

LEARNING MADE EASY



6th Edition

Java[®]

ALL-IN-ONE

for
dummies[®]
A Wiley Brand



Books
in one!

Doug Lowe

Author of *Networking For
Dummies*, 11th Edition



Java[®]

ALL-IN-ONE

6th Edition

by Doug Lowe

**for
dummies[®]**
A Wiley Brand

Java® All-in-One For Dummies®, 6th Edition

Published by: **John Wiley & Sons, Inc.**, 111 River Street, Hoboken, NJ 07030-5774, www.wiley.com

Copyright © 2020 by John Wiley & Sons, Inc., Hoboken, New Jersey

Published simultaneously in Canada

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 the prior written permission of the Publisher. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permissions>.

Trademarks: Wiley, For Dummies, the Dummies Man logo, Dummies.com, Making Everything Easier, and related trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and may not be used without written permission. Java is a registered trademark of Oracle America, Inc. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc. is not associated with any product or vendor mentioned in this book.

<p>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</p>

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, please contact our Customer Care Department within the U.S. at 877-762-2974, outside the U.S. at 317-572-3993, or fax 317-572-4002. For technical support, please visit <https://hub.wiley.com/community/support/dummies>.

Wiley publishes in a variety of print and electronic formats and by print-on-demand. Some material included with standard print versions of this book may not be included in e-books or in print-on-demand. If this book refers to media such as a CD or DVD that is not included in the version you purchased, you may download this

material at <http://booksupport.wiley.com>. For more information about Wiley products, visit www.wiley.com.

Library of Congress Control Number: 2020941824

ISBN 978-1-119-68045-1 (pbk); ISBN 978-1-119-68048-2 (ebk); ISBN 978-1-119-68051-2 (ebk)

Java® All-in-One For Dummies®

To view this book's Cheat Sheet, simply go to www.dummies.com and search for “Java® All-in-One For Dummies Cheat Sheet” in the Search box.

Table of Contents

Cover

Introduction

[About This Book](#)

[Foolish Assumptions](#)

[Icons Used in This Book](#)

[Beyond the Book](#)

[Where to Go from Here](#)

Book 1: Java Basics

Chapter 1: Welcome to Java

[What Is Java, and Why Is It So Great?](#)

[Java versus Other Languages](#)

[Important Features of the Java Language](#)

[On the Downside: Java's Weaknesses](#)

[Java Version Insanity](#)

[What's in a Name?](#)

Chapter 2: Installing and Using Java Tools

[Downloading and Installing the Java Development Kit](#)

[Using Java's Command-Line Tools](#)

[Using Java Documentation](#)

Chapter 3: Working with TextPad

[Downloading and Installing TextPad](#)

[Editing Source Files](#)

[Compiling a Program](#)

[Running a Java Program](#)

Book 2: Programming Basics

Chapter 1: Java Programming Basics

[Looking at the Infamous Hello, World! Program](#)

[Dealing with Keywords](#)

[Working with Statements](#)

[Working with Blocks](#)

[Creating Identifiers](#)

[Crafting Comments](#)

[Introducing Object-Oriented Programming](#)

[Importing Java API Classes](#)

Chapter 2: Working with Variables and Data Types

[Declaring Variables](#)

[Initializing Variables](#)

[Using Final Variables \(Constants\)](#)

[Working with Primitive Data Types](#)

[Working with Strings](#)

[Converting and Casting Numeric Data](#)

[Thinking Inside the Box](#)

[Understanding Scope](#)

[Shadowing Variables](#)

[Printing Data with System.out](#)

[Getting Input with the Scanner Class](#)

[Getting Input with the JOptionPane Class](#)

[Using enum to Create Your Own Data Types](#)

Chapter 3: Working with Numbers and Expressions

[Working with Arithmetic Operators](#)

[Dividing Integers](#)

[Combining Operators](#)

[Using the Unary Plus and Minus Operators](#)

[Using Increment and Decrement Operators](#)

[Using the Assignment Operator](#)

[Using Compound Assignment Operators](#)

[Using the Math Class](#)

