



## LOCATION OF VIDEONOTES IN THE TEXT

<b>Chapter 1</b>	Compiling and Running a Java Program, p. 14 Using an IDE, p. 14 Your First Java Program, p. 28
<b>Chapter 2</b>	Displaying Console Output, p. 37 Declaring Variables, p. 44 Simple Math Expressions, p. 60 The Miles-per-Gallon Problem, p. 115
<b>Chapter 3</b>	Writing Classes and Creating Objects, p. 123 Initializing an Object with a Constructor, p. 145 The Personal Information Class Problem, p. 178
<b>Chapter 4</b>	The <code>if</code> Statement, p. 181 The <code>if-else</code> Statement, p. 190 The <code>if-else-if</code> Statement, p. 205 The Time Calculator Problem, p. 256
<b>Chapter 5</b>	The <code>while</code> Loop, p. 263 The Pennies for Pay Problem, p. 327
<b>Chapter 6</b>	Returning Objects from Methods, p. 355 Aggregation, p. 367 The <code>InventoryItem</code> Class Copy Constructor Problem, p. 408
<b>Chapter 7</b>	Accessing Array Elements in a Loop, p. 418 Passing an Array to a Method, p. 436 The Charge Account Validation Problem, p. 504
<b>Chapter 8</b>	The Sentence Capitalizer Problem, p. 563
<b>Chapter 9</b>	Inheritance, p. 569 Polymorphism, p. 610 The Employee and Production-worker Classes problem, p. 639
<b>Chapter 10</b>	Handling Exceptions, p. 645 The Exception Project Problem, p. 699
<b>Chapter 11</b>	Creating a Simple GUI Application, p. 715 Handling Events, p. 724 The Monthly Sales Tax Problem, p. 798
<b>Chapter 12</b>	The <code>JList</code> Component, p. 804 The <code>JcomboBox</code> Component, p. 821 The Image Viewer Problem, p. 865

*(continued on the next page)*

## LOCATION OF VIDEONOTES IN THE TEXT *(continued)*



<b>Chapter 13</b>	Creating an Applet, p. 881 The House Applet Problem, p. 940
<b>Chapter 14</b>	Reducing a Problem with Recursion, p. 946 The Recursive Power Problem, p. 969
<b>Chapter 15</b>	Displaying Query Results in a JTable, p. 1008 The Coffee Inserter Problem, p. 1047

STARTING OUT WITH

**Java**<sup>™</sup>

Early Objects

**FOURTH EDITION**



STARTING OUT WITH

**Java**<sup>™</sup>

Early Objects

**FOURTH EDITION**

**Tony Gaddis**

Haywood Community College

**Addison-Wesley**

Boston Columbus Indianapolis New York San Francisco  
Upper Saddle River Amsterdam Cape Town Dubai London Madrid Milan  
Munich Paris Montreal Toronto Delhi Mexico City Sao Paulo Sydney  
Hong Kong Seoul Singapore Taipei Tokyo

Vice President/Editorial Director: *Marcia Horton*  
Editor-in-Chief: *Michael Hirsch*  
Editorial Assistant: *Stephanie Sellinger*  
Vice President of Marketing: *Patrice Jones*  
Marketing Manager: *Yezan Alayan*  
Marketing Coordinator: *Kathryn Ferranti*  
Vice President, Production: *Vince O'Brien*  
Managing Editor: *Jeff Holcomb*  
Production Project Manager: *Kayla Smith-Tarbox*  
Senior Operations Supervisor: *Lisa McDowell*  
Art Director: *Linda Knowles*  
Cover Designer: *Joyce Wells-Cosentino*  
Cover Image: *© ilker yuksel/iStockphoto*  
Media Editor: *Dan Sandin*  
Project Management: *Dennis Free, Aptara®, Inc.*  
Composition and Illustration: *Aptara®, Inc.*  
Printer/Binder: *Edwards Brothers*  
Cover Printer: *Coral Graphics*

Copyright © 2011, 2008, 2005 Pearson Education, Inc., publishing as Addison-Wesley. All rights reserved. Manufactured in the United States of America. This publication is protected by Copyright, and permission should be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or likewise. To obtain permission(s) to use material from this work, please submit a written request to Pearson Education, Inc., Permissions Department, 501 Boylston Street, Suite 900, Boston, Massachusetts 02116.

Many of the designations by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and the publisher was aware of a trademark claim, the designations have been printed in initial caps or all caps.

Library of Congress Cataloging-in-Publication  
Data available upon request.

