10-4: The CD Database System (Part 2)	10-29
Exercise 10-4: The CD Database System (Part 3)	10-30

Exercise 10-4: The CD Database System (Part 4) 10-31
Exercise 10-4: The CD Database System (Part 5) 10-32
Review Questions 10-33

11 Developing Classes 11-1

- Introduction 11-1
- Steps to Create a Class 11-2
 - Case Study: A Banking Application 11-2
- Step 1: Name the Class 11-3
 - A Minimal Class Definition 11-4
 - The main Method 11-6
- Exercise 11-1: A Minimally-Defined Class** 11-7
- Step 2: Determine and Declare the Attributes 11-8
- Exercise 11-2: Class Attributes** 11-11
- Step 3: Determine, Define, and Implement the Methods 11-12
 - Step 3a: Defining Methods - Determine Operations 11-12
 - Step 3b: Defining Methods - Define Signatures and Return Types 11-13
 - Naming Conventions for Java Methods 11-14
 - Case Study Example: Determine and Define the Methods 11-15
 - Step 3c: Implement the Methods 11-17
 - How Method Calls Work 11-18
 - Implementing the Deposit and Withdraw Methods 11-19
 - Local Variable Declarations 11-19
 - Methods Implemented in BankAccount Class 11-20
- Exercise 11-3: Implementing Methods** 11-23
- Initializing Objects: Constructors Revisited 11-24
 - Creating a No-Argument Constructor 11-25
 - Overloaded Constructors 11-26
- Exercise 11-4: Define a Constructor** 11-27
- Inheritance Revisited 11-29
 - The extends Keyword 11-30
 - Using Inheritance 11-31
 - The super Reference 11-33
 - Polymorphism 11-34
 - The Object Class 11-35
- Abstract Classes Revisited 11-36
 - The abstract Keyword 11-36
 - Using an Abstract Class 11-37
- Interfaces Revisited 11-40
 - Java's Comparable Interface 11-40
 - The implements Keyword 11-41
 - Using Interfaces 11-42
- Type Casting of Objects 11-44
 - Up-Casting 11-44
 - Down-Casting 11-44
- Access Modifiers: private, public and protected 11-45

The static Modifier	11-48
Exercise 11-5: Using a static Attribute	11-50
Exercise 11-6: A Banking Application	11-51
Review Questions	11-55
12 Introduction to GUI Programming in Java	12-1
Introduction	12-1
What is a GUI?	12-2
Swing and the AWT	12-4
The Abstract Windowing Toolkit	12-4
Swing	12-5
Java GUI Component Hierarchy	12-6
Exercise 12-1: Hello Swing – A Simple Swing Application	12-7
Steps in Creating A Simple Swing GUI	12-8
Step 1: Import Packages	12-9
Step 2: Create a Top-Level Container	12-9
Step 3: Declare General Containers and Basic Components	12-11
Step 4: Create and Add Components to the Container	12-12
Creating the JButton Component	12-13
Create the JLabel Component	12-14
Create the JPanel Component	12-15
ADD Components to the JPanel	12-16
Set JPanel To Be the Content Pane	12-17
Step 5: Create an Instance of the GUI Class	12-18
Step 6: Handle Events	12-19
Swing Components	12-20
Exercise 12-2: Run the SwingSet Demo	12-21
Top-Level Containers	12-22
JFrame	12-22
JDialog	12-23
Top-level Container's Panes	12-24
Packing and Viewing A Top-level Container	12-24
General Purpose Containers	12-26
JPanel	12-26
Exercise 12-3: Using JFrame and JPanel	12-28
JScrollPane	12-29
Layout Managers	12-31
The FlowLayout Manager	12-32
The BorderLayout Manager	12-33
The GridLayout Manager	12-34
Layout Management - the Mechanism	12-35
Specifying the Size of a Component	12-36
Popular Components	12-37
JButton	12-37
JLabel	12-38
JTextField	12-39