[Formatting Numbers](#)

[Recognizing Weird Things about Java Math](#)

Chapter 4: Making Choices

[Using Simple Boolean Expressions](#)

[Using if Statements](#)

[Using Mr. Spock's Favorite Operators \(Logical Ones, of Course\)](#)

[Using the Conditional Operator](#)

[Comparing Strings](#)

Chapter 5: Going Around in Circles (Or, Using Loops)

[Using Your Basic while Loop](#)

[Breaking Out of a Loop](#)

[Looping Forever](#)

[Using the continue Statement](#)

[Running do-while Loops](#)

[Validating Input from the User](#)

[Using the Famous for Loop](#)

[Nesting Your Loops](#)

Chapter 6: Pulling a Switcheroo

[Battling else-if Monstrosities](#)

[Using the switch Statement](#)

[Creating Character Cases](#)

[Intentionally Leaving Out a Break Statement](#)

[Switching with Strings](#)

[Enhanced Switch Features with Java 13](#)

Chapter 7: Adding Some Methods to Your Madness

[The Joy of Methods](#)

[The Basics of Making Methods](#)

[Methods That Return Values](#)

[Methods That Take Parameters](#)

Chapter 8: Handling Exceptions

[Understanding Exceptions](#)

[Catching Exceptions](#)

[Handling Exceptions with a Preemptive Strike](#)

[Catching All Exceptions at Once](#)

[Displaying the Exception Message](#)

[Using a finally Block](#)

[Handling Checked Exceptions](#)

[Throwing Your Own Exceptions](#)

Book 3: Object-Oriented Programming

Chapter 1: Understanding Object-Oriented Programming

[What Is Object-Oriented Programming?](#)

[Understanding Objects](#)

[Understanding the Life Cycle of an Object](#)

[Working with Related Classes](#)

[Designing a Program with Objects](#)

[Diagramming Classes with UML](#)

Chapter 2: Making Your Own Classes

[Declaring a Class](#)

[Working with Members](#)

[Using Getters and Setters](#)

[Overloading Methods](#)

[Creating Constructors](#)

[Finding More Uses for the this Keyword](#)

[Using Initializers](#)

[Using Records](#)

Chapter 3: Working with Statics

[Understanding Static Fields and Methods](#)

[Working with Static Fields](#)

[Using Static Methods](#)

[Counting Instances](#)

[Preventing Instances](#)

[Using Static Initializers](#)

Chapter 4: Using Subclasses and Inheritance

[Introducing Inheritance](#)

[Creating Subclasses](#)

[Overriding Methods](#)

[Protecting Your Members](#)

[Using this and super in Your Subclasses](#)

[Understanding Inheritance and Constructors](#)

[Using final](#)

[Casting Up and Down](#)

[Determining an Object's Type](#)

[Poly What?](#)

[Creating Custom Exceptions](#)

Chapter 5: Using Abstract Classes and Interfaces

[Using Abstract Classes](#)

[Using Interfaces](#)

[More Things You Can Do with Interfaces](#)

[Using Additional Interface Method Types](#)

Chapter 6: Using the Object and Class Classes

[The Mother of All Classes: Object](#)

[The toString Method](#)

[The equals Method](#)

[The clone Method](#)

[The Class Class](#)

Chapter 7: Using Inner Classes, Anonymous Classes, and Lambda Expressions

[Declaring Inner Classes](#)

[Using Static Inner Classes](#)

[Using Anonymous Inner Classes](#)

[Using Lambda Expressions](#)

Chapter 8: Working with Packages and the Java Module System

[Working with Packages](#)

[Putting Your Classes in a JAR File](#)

[Using Javadoc to Document Your Classes](#)

[Using the Java Module System](#)

Book 4: Strings, Arrays, and Collections

Chapter 1: Working with Strings

[Reviewing Strings](#)

[Using the String Class](#)

[Using the StringBuilder and StringBuffer Classes](#)

[Using the CharSequence Interface](#)

Chapter 2: Using Arrays

[Understanding Arrays](#)

[Creating Arrays](#)

[Initializing an Array](#)