10 9 8 7 6 5 4 3 2 1—EB—14 13 12 11 10

**Addison-Wesley**  
is an imprint of



[www.pearsonhighered.com](http://www.pearsonhighered.com)

ISBN 13: 978-0-132-16476-4  
ISBN 10: 0-132-16476-0

# Contents in Brief

	<b>Preface</b>	<b>xv</b>	
Chapter 1	<b>Introduction to Computers and Java</b>		1
Chapter 2	<b>Java Fundamentals</b>		31
Chapter 3	<b>A First Look at Classes and Objects</b>		119
Chapter 4	<b>Decision Structures</b>		181
Chapter 5	<b>Loops and Files</b>		263
Chapter 6	<b>A Second Look at Classes and Objects</b>		333
Chapter 7	<b>Arrays and the ArrayList Class</b>		413
Chapter 8	<b>Text Processing and Wrapper Classes</b>		511
Chapter 9	<b>Inheritance</b>		569
Chapter 10	<b>Exceptions and Advanced File I/O</b>		645
Chapter 11	<b>GUI Applications—Part 1</b>		701
Chapter 12	<b>GUI Applications—Part 2</b>		803
Chapter 13	<b>Applets and More</b>		869
Chapter 14	<b>Recursion</b>		943
Chapter 15	<b>Databases</b>		971
Appendix A	<b>Getting Started with Alice</b>		1049
	<b>Index</b>		1077
	<b>Appendixes B–M</b>		<a href="#">Available on the book's online resource page</a>
	<b>Case Studies 1–5</b>		<a href="#">Available on the book's online resource page</a>





# Contents

Preface xv

## Chapter 1 Introduction to Computers and Java 1

1.1	Introduction . . . . .	1
1.2	Why Program? . . . . .	1
1.3	Computer Systems: Hardware and Software . . . . .	2
1.4	Programming Languages . . . . .	6
1.5	What Is a Program Made of? . . . . .	9
1.6	The Programming Process . . . . .	16
1.7	Object-Oriented Programming . . . . .	19
	<i>Review Questions and Exercises</i>	24
	<i>Programming Challenge</i>	28

## Chapter 2 Java Fundamentals 31

2.1	The Parts of a Java Program . . . . .	31
2.2	The print and println Methods, and the Java API . . . . .	37
2.3	Variables and Literals . . . . .	43
2.4	Primitive Data Types . . . . .	49
2.5	Arithmetic Operators . . . . .	60
2.6	Combined Assignment Operators . . . . .	69
2.7	Conversion between Primitive Data Types . . . . .	70
2.8	Creating Named Constants with final . . . . .	74
2.9	The String Class . . . . .	76
2.10	Scope . . . . .	81
2.11	Comments . . . . .	83
2.12	Programming Style . . . . .	87
2.13	Reading Keyboard Input . . . . .	89
2.14	Dialog Boxes . . . . .	97
2.15	The printf Method . . . . .	104
2.16	Common Errors to Avoid . . . . .	107
	<i>Review Questions and Exercises</i>	109
	<i>Programming Challenges</i>	113

<b>Chapter 3</b>	<b>A First Look at Classes and Objects</b>	<b>119</b>
3.1	Classes	119
3.2	More about Passing Arguments	139
3.3	Instance Fields and Methods	142
3.4	Constructors	145
3.5	A BankAccount Class	151
3.6	Classes, Variables, and Scope	162
3.7	Packages and import Statements	163
3.8	Focus on Object-Oriented Design: Finding the Classes and Their Responsibilities	164
3.9	Common Errors to Avoid	173
	<i>Review Questions and Exercises</i>	173
	<i>Programming Challenges</i>	177
<b>Chapter 4</b>	<b>Decision Structures</b>	<b>181</b>
4.1	The if Statement	181
4.2	The if-else Statement	190
4.3	The Payroll Class	193
4.4	Nested if Statements	197
4.5	The if-else-if Statement	205
4.6	Logical Operators	210
4.7	Comparing String Objects	218
4.8	More about Variable Declaration and Scope	223
4.9	The Conditional Operator (Optional)	225
4.10	The switch Statement	226
4.11	Formatting Numbers with the DecimalFormat Class	236
4.12	Focus on Problem Solving: The SalesCommission Class	240
4.13	The Random Class	247
4.14	Common Errors to Avoid	250
	<i>Review Questions and Exercises</i>	251
	<i>Programming Challenges</i>	256
<b>Chapter 5</b>	<b>Loops and Files</b>	<b>263</b>
5.1	The Increment and Decrement Operators	263
5.2	The while Loop	267
5.3	Using the while Loop for Input Validation	274
5.4	The do-while Loop	277
5.5	The for Loop	280
5.6	Running Totals and Sentinel Values	291
5.7	Nested Loops	296
5.8	The break and continue Statements	299
5.9	Deciding Which Loop to Use	299