Exercise 12-4: Creating a Login Screen	12-41
JTextArea	12-43
JComboBox	12-45
JCheckBox	12-47
JRadioButton	12-48
JLists	12-49
Exercise 12-5: Creating a ‘Create Customer’ Screen	12-50
Pop-Up Menus	12-53
Exercise 12-6: Creating a Main Screen	12-54
Event Handling	12-55
Java’s Event Model	12-55
Steps for Handling Events	12-56
Naming Conventions	12-59
Event Handling	12-60
Event Listeners	12-61
Event Listeners Using Inner Classes	12-61
Event Listeners Using Anonymous Inner Classes	12-63
Handling Events In a Container Class	12-64
Adapter Classes	12-65
The ActionEvent Class	12-67
The WindowEvent Class	12-69
Closing a JFrame	12-70
Another approach for closing a JFrame	12-71
Exercise 12-7: Handling Login Events	12-72
Exercise 12-8: Handling Menu Item Events	12-73
Exercise 12-9: Handling Create Customer Events	12-74
Exercise 12-10: Handling Deposit/Withdrawal Events	12-75
Review Questions	12-77

13 Data Structures

Introduction	13-1
What are Data Structures?	13-2
Stacks	13-3
Stack Operations	13-4
Implementing a Stack Using an Array	13-4
Stack Example	13-7
Exercise 13-1: Using the StackArray Class	13-8
Queues	13-8
Queue Operations	13-9
Implementing a Queue Using an Array	13-10
Queue Example	13-14
Exercise 13-2: Using the QueueArray Class	13-15
Linked Lists	13-16
Linked List vs. Array	13-17
Linked List Operations	13-18
Implementation of a Linked List	13-20

Linked List Example	13-26
Exercise 13-3: Implementing a Stack using a Linked List	13-27
Exercise 13-4: Implementing a Queue using a Linked List	13-28
Recursion	13-29
Example Problem that Cannot be Solved Using Recursion	13-29
Example Problem that Can be Solved Using Recursion	13-29
Exercise 13-5: Find the Nth Number in a Fibonacci Sequence	13-32
Binary Tree	13-33
Binary Search Tree	13-34
Binary Search Tree Operations.	13-35
Inserting an Item	13-35
Locating an Item	13-37
Deleting an Item	13-37
Tree Traversal	13-41
Implementation of a Binary Search Tree	13-43
Binary Tree Example	13-51
Exercise 13-6: Using the BinaryTree Class.	13-54
Exercise 13-7: Inserting Data Into a Binary Search Tree	13-54
Exercise 13-8: Traversing a Binary Search Tree	13-55
Java Data Structures	13-55
List Implementations	13-56
The ArrayList Class	13-56
Methods of the ArrayList Class	13-57
Example Using the ArrayList Class	13-58
Exercise 13-9: Using the ArrayList Class	13-59
Wrapper Classes	13-59
Wrapper Class Example.	13-61
Exercise 13-10: Using the Wrapper Class	13-62
Vectors	13-62
Methods of the Vector class	13-63
Example Using the Vector Class	13-64
Exercise 13-11: Using the Vector Class	13-65
The LinkedList Class	13-65
Methods of the LinkedList Class	13-66
Example Using the LinkedList Class	13-67
Exercise 13-12: Using the LinkedList Class	13-68
Set Implementations	13-69
The TreeSet Class	13-69
Methods of the TreeSet Class	13-70
Example Using the TreeSet Class	13-71
Exercise 13-13: Using the TreeSet Class	13-72
Map Implementations	13-73
The HashMap Class	13-73
Exercise 13-14: Using the HashMap Class	13-76
The TreeMap Class	13-76
The Hashtable Class	13-76

Exercise 13-15: Using the Hashtable Class	13-79
Enumerating Values of a Data Structure	13-80
The Enumeration Interface	13-80
Enumeration Interface Example	13-81
Iteration Interface	13-82
Exercise 13-16: Using Vectors and Wrapper Classes	13-83
Review Questions	13-84
14 Exception Handling	14-1
Introduction	14-1
What is an Exception?	14-2
How Java Handles Exceptions	14-3
Java's Catch or Specify Requirement	14-4
Checked vs. Unchecked Exceptions	14-4
Exercise 14-1: Java Exceptions	14-5
Exception Handling with Try-Catch-Finally	14-5
The try Block	14-7
The catch Block	14-7
Catching Multiple Exceptions	14-9
The finally Block	14-10
Exercise 14-2: Catching Exceptions	14-11
Exercise 14-3: More on Catching Exceptions	14-12
Exception Handling with Throws Clause	14-13
Propagating Exceptions	14-14
Exceptions Example	14-15
Java Exception Hierarchy	14-17
The Throwable Class	14-17
The Exception Class	14-19
The Error Class	14-21
User-Defined Exceptions	14-22
Exercise 14-4: User-Defined Exceptions	14-23
Review Questions	14-24
15 Java I/O	15-1
Introduction	15-1
What is I/O?	15-2
Streams	15-2
Java I/O	15-3
Reading Strings from Standard Input	15-5
Exercise 15-1: Reading from STDIN	15-7
Writing to Standard Output	15-8
Text File I/O	15-9
Reading from a Text File	15-9
Writing to a Text File	15-10