[Using for Loops with Arrays](#)

[Solving Homework Problems with Arrays](#)

[Using the Enhanced for Loop](#)

[Using Arrays with Methods](#)

[Using Varargs](#)

[Using Two-Dimensional Arrays](#)

[Working with a Fun but Complicated Example: A Chessboard](#)

[Using the Arrays Class](#)

Chapter 3: Using the ArrayList Class

[Understanding the ArrayList Class](#)

[Creating an ArrayList Object](#)

[Adding Elements](#)

[Accessing Elements](#)

[Printing an ArrayList](#)

[Using an Iterator](#)

[Updating Elements](#)

[Deleting Elements](#)

Chapter 4: Using the LinkedList Class

[Understanding the LinkedList Class](#)

[Creating a LinkedList](#)

[Adding Items to a LinkedList](#)

[Retrieving Items from a LinkedList](#)

[Updating LinkedList Items](#)

[Removing LinkedList Items](#)

Chapter 5: Creating Generic Collection Classes

[Why Generics?](#)

[Creating a Generic Class](#)

[A Generic Stack Class](#)

[Using Wildcard-Type Parameters](#)

[A Generic Queue Class](#)

Chapter 6: Using Bulk Data Operations with Collections

[Looking At a Basic Bulk Data Operation](#)

[Looking Closer at the Stream Interface](#)

[Using Parallel Streams](#)

Book 5: Programming Techniques

Chapter 1: Programming Threads

[Understanding Threads](#)

[Creating a Thread](#)

[Implementing the Runnable Interface](#)

[Creating Threads That Work Together](#)

[Using an Executor](#)

[Synchronizing Methods](#)

[Creating a Lock](#)

[Coping with Threadus Interruptus](#)

Chapter 2: Using Regular Expressions

[Creating a Program for Experimenting with Regular Expressions](#)

[Performing Basic Character Matching](#)

[Using Regular Expressions in Java Programs](#)

Chapter 3: Using Recursion

[Calculating the Classic Factorial Example](#)

[Displaying Directories](#)

[Writing Your Own Sorting Routine](#)

Chapter 4: Working with Dates and Times

[Pondering How Time is Represented](#)

[Picking the Right Date and Time Class for Your Application](#)

[Using the now Method to Create a Date-Time Object](#)

[Using the parse Method to Create a Date-Time Object](#)

[Using the of Method to Create a Date-Time Object](#)

[Looking Closer at the LocalDate Class](#)

[Extracting Information About a Date](#)

[Comparing Dates](#)

[Calculating with Dates](#)

[Formatting Dates](#)

[Looking at a Fun Birthday Calculator](#)

Chapter 5: IoT Programming with Raspberry Pi

[Introducing the Raspberry Pi](#)
[Setting Up a Raspberry Pi](#)
[Installing Java on a Raspberry Pi](#)
[Installing the Pi4J Library](#)
[Configuring the Geany Integrated Development Environment for Java Development](#)
[Examining GPIO Ports](#)
[Connecting an LED to a GPIO Port](#)
[Building a Raspberry Pi LED Circuit](#)
[Examining the Pi4J Library](#)
[Importing GPIO Types](#)
[Instantiating a GpioController](#)
[Provisioning GPIO Pins](#)
[Controlling the Pin State](#)
[The Morse Code Program](#)
[The Cylon Eyes Program](#)
[Working with Input Pins](#)
[Running the Button Switcher Program](#)
[Finding a Better Way to Handle Input Events](#)

Book 6: JavaFX

Chapter 1: Hello, JavaFX!