5.10	Introduction to File Input and Output . . . . .	300
5.11	Common Errors to Avoid . . . . .	320
	<i>Review Questions and Exercises</i>	321
	<i>Programming Challenges</i>	326

## Chapter 6 **A Second Look at Classes and Objects** 333

6.1	Static Class Members . . . . .	333
6.2	Overloaded Methods . . . . .	340
6.3	Overloaded Constructors . . . . .	345
6.4	Passing Objects as Arguments to Methods . . . . .	352
6.5	Returning Objects from Methods . . . . .	355
6.6	The toString Method . . . . .	358
6.7	Writing an equals Method . . . . .	362
6.8	Methods That Copy Objects . . . . .	364
6.9	Aggregation . . . . .	367
6.10	The this Reference Variable . . . . .	380
6.11	Inner Classes . . . . .	383
6.12	Enumerated Types . . . . .	386
6.13	Garbage Collection . . . . .	395
6.14	Focus on Object-Oriented Design: Class Collaboration . . . . .	397
6.15	Common Errors to Avoid . . . . .	401
	<i>Review Questions and Exercises</i>	402
	<i>Programming Challenges</i>	407

## Chapter 7 **Arrays and the ArrayList Class** 413

7.1	Introduction to Arrays . . . . .	413
7.2	Processing Array Contents . . . . .	424
7.3	Passing Arrays as Arguments to Methods . . . . .	436
7.4	Some Useful Array Algorithms and Operations . . . . .	440
7.5	Returning Arrays from Methods . . . . .	452
7.6	String Arrays . . . . .	454
7.7	Arrays of Objects . . . . .	458
7.8	The Sequential Search Algorithm . . . . .	462
7.9	The Selection Sort and the Binary Search Algorithms . . . . .	465
7.10	Two-Dimensional Arrays . . . . .	473
7.11	Arrays with Three or More Dimensions . . . . .	485
7.12	Command-Line Arguments and Variable-Length Argument Lists . . . . .	486
7.13	The ArrayList Class . . . . .	490
7.14	Common Errors to Avoid . . . . .	498
	<i>Review Questions and Exercises</i>	499
	<i>Programming Challenges</i>	503

**Chapter 8 Text Processing and Wrapper Classes 511**

8.1 Introduction to Wrapper Classes . . . . . 511  
 8.2 Character Testing and Conversion with the Character Class. . . . . 512  
 8.3 More about String Objects . . . . . 519  
 8.4 The StringBuilder Class . . . . . 533  
 8.5 Tokenizing Strings . . . . . 542  
 8.6 Wrapper Classes for the Numeric Data Types. . . . . 551  
 8.7 Focus on Problem Solving: The TestScoreReader Class . . . . . 555  
 8.8 Common Errors to Avoid . . . . . 559  
*Review Questions and Exercises* 559  
*Programming Challenges* 563

**Chapter 9 Inheritance 569**

9.1 What Is Inheritance? . . . . . 569  
 9.2 Calling the Superclass Constructor. . . . . 580  
 9.3 Overriding Superclass Methods . . . . . 588  
 9.4 Protected Members . . . . . 596  
 9.5 Classes That Inherit from Subclasses. . . . . 603  
 9.6 The Object Class. . . . . 608  
 9.7 Polymorphism . . . . . 610  
 9.8 Abstract Classes and Abstract Methods . . . . . 615  
 9.9 Interfaces . . . . . 621  
 9.10 Common Errors to Avoid . . . . . 632  
*Review Questions and Exercises* 633  
*Programming Challenges* 639

**Chapter 10 Exceptions and Advanced File I/O 645**

10.1 Handling Exceptions . . . . . 645  
 10.2 Throwing Exceptions . . . . . 666  
 10.3 Advanced Topics: Binary Files,  
 Random Access Files, and Object Serialization . . . . . 674  
 10.4 Common Errors to Avoid . . . . . 690  
*Review Questions and Exercises* 691  
*Programming Challenges* 697

**Chapter 11 GUI Applications—Part 1 701**

11.1 Introduction. . . . . 701  
 11.2 Dialog Boxes . . . . . 704  
 11.3 Creating Windows . . . . . 715  
 11.4 Equipping GUI Classes with a main Method. . . . . 739  
 11.5 Layout Managers . . . . . 741  
 11.6 Radio Buttons and Check Boxes. . . . . 758  
 11.7 Borders . . . . . 771