Exercise 15-2: Reading and Writing Text Files	15-11
StringTokenizer	15-12
Exercise 15-3: Using StringTokenizer	15-14
Supplemental Exercises	15-15
Exercise 15-4: Reading from STDIN (optional)	15-15
Exercise 15-5: Reading and Writing Files (optional)	15-16
Review Questions	15-18
A First Bank of Java GUI Incremental Design	A-1
Introduction	A-1
Requirements	A-2
The User Interface	A-4
Login Screen	A-4
Main Screen	A-5
“Create Customer” Screen	A-6
“Create Account” Screen	A-7
“Customer Details” Screen	A-8
“Account Details” Screen	A-8
“Search” Screen	A-9
Classes of the GUI Application	A-11
MainApp Class	A-11
LoginPanel	A-11
MainPanel	A-12
CreateCustomerPanel	A-13
CustomerDetailsPanel	A-14
CreateAccountPanel	A-15
AccountDetailsPanel	A-16
SearchPanel	A-17
Customer	A-17
BankAccount, CheckingAccount and SavingsAccount	A-18
Listener Classes	A-18
Structure of the Java Files	A-19
Persistent Data Storage	A-19
B Java’s Reserved Keywords, Operators and Flowchart Symbols	B-1
Java’s Reserved Keywords	B-1
Operators and their Precedence	B-2
Flowchart Symbols	B-3
Index	I-1

Index

A

abs (double x) 11-49
abs(int a) 11-49
abstract class 9-15
Abstract Classes 11-36
abstract methods 9-15
Access Modifiers 11-45
access modifiers 10-9
 private 10-10, 11-46
 protected 10-10, 11-46
 public 10-10, 11-46
ActionEvent class 12-67
adapter classes 12-65
adding components to top-level container 12-12
Addition operator 3-46
algorithm 1-4
alias 10-21
applets 2-12
application 1-2
array 8-1
 create 8-5
 index position 8-9
ArrayIndexOutOfBoundsException 14-3
ArrayList 13-55
ArrayList Class 13-56
assembly language 2-5
assignment operator 3-7
attribute signatures 10-12
autoflush 15-8
AWT 12-4

B

BankAccount.java 10-3
basic components 12-20
boolean 3-37
BufferedReader class 15-6
bytecode 2-8
bytes streams 15-3

C

cardinality symbols 9-32
catch 14-5
catch block 14-7
changing array elements 8-8
character 3-27
character streams 15-3
checked exception 14-4
class 2-19, 9-3, 11-1
 attribute declaration 11-8
 implementing methods 11-17
 method definition 11-12
 naming conventions 11-3
 steps to create 11-2
 template for object 9-2
class code example 9-17
Classes 9-3
Class-Object Relationship 9-28
closing a JFrame 12-70
command-line parameters 8-30
comments 2-20
Comparable Interface 11-40
comparison operators 4-4
Comparisons 7-15
compiler 1-5
 purpose 2-7
component 12-6
components 12-6
 basic 12-20
 frequently used components 12-37
Concatenation 7-9
concrete class 9-15
console applications 2-13
constructor
 default 11-24
 no argument 11-25
 overloaded 11-24, 11-26
constructor method 10-15
Constructor Overloading 10-17