[Perusing the Possibilities of JavaFX](#)
[Getting Ready to Run JavaFX](#)
[Looking at a Simple JavaFX Program](#)
[Importing JavaFX Packages](#)
[Extending the Application Class](#)
[Launching the Application](#)
[Overriding the start Method](#)
[Creating a Button](#)
[Handling an Action Event](#)
[Creating a Layout Pane](#)

[Making a Scene](#)

[Setting the Stage](#)

[Examining the Click Counter Program](#)

Chapter 2: Handling Events

[Examining Events](#)

[Handling Events](#)

[Implementing the EventHandler Interface](#)

[Handling Events with Inner Classes](#)

[Handling Events with Anonymous Inner Classes](#)

[Using Lambda Expressions to Handle Events](#)

Chapter 3: Setting the Stage and Scene Layout

[Examining the Stage Class](#)

[Examining the Scene Class](#)

[Switching Scenes](#)

[Creating an Alert Box](#)

[Exit, Stage Right](#)

Chapter 4: Using Layout Panes to Arrange Your Scenes

[Working with Layout Panes](#)

[Using the HBox Layout](#)

[Spacing Things Out](#)

[Adding Space with Margins](#)

[Adding Space by Growing Nodes](#)

[Using the VBox Layout](#)

[Aligning Nodes in a Layout Pane](#)

[Using the Flow Layout](#)

[Using the Border Layout](#)

[Using the GridPane Layout](#)

Chapter 5: Getting Input from the User

[Using Text Fields](#)

[Validating Numeric Data](#)

[Using Check Boxes](#)

[Using Radio Buttons](#)

Chapter 6: Choosing from a List

[Using Choice Boxes](#)

[Working with Observable Lists](#)

[Listening for Selection Changes](#)

[Using Combo Boxes](#)

[Using List Views](#)

[Using Tree Views](#)

Book 7: Web Programming

Chapter 1: Creating Servlets

[Understanding Servlets](#)

[Using Tomcat](#)

[Creating a Simple Servlet](#)

[Running a Servlet](#)

[Improving the HelloWorld Servlet](#)

[Getting Input from the User](#)

[Using Classes in a Servlet](#)

Chapter 2: Using JavaServer Pages

[Understanding JavaServer Pages](#)

[Using UEL Expressions](#)

[Looking at Core Tags](#)

[Formatting Numbers](#)

[Considering the Controller Servlet](#)

[The ListMovies Application Meets JSP](#)

Chapter 3: Using JavaBeans

[Getting to Know JavaBeans](#)

[Looking Over a Sample Bean](#)

[Using Beans with JSP Pages](#)

[Scoping Your Beans](#)

Chapter 4: Using HttpClient

[Understanding HTTP](#)

[Getting Started with Java's HTTP Client Library](#)

[Putting It All Together](#)

[The HTTP Tester Program](#)

[Index](#)

[About the Author](#)

[Advertisement Page](#)

[Connect with Dummies](#)

[End User License Agreement](#)

List of Tables

Book 1 Chapter 2

[TABLE 2-1 Subfolders of the JDK Root Folder](#)

[TABLE 2-2 Java Compiler Options](#)

[TABLE 2-3 Common Java Command Options](#)

Book 2 Chapter 1

[TABLE 1-1 Java's Keywords](#)

Book 2 Chapter 2

[TABLE 2-1 Java's Primitive Types](#)

[TABLE 2-2 Escape Sequences for Character Constants](#)

[TABLE 2-3 Wrapper Classes for the Primitive Types](#)

[TABLE 2-4 Methods That Convert Strings to Numeric Primitive Types](#)

[TABLE 2-5 Static Fields of the System Object](#)

[TABLE 2-6 Scanner Class Methods That Get Input Values](#)

[TABLE 2-7 Scanner Class Methods That Check for Valid Input Values](#)

Book 2 Chapter 3

[TABLE 3-1 Java's Arithmetic Operators](#)

[TABLE 3-2 Compound Assignment Operators](#)

[TABLE 3-3 Constants of the Math Class](#)

[TABLE 3-4 Commonly Used Mathematical Functions Provided by the Math Class](#)

[TABLE 3-5 Rounding Functions Provided by the Math Class](#)

[TABLE 3-6 Methods of the NumberFormat Class](#)

[TABLE 3-7 Special Constants of the float and double Classes](#)

Book 2 Chapter 4

[TABLE 4-1 Relational Operators](#)

[TABLE 4-2 Logical Operators](#)

Book 2 Chapter 8

[TABLE 8-1 Methods of the Exception Class](#)