11.8 Focus on Problem Solving: Extending the JPanel Class . . . . . 773  
 11.9 Splash Screens . . . . . 787  
 11.10 Using Console Output to Debug a GUI Application . . . . . 788  
 11.11 Common Errors to Avoid . . . . . 792  
*Review Questions and Exercises* 793  
*Programming Challenges* 798

**Chapter 12 GUI Applications—Part 2 803**

12.1 Read-Only Text Fields . . . . . 803  
 12.2 Lists . . . . . 804  
 12.3 Combo Boxes . . . . . 821  
 12.4 Displaying Images in Labels and Buttons . . . . . 827  
 12.5 Mnemonics and Tool Tips . . . . . 833  
 12.6 File Choosers and Color Choosers . . . . . 835  
 12.7 Menus . . . . . 839  
 12.8 More about Text Components: Text Areas and Fonts . . . . . 848  
 12.9 Sliders . . . . . 853  
 12.10 Look and Feel . . . . . 857  
 12.11 Common Errors to Avoid . . . . . 859  
*Review Questions and Exercises* 860  
*Programming Challenges* 865

**Chapter 13 Applets and More 869**

13.1 Introduction to Applets . . . . . 869  
 13.2 A Brief Introduction to HTML . . . . . 871  
 13.3 Creating Applets with Swing . . . . . 881  
 13.4 Using AWT for Portability . . . . . 889  
 13.5 Drawing Shapes . . . . . 894  
 13.6 Handling Mouse Events . . . . . 914  
 13.7 Timer Objects . . . . . 925  
 13.8 Playing Audio . . . . . 929  
 13.9 Common Errors to Avoid . . . . . 934  
*Review Questions and Exercises* 934  
*Programming Challenges* 940

**Chapter 14 Recursion 943**

14.1 Introduction to Recursion . . . . . 943  
 14.2 Solving Problems with Recursion . . . . . 946  
 14.3 Examples of Recursive Methods . . . . . 950  
 14.4 A Recursive Binary Search Method . . . . . 957  
 14.5 The Towers of Hanoi . . . . . 960  
 14.6 Common Errors to Avoid . . . . . 964  
*Review Questions and Exercises* 965  
*Programming Challenges* 968

## Chapter 15 **Databases** 971

15.1	Introduction to Database Management Systems . . . . .	971
15.2	Tables, Rows, and Columns . . . . .	974
15.3	The SQL SELECT Statement . . . . .	977
15.4	Introduction to JDBC . . . . .	986
15.5	Inserting Rows . . . . .	994
15.6	Updating and Deleting Existing Rows . . . . .	996
15.7	Creating and Deleting Tables . . . . .	998
15.8	Creating a New Database with Java DB . . . . .	1001
15.9	Scrollable Result Sets . . . . .	1003
15.10	Result Set Meta Data . . . . .	1005
15.11	Displaying Query Results in a JTable . . . . .	1008
15.12	Relational Data . . . . .	1019
15.13	Advanced Topics . . . . .	1040
15.14	Common Errors to Avoid . . . . .	1042
	<i>Review Questions and Exercises</i> . . . . .	1042
	<i>Programming Challenges</i> . . . . .	1047

## Appendix A **Getting Started with Alice** 1049

### **Index** 1077

**Available on the book's online resource page at  
[www.pearsonhighered.com/gaddis](http://www.pearsonhighered.com/gaddis):**

Appendix B	The ASCII/Unicode Characters
Appendix C	Operator Precedence and Associativity
Appendix D	Java Key Words
Appendix E	Installing the JDK and JDK Documentation
Appendix F	Using the javadoc Utility
Appendix G	More about the Math Class
Appendix H	Packages
Appendix I	Working with Records and Random-Access Files
Appendix J	Installing Java DB
Appendix K	The QuickSort Algorithm
Appendix L	Answers to Checkpoints Questions
Appendix M	Answers to Odd-Numbered Review Questions
Case Study 1	The Amortization Class
Case Study 2	The PinTester Class
Case Study 3	Parallel Arrays
Case Study 4	The SerialNumber Class
Case Study 5	A Simple Text Editor Application

# Preface

Welcome to *Starting Out with Java: Early Objects*, Fourth Edition. This book is intended for a one-semester or a two-quarter CS1 course. Although it is written for students with no prior programming background, even experienced students will benefit from its depth of detail.