- containers 12-22
 - general 12-20
 - top-level 12-20
- content pane 12-24
- CostCalculator.java 6-17
- Counter-controlled loops 5-3
- counting-controlled 5-1
- creating a top level container 12-9
- Creating Objects 10-8
- D**
- data hiding 9-10
- Database class 10-27
- decrement operator (--) 3-63
- default constructor 11-24
- Defining a Class 9-5
- divide and conquer 6-2
- Division 3-52
- division operator 3-52
- do while loop 5-11
- DOS
 - cd 1-12
 - dir 1-12
- DOS prompt 1-11
- dot operator 10-9, 10-11
- double 3-22
- do-while loop 5-2
- down-casting 11-44
- dual if-statement 4-9
- E**
- ELSE 1-7
- embedded applications 2-14
- encapsulation 9-3, 9-9, 9-10
- Enumeration 13-80
- Enumeration Interface 13-80
- Error class
 - sample subclasses 14-21
- errors
 - examples of 14-2
- escape sequences 7-3
- event 12-55
- event classes 12-60
 - and their Listeners/Adapters 12-60
- event handler 12-19
- event handling 12-55
 - in a container class 12-64
- event loop 12-55
- event object 12-56
- events
 - anonymous inner class 12-63
 - inner class 12-61
- exception 14-2
 - catch block 14-7
 - catching multiple exceptions 14-9
 - class hierarchy 14-17
 - coding example 14-15
 - examples of 14-2
 - Exception class 14-19
 - handler 14-5
 - how Java handles them 14-3
 - Throwable class 14-17
 - try block 14-7
 - user-defined 14-22
 - vs. error 14-2
 - Exception class 14-19
 - sample subclasses 14-19
 - exception handler 14-3
 - exception handlers
 - generalized 14-9
 - specialized 14-9
- Exercise
 - Assigning a Value to a Variable of Type boolean 3-40
 - Assigning Values to Variables of Type character 3-30
 - Assigning Values to Variables of Type double 3-25
 - Assigning Values to Variables of Type integer 3-14
 - Calling Methods I 6-13
 - Calling Methods II 6-20
 - Change the Order of Operations 3-66
 - Changing the String Value 3-37
 - Compiling and Running Java Programs 1-16
 - Create a Simple Java Application 1-17
 - Creating an Array 8-6
 - Creating an Array of Strings 8-21
 - Creating and Using Objects 10-10
 - Declare a Variable of Type boolean 3-39
 - Declare a Variable of Type character (char) 3-29
 - Declare a Variable of Type double 3-24
 - Declare a Variable of Type integer 3-13
 - Distinguishing Different Types of Methods 6-15
 - Fibonacci Numbers 8-31
 - Find Valid Variable Names 3-9
 - Find Values of Type character 3-31
 - Find Your Destination 8-29
 - Identifying Array Elements 8-10
 - Identifying Parts of a Method 6-7, 6-12
 - Keeping Track of Variable Values 5-7
 - Map Values to Variables 3-40

- Reading & Writing Files (optional) 15-16
- Reading and Writing Files 15-11, 15-14
- Reading from STDIN 15-7, 15-15
- The CD Database System 10-28
- The main Method 2-20
- Understanding Multiple If-Statements 4-23, 6-21
- Using a Switch Statement 4-32
- Using a While Loop 5-8
- Using an If-Statement 4-17, 4-19, 4-24, 5-14
- Using Comments 2-22
- Using Comparison Operators 4-5, 4-8
- Using Learning IO 3-21
- Using Operators 3-56, 3-59, 3-60, 3-62, 3-64
- Using String Variables 3-35
- Using the Hashtable Class 13-79
- Using the Person Class 10-19
- Using the Vector Class 13-65, 13-68
- Using Vectors and Wrapper Classes 13-83
- Writing a Program 3-36
- extends keyword 11-30
- extension 1-14
- F**
- Figure 2-6
 - High to low level code 2-6
- Figure 3-2
 - Adding Numbers Program 3-3
- File I/O 15-9
- FileNotFoundException 14-3, 14-5
- FileReader class 15-9
- FileWriter class 15-10
- finally 14-5
- flow control 4-2
- for loop 5-2
- frame 12-10
- G**
- garbage collection 10-23
- general containers 12-20
- GridLayout 12-34
- GUI Applications 2-14
- H**
- handle events 12-19
- HashMap 13-73
- HashMap class 13-73
- HashTable 13-73
- Hashtable class 13-76
- I**
- IF 1-7
- if-statement without else 4-14
- immutable 7-5
- implements Keyword 11-41
- import 3-20
- increment operator (++) 2-7, 3-63
- index 8-1
- IndexOutOfBoundsException exception 8-15
- inheritance 9-12, 11-31, 11-42
- inheritance hierarchy Java IO 15-3
- inner class
 - anonymous 12-63
- InputStream class 15-3
- int 3-12
- integer 3-12
- Interfaces 11-40
- interfaces 9-16
- interpreter 1-5
- iteration 5-2
- iterator 13-82
- Iterator interface 13-82
- J**
- jar files 3-18
- java 2-10
- Java API documentation 10-14
- Java Applications 2-12
- Java GUI components 12-6
- Java I/O 15-3
- Java IO class 3-19
- Java program 2-19
- Java programming language 2-2
- java.lang.Comparable 11-40
- java.lang.Math 11-48
- java.lang.String 7-6
- java.lang.String class 7-1
- java.util.Vector 13-63
- javac 2-9
- JCheckBox class 12-47
- JComboBox class 12-45
- JDialog 12-23
- JDK 2-2
- JFrame 12-22
 - closing 12-71
- JLabel class 12-38
- JList class 12-49
- JPanel 12-26
- JRadioButton class 12-48
- JScrollPane class 12-29
- JTextField class 12-39
- JVM 2-8
- L**
- layout manager
 - how one works 12-35
- LengthConversion.java 6-19
- LinkedList 13-56, 13-65