Book 3 Chapter 1

[TABLE 1-1 Visibility Indicators for Class Variables and Methods](#)

Book 3 Chapter 6

[TABLE 6-1 Methods of the Object Class](#)

Book 3 Chapter 8

[TABLE 8-1 Options for the jar Command](#)

[TABLE 8-2 Commonly Used Javadoc Tags](#)

Book 4 Chapter 1

[TABLE 1-1 String Class Methods](#)

[TABLE 1-2 StringBuilder Methods](#)

Book 4 Chapter 2

[TABLE 2-1 Handy Methods of the Arrays Class](#)

Book 4 Chapter 3

[TABLE 3-1 The ArrayList Class](#)

[TABLE 3-2 The Iterator Interface](#)

Book 4 Chapter 4

[TABLE 4-1 The LinkedList Class](#)

Book 4 Chapter 6

[TABLE 6-1 The Stream and Related Interfaces](#)

Book 5 Chapter 1

[TABLE 1-1 Constructors and Methods of the Thread Class](#)

Book 5 Chapter 2

[TABLE 2-1 Character Classes](#)

[TABLE 2-2 Quantifiers](#)

Book 5 Chapter 4

[TABLE 4-1 Ten Date-Time Classes in java.time](#)

[TABLE 4-2 Date-Time of Methods](#)

[TABLE 4-3 Methods of the LocalDate Class](#)

[TABLE 4-4 Formatting Characters for the DateTimeFormatter Class](#)

Book 5 Chapter 5

[TABLE 5-1 GPIO Ports and Pin Locations](#)

[TABLE 5-2 Commonly Used GPIO Members](#)

[TABLE 5-3 The GpioFactory Class](#)

[TABLE 5-4 The GpioController Interface](#)

[TABLE 5-5 The PinMode enum](#)

[TABLE 5-6 The GpioPinDigitalOutput Interface](#)

[TABLE 5-7 How the LEDs Are Connected](#)

[TABLE 5-8 The PinPullResistance enum](#)

[TABLE 5-9 The GpioPinDigitalInput Interface](#)

Book 6 Chapter 2

[TABLE 2-1 Commonly Used Event Classes](#)

[TABLE 2-2 The EventHandler Interface](#)

Book 6 Chapter 3

[TABLE 3-1 Commonly Used Methods of the Stage Class](#)

[TABLE 3-2 Commonly Used Constructors and Methods of the Scene class](#)

[TABLE 3-3 Commonly Used Constructors and Methods of the Alert class](#)

Book 6 Chapter 4

[TABLE 4-1 HBox Constructors and Methods](#)

[TABLE 4-2 Insets Constructors](#)

[TABLE 4-3 The Priority Enumeration](#)

[TABLE 4-4 VBox Constructors and Methods](#)

[TABLE 4-5 The Pos Enumeration](#)

[TABLE 4-6 FlowPane Constructors and Methods](#)

[TABLE 4-7 BorderPane Constructors and Methods](#)

[TABLE 4-8 GridPane Constructors and Methods](#)

[TABLE 4-9 The ColumnConstraints Class](#)

[TABLE 4-10 The RowConstraints Class](#)

Book 6 Chapter 5

[TABLE 5-1 Handy TextField Constructors and Methods](#)

[TABLE 5-2 Methods That Convert Strings to Numbers](#)

[TABLE 5-3 Notable CheckBox Constructors and Methods](#)

[TABLE 5-4 Various RadioButton Constructors and Methods](#)

Book 6 Chapter 6

[TABLE 6-1 Common ChoiceBox Constructors and Methods](#)

[TABLE 6-2 Commonly Used ObservableList Methods](#)

[TABLE 6-3 Common ComboBox Constructors and Methods](#)

[TABLE 6-4 Common ListView Constructors and Methods](#)

[TABLE 6-5 The TreeItem Class](#)

[TABLE 6-6 The TreeView Class](#)

Book 7 Chapter 1

[TABLE 1-1 The HttpServlet Class](#)

[TABLE 1-2 Just Enough HTML to Get By](#)