## Early Objects, Late Graphics

The approach taken by this text can be described as “early objects, late graphics.” The student is introduced to object-oriented programming (OOP) early in the book. The fundamentals of control structures, classes, and the OOP paradigm are thoroughly covered before moving on to graphics and more powerful applications of the Java language.

As with all the books in the *Starting Out With* series, the hallmark of this text is its clear, friendly, and easy-to-understand writing. In addition, it is rich in example programs that are concise and practical.

## New to this edition:

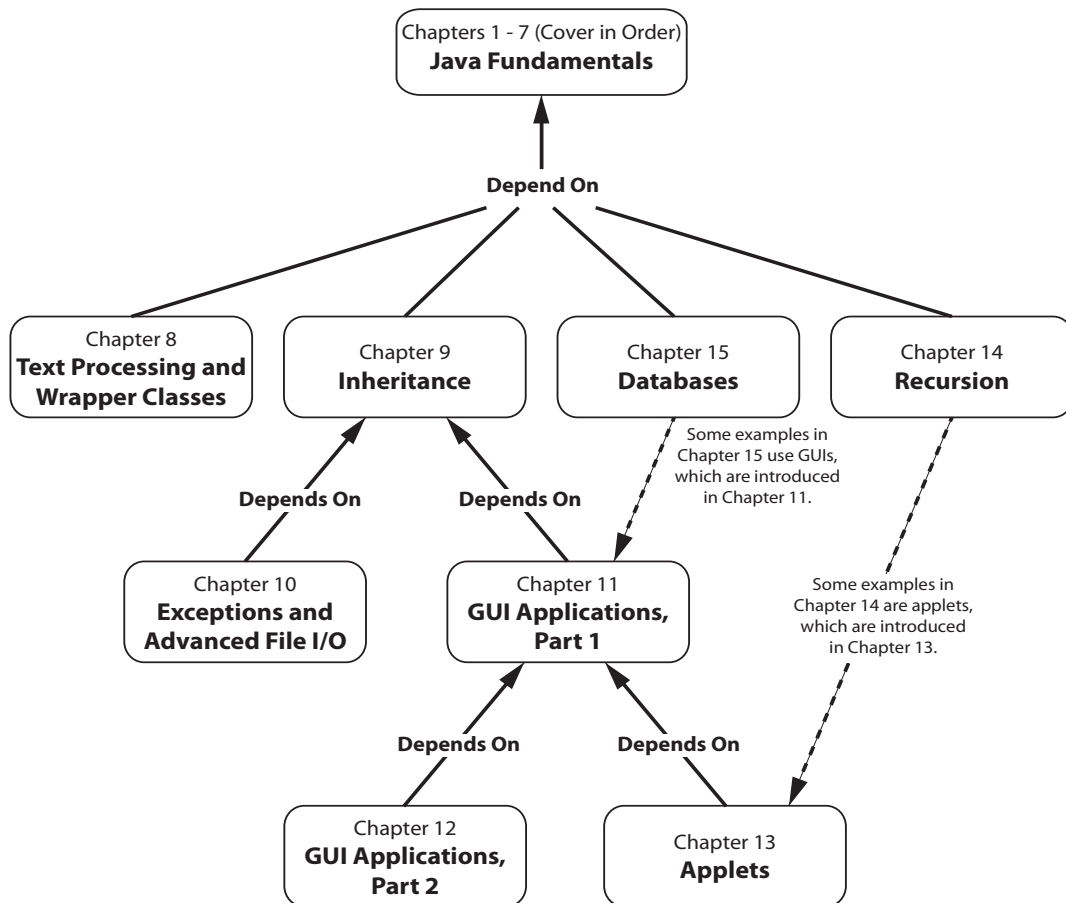
- A new chapter on databases: New to this edition is Chapter 15, “Databases.” This chapter introduces the student to databases and SQL using Java DB.
- New Java 7 topics added: This edition discusses some of the new language features added in Java 7. Specifically, the ability to switch on a string expression is discussed in Chapter 4, and type inference is discussed in Chapter 7.
- Improved organization in Chapter 4: The section covering the `if/else if` statement has been simplified and now appears immediately following the section on nested `if` statements. These sections have been rewritten to highlight the similarities between an `if/else if` statement and a nested `if` statement.
- A new section on `System.out.printf`: This new section has been added to Chapter 2.
- New *In the Spotlight* sections: Many of the chapters have new sections titled *In the Spotlight*. Each of these provides a programming problem and a detailed, step-by-step analysis showing the student how to solve it.
- Online VideoNotes: An extensive series of online VideoNotes have been developed to accompany this text. Throughout the book, VideoNote icons alert the student to videos covering specific topics. Additionally, one Programming Challenge at the end of each chapter now has an accompanying VideoNote explaining how to develop the problem’s solution. The videos are available at [www.pearsonhighered.com/gaddis](http://www.pearsonhighered.com/gaddis).