LinkedList Class 13-65
List implementation classes 13-56
listener 12-61
 steps for setting up to listen 12-56
local variable 11-20
loop 5-2

M

machine instructions 1-5, 2-4
main method 2-19
Map interface 13-73
matrix 8-4
max(double x, double y) 11-49
Member Signatures 10-12
memory of the computer 3-25
menus 12-53
method 6-5
 defining 6-6
Method Definition 11-13
method overloading 10-17
methods 2-18
 how methods calls work 11-18
 naming conventions 11-14
 signature of 11-16
min(double x, double y) 11-49
minimal Java class 11-4
modularity 2-18
modularization 6-2
module 2-18
modulo operator (%) 3-54
Multiple Aliases 10-20
multiple if-statements 4-20
multiple inheritance 9-16
multiplication operator 3-50

N

naming conventions 12-59
new operator 8-7
no argument constructor 11-25
non-printable characters 7-3
null reference 10-25

O

object 9-3
Object Class 11-35
object variable
 declare 10-7
objects 2-18
OO Programming 9-2
operands 3-4
operators 3-4, 3-45
OutputStream class 15-3
overloaded 11-24
overloaded operator 7-9

P

pack method 12-24
Panel class 12-26
parameter 6-8
 passed by reference 6-8
 passed by value 6-8
Passing Parameters 6-8
Person class 10-16
polymorphism 11-34
pow(double x, double y) 11-49
precedence of operators 3-65
primitive types 10-2
private 11-45
private access modifier 11-46
problem 1-3
programming language 1-4
protected 11-45
protected access modifier 11-46
pseudocode 1-4, 1-5, 3-2
 choosing between alternative actions 1-7
 output 1-7
 performing arithmetic 1-7
 repeating actions 1-8
 storing a value 1-7
public 11-45
public access modifier 11-46
public methods 9-9

Q

Queues 13-8

R

random() 11-49
reading from a text file 15-9
red-black tree 13-69
reference 10-7
Repetition 4-2
repetition statement 5-2
reserved characters 7-3
return 6-10
return type 6-10

S

Selection 4-2
sentinel-controlled 5-1
Sentinel-controlled loops 5-3
servlets 2-12, 2-16
Set Implementations 13-69
Set interface 13-69
setDefaultCloseOperation() method 12-71
signature
 attribute 10-12
 method 10-12
signatures 11-16

sin(double a) 11-49
sin(double a) 11-49
software 1-2
sqrt(double x) 11-49
Stack 13-3
standard input 15-5
standard out 15-8
statement 3-4
statements 2-6
static 11-48
STDIN 15-5
Streams 15-2
String 3-32
String Buffer Methods 7-19
String Methods 7-6
StringBuffer 7-1
StringBuffer class 7-17
Strings 7-2
StringTokenizer 15-6, 15-12
subclass 9-12
Substrings 7-12
Subtraction operator 3-48
super reference 11-33
superclass 9-12
Swing 12-5
Swing components 12-20
syntax 3-6
System class 15-5

T

this reference 10-13
thread 13-62
Throwable class 14-17
throwing the exception 14-3
throws clause 14-13
throws keyword 14-13
top-level containers 12-20, 12-22
TreeMap 13-73
TreeMap class 13-76
TreeSet Class 13-69, 13-70
try 14-5
try block 14-7
Type Casting 11-44

U

UML notation
 cardinality symbols 9-32
 class 9-5
 class diagram 9-24
 class relationships 9-30
 class relationships with cardinality 9-30
 class-object relationship 9-28
unchecked exceptions 14-4

up-casting 11-44
user-defined exception 14-22

V

variable
 declaration 3-6
variable name 3-8
Vector 13-56
Vector class 13-62
Vectors 13-62

W

while loop 5-2, 5-3
WindowEvent class 12-69, 12-70
windows applications 2-14
wrapper classes 13-59
writing to a text file 15-10
writing to standard output 15-8