Book 7 Chapter 2

[TABLE 2-1 UEL Expression Operators](#)

[TABLE 2-2 JSTL Tag Prefixes and <taglib> Directives](#)

[TABLE 2-3 Core Tags](#)

[TABLE 2-4 fmt:formatNumber Attributes](#)

Book 7 Chapter 3

[TABLE 3-1 JSP Tags for Working with Beans](#)

[TABLE 3-2 Scope Settings](#)

Book 7 Chapter 4

[TABLE 4-1 The HttpClient Class](#)

[TABLE 4-2 The HttpRequest and Its Builder Interface](#)

[TABLE 4-3 The `HttpResponse` Class](#)

[TABLE 4-4 The `BodyHandlers` Class](#)

List of Illustrations

Book 1 Chapter 2

[FIGURE 2-1: The Environment Variables dialog box.](#)

[FIGURE 2-2: Editing the Path variable.](#)

[FIGURE 2-3: A home page from the Java SE 14 API documentation.](#)

[FIGURE 2-4: The documentation page for the `String` class.](#)

Book 1 Chapter 3

[FIGURE 3-1: Configuring tools in TextPad.](#)

[FIGURE 3-2: Editing a Java file in TextPad.](#)

[FIGURE 3-3: Error messages displayed by the Java compiler.](#)

[FIGURE 3-4: Running a program.](#)

Book 2 Chapter 1

[FIGURE 1-1: The class in Listing 1-4 displays this dialog box.](#)

Book 2 Chapter 2

[FIGURE 2-1: Numeric type conversions that are done automatically.](#)

[FIGURE 2-2: A dialog box displayed by the `JOptionPane` class.](#)

Book 2 Chapter 4

[FIGURE 4-1: The flowchart for an if statement.](#)

[FIGURE 4-2: The flowchart for an if-else statement.](#)

[FIGURE 4-3: The flowchart for a sequence of else-if statements.](#)

Book 2 Chapter 5

[FIGURE 5-1: The flowchart for a while loop.](#)

[FIGURE 5-2: The flowchart for a do-while loop.](#)

[FIGURE 5-3: The flowchart for a for loop.](#)

Book 2 Chapter 6

[FIGURE 6-1: The flowchart for a switch statement.](#)

Book 2 Chapter 8

[FIGURE 8-1: This program has slipped into the Exception Zone.](#)

[FIGURE 8-2: Why you have to call next to discard the invalid input.](#)

Book 3 Chapter 1

[FIGURE 1-1: Three-layered design.](#)

[FIGURE 1-2: A simple class diagram.](#)

[FIGURE 1-3: A class.](#)

Book 3 Chapter 4

[FIGURE 4-1: The hierarchy of exception classes.](#)

Book 3 Chapter 5

[FIGURE 5-1: The Tick Tock application in action.](#)

Book 3 Chapter 8

[FIGURE 8-1: A Javadoc index page.](#)

[FIGURE 8-2: Documentation for the Employee class.](#)

Book 4 Chapter 2

[FIGURE 2-1: A classic chessboard.](#)

Book 4 Chapter 5

[FIGURE 5-1: A Raspberry Pi 4.](#)

[FIGURE 5-2: Geany, the IDE for Raspberry Pi programming.](#)

[FIGURE 5-3: The Set Build Commands window.](#)

[FIGURE 5-4: Raspberry Pi header pins \(Pi versions 2, 3, and 4\).](#)

[FIGURE 5-5: Connecting an LED to a GPIO port.](#)

[FIGURE 5-6: The Raspberry Pi LED flasher.](#)

[FIGURE 5-7: A schematic diagram for the Cylon Eyes circuit.](#)

[FIGURE 5-8: The breadboard layout for the Cylon Eyes circuit.](#)

[FIGURE 5-9: Active-high and active-low input circuits.](#)

[FIGURE 5-10: A schematic diagram for the Button Switcher circuit.](#)

[FIGURE 5-11: The completed Button Switcher circuit.](#)

Book 6 Chapter 1

[FIGURE 1-1: A typical JavaFX program.](#)

[FIGURE 1-2: The Click Me program.](#)