- **Additional programming problems:** Additional Programming Challenge problems have been added to most chapters. Several of these are simple games that will challenge and motivate students.

## Organization of the Text

The text teaches Java step-by-step. Each chapter covers a major set of topics and builds knowledge as students progress through the book. Although the chapters can be easily taught in their existing sequence, there is some flexibility. Figure P-1 shows chapter dependencies. Each box represents a chapter or a group of chapters. A solid-line arrow points from one chapter to the chapter that must be covered previously. A dotted-line arrow indicates that only a section or minor portion of the chapter depends on another chapter.

**Figure P-1** Chapter Dependencies



## Brief Overview of Each Chapter

**Chapter 1: Introduction to Computers and Java.** This chapter provides an introduction to the field of computer science, and covers the fundamentals of hardware, software, and programming languages. The elements of a program, such as key words,



variables, operators, and punctuation are discussed through the examination of a simple program. An overview of entering source code, compiling it, and executing it is presented. A brief history of Java is also given. The chapter concludes with a primer on OOP.

**Chapter 2: Java Fundamentals.** This chapter gets the student started in Java by introducing data types, identifiers, variable declarations, constants, comments, program output, and arithmetic operations. The conventions of programming style are also introduced. The student learns to read console input with the `Scanner` class, or as an option, through dialog boxes with `JOptionPane`.

**Chapter 3: A First Look at Classes and Objects.** This chapter introduces the student to classes. Once the student learns about fields and methods, UML diagrams are introduced as a design tool. The student learns to write simple `void` methods, as well as simple methods that return a value. Arguments and parameters are also discussed. Finally, the student learns how to write constructors, and the concept of the default constructor is discussed. A `BankAccount` class is presented as a case study, and a section on object-oriented design is included. This section leads the students through the process of identifying classes and their responsibilities within a problem domain. There is also a section that briefly explains packages and the `import` statement.

**Chapter 4: Decision Structures.** Here the student explores relational operators and relational expressions and is shown how to control the flow of a program with the `if`, `if/else`, and `if/else if` statements. The conditional operator and the `switch` statement are also covered. This chapter also discusses how to compare `String` objects with the `equals`, `compareTo`, `equalsIgnoreCase`, and `compareToIgnoreCase` methods. Formatting numeric output with the `DecimalFormat` class is covered. An object-oriented case study shows how lengthy algorithms can be decomposed into several methods.

**Chapter 5: Loops and Files.** This chapter covers Java's repetition control structures. The `while` loop, `do-while` loop, and `for` loop are taught, along with common uses for these devices. Counters, accumulators, running totals, sentinels, and other application-related topics are discussed. Simple file operations for reading and writing text files are also covered.

**Chapter 6: A Second Look at Classes and Objects.** This chapter shows students how to write classes with added capabilities. Static methods and fields, interaction between objects, passing objects as arguments, and returning objects from methods are discussed. Aggregation and the "has a" relationship is covered, as well as enumerated types. A section on object-oriented design shows how to use CRC (class, responsibilities, and collaborations) cards to determine the collaborations among classes.

**Chapter 7: Arrays and the ArrayList Class.** In this chapter students learn to create and work with single and multidimensional arrays. Numerous array-processing techniques are demonstrated, such as summing the elements in an array, finding the highest and lowest values, and sequentially searching an array are also discussed. Other topics, including ragged arrays and variable-length arguments (`varargs`), are also discussed. The `ArrayList` class is introduced and Java's generic types are briefly discussed and demonstrated.

**Chapter 8: Text Processing and Wrapper Classes.** This chapter discusses the numeric and character wrapper classes. Methods for converting numbers to strings, testing the case of characters, and converting the case of characters are covered. Autoboxing and unboxing are also discussed. More `String` class methods are covered, including using the `split` method to tokenize strings. The chapter also covers the `StringBuilder` and `StringTokenizer` classes.

**Chapter 9: Inheritance.** The study of classes continues in this chapter with the subjects of inheritance and polymorphism. The topics covered include superclass and subclass constructors, method overriding, polymorphism and dynamic binding, protected and package access, class hierarchies, abstract classes and methods, and interfaces.

**Chapter 10: Exceptions and Advanced File I/O.** In this chapter the student learns to develop enhanced error trapping techniques using exceptions. Handling an exception is covered, as well as developing and throwing custom exceptions. This chapter also discusses advanced techniques for working with sequential access, random access, text, and binary files.

**Chapter 11: GUI Applications, Part 1.** This chapter presents the basics of developing graphical user interface (GUI) applications with Swing. Fundamental Swing components and the basic concepts of event-driven programming are covered.

**Chapter 12: GUI Applications, Part 2.** This chapter continues the study of GUI application development. More advanced components, as well as menu systems and look-and-feel, are covered.

**Chapter 13: Applets and More.** Here the student applies his or her knowledge of GUI development to the creation of applets. In addition to using Swing applet classes, Abstract Windowing Toolkit classes are also discussed for portability. Drawing simple graphical shapes is also discussed.

**Chapter 14: Recursion.** This chapter presents recursion as a problem-solving technique. Numerous examples of recursion are demonstrated.

**Chapter 15: Databases.** This chapter introduces the student to database programming. The basic concepts of database management systems and SQL are first presented. Then the student learns to use JDBC to write database applications in Java. Relational data is covered, and numerous example programs are presented throughout the chapter.

**Appendix A.** Getting Started with Alice

## Features of the Text

**Concept Statements** Each major section of the text starts with a concept statement. This statement summarizes the ideas of the section.

**Example Programs** The text has an abundant number of complete example programs, each designed to highlight the topic currently being studied. In most cases, these are practical, real-world examples. Source code for these programs is provided so that students can run the programs themselves.

**Program Output** After each example program there is a sample of its screen output. This immediately shows the student how the program should function.



### Checkpoints

Checkpoints are questions placed throughout each chapter as a self-test study aid. Answers for all Checkpoint questions are found in Appendix L (available for download) so students can check how well they have learned a new topic. To download Appendix L, go to the Gaddis resource page at [www.pearsonhighered.com/gaddis](http://www.pearsonhighered.com/gaddis).



**NOTE:** Notes appear at appropriate places throughout the text. They are short explanations of interesting or often misunderstood points relevant to the topic at hand.



**WARNING!** Warnings are notes that caution the student about certain Java features, programming techniques, or practices that can lead to malfunctioning programs or lost data.



**VideoNotes.** A series of online videos, developed specifically for this book, are available for viewing at [www.pearsonhighered.com/gaddis](http://www.pearsonhighered.com/gaddis). Icons appear throughout the text alerting the student to videos about specific topics.

**Case Studies** Case studies that simulate real-world applications appear in many chapters throughout the text, with complete code provided for each. These case studies are designed to highlight the major topics of the chapter in which they appear.

**Review Questions and Exercises** Each chapter presents a thorough and diverse set of review questions and exercises. They include Multiple Choice and True/False, Find the Error, Algorithm Workbench, and Short Answer.

**Programming Challenges** Each chapter offers a pool of programming challenges designed to solidify students' knowledge of topics at hand. In most cases the assignments present real-world problems to be solved.



**In the Spotlight.** Many of the chapters provide an *In the Spotlight* section that presents a programming problem, along with detailed, step-by-step analysis showing the student how to solve it.

## Supplements

### Companion Website

Many student resources are available for this book from the book's Companion Website. The following items are available at [www.pearsonhighered.com/gaddis](http://www.pearsonhighered.com/gaddis) using the Access Code bound into the front of the book:

- The source code for each example program in the book
- Access to the book's companion VideoNotes
- Appendixes B–M (listed in the Table of Contents)
- A collection of five valuable Case Studies (listed in the Table of Contents)
- Links to download the Java™ Edition Development Kit
- Links to download numerous programming environments, including jGRASP™, Eclipse™, TextPad™, NetBeans™, JCreator, and DrJava

### **MyCodeMate—Your Own T. A. Just a Click Away**



Addison-Wesley's *MyCodeMate* is a book-specific Web resource that provides tutorial help and evaluation of student programs. Example programs throughout the book and selected Programming Challenges from every chapter have been integrated into *MyCodeMate*. Using this tool, a student is able to write and compile programs from any computer with Internet access, and receive guidance and feedback on how to proceed and on how to address compiler error messages. Instructors can track each student's progress on Programming Challenges from the text or can develop projects of their own. A complimentary subscription to *MyCodeMate* is offered when the access code is ordered in a package with a new copy of this text. Subscriptions can also be purchased online. For more information visit [www.mycodemate.com](http://www.mycodemate.com), or contact your campus Addison-Wesley representative.