[FIGURE 1-3: The Click Counter program in action.](#)

Book 6 Chapter 2

[FIGURE 2-1: The AddSubtract1 program.](#)

Book 6 Chapter 3

[FIGURE 3-1: The SceneSwitcher program.](#)

[FIGURE 3-2: An Alert dialog box.](#)

Book 6 Chapter 4

[FIGURE 4-1: Four commonly used types of layout panes.](#)

[FIGURE 4-2: Using a spacer node to space out buttons in an HBox pane.](#)

[FIGURE 4-3: Three buttons centered in a VBox layout pane.](#)

[FIGURE 4-4: A flow layout pane with five buttons.](#)

[FIGURE 4-5: How the border layout carves things up.](#)

[FIGURE 4-6: Sketching out a panel.](#)

[FIGURE 4-7: The Pizza Order application in action.](#)

Book 6 Chapter 5

[FIGURE 5-1: The Role Player application in action.](#)

[FIGURE 5-2: Three check boxes.](#)

[FIGURE 5-3: A frame with three radio buttons.](#)

Book 6 Chapter 6

[FIGURE 6-1: A scene with a choice box.](#)

[FIGURE 6-2: A combo box.](#)

[FIGURE 6-3: A list view control.](#)

[FIGURE 6-4: A horizontal list view control.](#)

[FIGURE 6-5: A tree view control.](#)

[FIGURE 6-6: A tree view control with the root node hidden.](#)

Book 7 Chapter 1

[FIGURE 1-1: Tomcat's home page.](#)

[FIGURE 1-2: Tomcat's server status page.](#)

[FIGURE 1-3: The HelloWorld servlet displayed in a browser.](#)

[FIGURE 1-4: A simple input form.](#)

[FIGURE 1-5: The ListMovies servlet.](#)

Book 7 Chapter 2

[FIGURE 2-1: The ListMovies JSP in action.](#)

Book 7 Chapter 3

[FIGURE 3-1: The Triangle.jsp page displayed in a browser.](#)

[FIGURE 3-2: A super-simple shopping cart application.](#)

Introduction

Welcome to *Java All-in-One For Dummies*, 6th Edition — the one Java book that’s designed to replace an entire shelf full of the dull, tedious titles you’d otherwise have to buy. This book contains all the basic information you need to know to get going with Java programming, starting with writing statements and using variables and ending with techniques for writing programs that use animation and play games. Along the way, you find plenty of not-so-basic information about programming user interfaces, working with classes and objects, creating web applications, and dealing with files and databases.

You can (and probably should, eventually) buy separate books on each of these topics. It won’t take long before your bookshelf is bulging with 10,000 or more pages of detailed information about every imaginable nuance of Java programming. But before you’re ready to tackle each of those topics in depth, you need to get a bird’s-eye picture. That’s what this book is about.

And if you already *own* 10,000 pages or more of Java information, you may be overwhelmed by the amount of detail and wonder, “Do I really need to *read* 1,200 pages about JSP just to create a simple web page? And do I really *need* a six-pound book on JavaFX?” Truth is, most 1,200-page programming books have about 200 pages of really useful information — the kind you use every day — and about 1,000 pages of excruciating details that apply mostly if you’re writing guidance-control programs for nuclear missiles or trading systems for the New York Stock Exchange.

The basic idea here is that I’ve tried to wring out the 100-or-so most useful pages of information on these

different Java programming topics: setup and configuration, basic programming, object-oriented programming, advanced programming techniques, JavaFX, file and database programming, web programming, and animation and game programming. Thus you get a nice, trim book.

So whether you're just getting started with Java programming or you're a seasoned pro, you've found the right book.

About This Book

Java All-in-One For Dummies, 6th Edition, is a reference for all the great things (and maybe a few not-so-great things) that you may need to know when you're writing Java programs. You can, of course, buy a huge 1,200-page book on each of the programming topics covered in this book. But then, who would carry them home from the bookstore for you? And where would you find the shelf space to store them? And when will you find the time to read them?