### Instructor Resources

The following supplements are available to qualified instructors only. Visit the Pearson Education Instructor Resource Center ([www.pearsonhighered.com/irc](http://www.pearsonhighered.com/irc)) or send an email to [computing@aw.com](mailto:computing@aw.com) for information on how to access them:

- Answers to all Review Questions in the text
- Solutions for all Programming Challenges in the text
- PowerPoint presentation slides for every chapter
- Computerized test bank

## Acknowledgments

There have been many helping hands in the development and publication of this text. I would like to thank the following faculty reviewers for their helpful suggestions and expertise during the production of this text:

Ahmad Abuhejleh  
*University of Wisconsin—River Falls*

Colin Archibald  
*Valencia CC*

Ijaz Awani  
*Savannah State University*

Dr. Charles W. Bane  
*Tarleton State University*

Dwight Barnett  
*Virginia Tech*

Asoke Bhattacharyya  
*Saint Xavier University, Chicago*

Marvin Bishop  
*Manhattan College*

Heather Booth  
*University Tennessee—Knoxville*

David Boyd  
*Valdosta University*

Julius Brandstatter  
*Golden Gate University*

Kim Cannon  
*Greenville Tech*

James Chegwidden  
*Tarrant County College*

Kay Chen  
*Bucks County Community College*

Brad Chilton  
*Tarleton State University*

Diane Christie  
*University of Wisconsin, Stout*

Cara Cocking  
*Marquette University*

Walter C. Daugherty  
*Texas A & M University*

Michael Doherty  
*University of the Pacific*

Jeanne M. Douglas  
*University of Vermont*

Sander Eller  
*California Polytechnic University—  
Pomona*

Brooke Estabrook-Fishinghawk  
*Mesa Community College*

Mike Fry  
*Lebanon Valley College*

Georgia R. Grant  
*College of San Mateo*

Chris Haynes  
*Indiana University*

Ric Heishman  
*Northern Virginia Community College*

Deedee Herrera  
*Dodge City Community College*

Mary Hovik  
*Lehigh Carbon Community College*

Brian Howard  
*DePauw University*

Norm Jacobson  
*University of California at Irvine*

Dr. Stephen Judd  
*University of Pennsylvania*

Harry Lichtbach  
*Evergreen Valley College*

Michael A. Long  
*California State University, Chico*

Tim Margush  
*University of Akron*

Blayne E. May eld  
*Oklahoma State University*

Scott McLeod  
*Riverside Community College*

Dean Mellas  
*Cerritos College*

Georges Merx  
*San Diego Mesa College*

Martin Meyers  
*California State University, Sacramento*

Pati Milligan  
*Baylor University*

Godfrey Muganda  
*North Central College*

Steve Newberry  
*Tarleton State University*

Lynne O'Hanlon  
*Los Angeles Pierce College*

Merrill Parker  
*Chattaonooga State Technical  
Community College*

Bryson R. Payne  
*North Georgia College and State  
University*

Rodney Pearson  
*Mississippi State University*

Peter John Polito  
*Springfield College*

Charles Robert Putnam  
*California State University, Northridge*

Dr. Y. B. Reddy  
*Grambling State University*

Carolyn Schauble  
*Colorado State University*

Bonnie Smith  
*Fresno City College*

Daniel Spiegel  
*Kutztown University*

Caroline St. Clair  
*North Central College*

Karen Stanton  
*Los Medanos College*

Peter H. Van Der Goes  
*Rose State College*

Tuan A Vo  
*Mt. San Antonio College*

Xiaoying Wang  
*University of Mississippi*

Special thanks goes to Chris Rich for his help in testing the Database chapter. I would like to thank my family for all the patience, love, and support they have shown me throughout this project. I would also like to thank everyone at Addison-Wesley for making the *Starting Out With* series so successful. I am extremely fortunate to have Michael Hirsch as my editor, and Stephanie Sellinger as editorial assistant. I am also fortunate to work with Yez Alyan and the computer science marketing team at Pearson. They do a great job getting my books out to the academic community. I had a great production team led by Jeff Holcomb, Managing Editor, and Kayla Smith-Tarbox, Production Project Manager. Thanks to you all!

## About the Author

Tony Gaddis is the principal author of the *Starting Out With* series of textbooks. Tony has nearly twenty years experience teaching computer science courses at Haywood Community College in North Carolina. He is a highly acclaimed instructor who was previously selected as the North Carolina Community College Teacher of the Year and has received the Teaching Excellence award from the National Institute for Staff and Organizational Development. Besides Java™ books, the Starting Out series includes introductory books using the C11 programming language, Microsoft® Visual Basic®, Microsoft® C#®, Python, Programming Logic and Design, and Alice, all published by Addison-Wesley.