In this book, all the information you need is conveniently packaged for you in-between one set of covers. And all of the information is current for the newest release of Java, known as JDK 14. This book doesn't pretend to be a comprehensive reference for every detail on every possible topic related to Java programming. Instead, it shows you how to get up and running fast so that you have more time to do the things you really want to do. Designed using the easy-to-follow *For Dummies* format, this book helps you get the information you need without laboring to find it.

Java All-in-One For Dummies, 6th Edition, is a big book made up of nine smaller books — minibooks, if you will.

Each of these minibooks covers the basics of one key element of programming, such as installing Java and compiling and running programs, or using basic Java statements, or using JavaFX to write GUI applications.

Whenever one big thing is made up of several smaller things, confusion is always a possibility. That's why this book has multiple access points. At the beginning is a detailed table of contents that covers the entire book. Then each minibook begins with a minitable of contents that shows you at a minigance what chapters are included in that minibook. Useful running heads appear at the top of each page to point out the topic discussed on that page. And handy thumbtabs run down the side of the pages to help you find each minibook quickly. Finally, a comprehensive index lets you find information anywhere in the entire book.

Foolish Assumptions

You and I have never met, so it is difficult for me to make any assumptions about why you are interested in this book. However, let's start with a few basic assumptions:

- » **You own or have access to a relatively modern computer.** The examples were created on a Windows computer, but you can learn to program in Java just as easily on a Mac or Linux computer.
- » **You're an experienced computer user.** In other words, I assume that you know the basics of using your computer, such as starting programs and working with the file system.
- » **You're interested in learning how to write programs in the Java language.** Since that's what this book teaches, it's a fair assumption.

I do *not* make any assumptions about any previous programming experience in Java or in any other programming language. Nor do I make any assumptions about *why* you want to learn about Java programming. There are all sorts of valid reasons for learning Java. Some want to learn Java for professional reasons; maybe you want to become a professional Java programmer, or maybe you are a C# or C++ programmer who occasionally needs to work in Java. On the other hand, maybe you think programming in Java would make an interesting hobby.

Regardless of your motivation, I *do* assume that you are a reasonably intelligent person. You don't have to have a degree in advanced physics, or a degree in anything at all for that matter, to master Java programming. All you have to be is someone who wants to learn and isn't afraid to try.

Icons Used in This Book

Like any *For Dummies* book, this book is chock-full of helpful icons that draw your attention to items of particular importance. You find the following icons throughout this book:



WARNING Danger, Will Robinson! This icon highlights information that may help you avert disaster.



REMEMBER Did I tell you about the memory course I took?



TIP

Pay special attention to this icon; it lets you know that some particularly useful tidbit is at hand.



TECHNICAL
STUFF

Hold it — overly technical stuff is just around the corner. Obviously, because this is a programming book, almost every paragraph of the next 900 or so pages could get this icon. So I reserve it for those paragraphs that go into greater depth, down into explaining how something works under the covers — probably deeper than you really need to know to use a feature, but often enlightening.

Beyond the Book

In addition to the material in the print or e-book you're reading right now, this product also comes with some access-anywhere goodies on the web. Check out the free Cheat Sheet for more on Java. To get this Cheat Sheet, simply go to www.dummies.com and type **Java All-in-One For Dummies Cheat Sheet** in the Search box.

Visit www.dummies.com/go/javaaiofd6e to dive even deeper into Java. You can find and download the code used in the book at that link. You can also download a bonus minibook covering how to use Java with files and databases.

Where to Go from Here

This isn't the kind of book you pick up and read from start to finish, as if it were a cheap novel. If I ever see

you reading it at the beach, I'll kick sand in your face. Beaches are for reading romance novels or murder mysteries, not programming books. Although you could read straight through from start to finish, this book is a reference book, the kind you can pick up, open to just about any page, and start reading. You don't have to memorize anything in this book. It's a "need-to-know" book: You pick it up when you need to know something. Need a reminder on the constructors for the `ArrayList` class? Pick up the book. Can't remember the goofy syntax for anonymous inner classes? Pick up the book. After you find what you need, put the book down and get on with your life.