The Java® Tutorial

A Short Course on the Basics

Sixth Edition

Raymond Gallardo, Scott Hommel, Sowmya Kannan,
Joni Gordon, Sharon Biocca Zakhour

ORACLE

The Java® Tutorial

Sixth Edition

Java Tutorial, The: A Short Course on the Basics

Table of Contents

Contents

Preface

About the Authors

Chapter 1 Getting Started

The Java Technology Phenomenon

The Java Programming Language

The Java Platform

What Can Java Technology Do?

How Will Java Technology Change My Life?

The "Hello World!" Application

"Hello World!" for the NetBeans IDE

"Hello World!" for Microsoft Windows

"Hello World!" for Solaris and Linux

A Closer Look at the "Hello World!" Application

Source Code Comments

The HelloWorldApp Class Definition

The main Method

Common Problems (and Their Solutions)

Compiler Problems

Runtime Problems

Questions and Exercises: Getting Started

Questions

Exercises

Pearson

Table of Contents

Answers

Chapter 2 Object-Oriented Programming Concepts

What Is an Object?

What Is a Class?

What Is Inheritance?

What Is an Interface?

What Is a Package?

Questions and Exercises: Object-Oriented Programming Concepts

Questions

Exercises

Answers

Chapter 3 Language Basics

Variables

Naming

Primitive Data Types

Arrays

Summary of Variables

Questions and Exercises: Variables

Operators

Assignment, Arithmetic, and Unary Operators

Equality, Relational, and Conditional Operators

Bitwise and Bit Shift Operators

Summary of Operators

Questions and Exercises: Operators

Expressions, Statements, and Blocks

Expressions

Statements

Blocks

Table of Contents

- Questions and Exercises: Expressions, Statements, and Blocks
- Control Flow Statements
    - The if-then and if-then-else Statements
    - The switch Statement
    - The while and do-while Statements
    - The for Statement
    - Branching Statements
    - Summary of Control Flow Statements
    - Questions and Exercises: Control Flow Statements
- Chapter 4 Classes and Objects
    - Classes
        - Declaring Classes
        - Declaring Member Variables
        - Defining Methods
        - Providing Constructors for Your Classes
        - Passing Information to a Method or a Constructor
    - Objects
        - Creating Objects
        - Using Objects
    - More on Classes
        - Returning a Value from a Method
        - Using the this Keyword
        - Controlling Access to Members of a Class
        - Understanding Class Members
        - Initializing Fields
        - Summary of Creating and Using Classes and Objects
        - Questions and Exercises: Classes
        - Questions and Exercises: Objects
    - Nested Classes

Table of Contents

Why Use Nested Classes?

Static Nested Classes

Inner Classes

Shadowing

Serialization

Inner Class Example

Local and Anonymous Classes

Modifiers

Local Classes

Anonymous Classes

Lambda Expressions

When to Use Nested Classes, Local Classes, Anonymous Classes, and Lambda Expressions

Questions and Exercises: Nested Classes

Enum Types

Questions and Exercises: Enum Types

Chapter 5 Annotations

Annotations Basics

The Format of an Annotation

Where Annotations Can Be Used

Declaring an Annotation Type

Predefined Annotation Types

Annotation Types Used by the Java Language

Annotations That Apply to Other Annotations

Type Annotations and Pluggable Type Systems

Repeating Annotations

Step 1: Declare a Repeatable Annotation Type

Step 2: Declare the Containing Annotation Type

Retrieving Annotations

Table of Contents

Design Considerations
Questions and Exercises: Annotations
Questions
Exercise
Answers
Chapter 6 Interfaces and Inheritance
Interfaces
Interfaces in Java
Interfaces as APIs
Defining an Interface
Implementing an Interface
Using an Interface as a Type
Evolving Interfaces
Default Methods
Summary of Interfaces
Questions and Exercises: Interfaces
Inheritance
The Java Platform Class Hierarchy
An Example of Inheritance
What You Can Do in a Subclass
Private Members in a Superclass
Casting Objects
Multiple Inheritance of State, Implementation, and Type
Overriding and Hiding Methods
Polymorphism
Hiding Fields
Using the Keyword super
Object as a Superclass
Writing Final Classes and Methods

Table of Contents

Abstract Methods and Classes

Summary of Inheritance

Questions and Exercises: Inheritance

Chapter 7 Generics

Why Use Generics?

Generic Types

A Simple Box Class

A Generic Version of the Box Class

Type Parameter Naming Conventions

Invoking and Instantiating a Generic Type

The Diamond

Multiple Type Parameters

Parameterized Types

Raw Types

Generic Methods

Bounded Type Parameters

Multiple Bounds

Generic Methods and Bounded Type Parameters

Generics, Inheritance, and Subtypes

Generic Classes and Subtyping

Type Inference

Type Inference and Generic Methods

Type Inference and Instantiation of Generic Classes

Type Inference and Generic Constructors of Generic and Nongeneric Classes

Target Types

Wildcards

Upper-Bounded Wildcards

Unbounded Wildcards

Table of Contents

Lower-Bounded Wildcards

Wildcards and Subtyping

Wildcard Capture and Helper Methods

Guidelines for Wildcard Use

Type Erasure

Erasure of Generic Types

Erasure of Generic Methods

Effects of Type Erasure and Bridge Methods

Nonreifiable Types and Varargs Methods

Restrictions on Generics

Cannot Instantiate Generic Types with Primitive Types

Cannot Create Instances of Type Parameters

Cannot Declare Static Fields Whose Types Are Type Parameters

Cannot Use Casts or instanceof with Parameterized Types

Cannot Create Arrays of Parameterized Types

Cannot Create, Catch, or Throw Objects of Parameterized Types

Cannot Overload a Method Where the Formal Parameter Types of Each Overload Erase to the Same Raw Type

Questions and Exercises: Generics

Answers

Chapter 8 Packages

Creating and Using Packages

Creating a Package

Naming a Package

Using Package Members

Managing Source and Class Files

Summary of Creating and Using Packages

Questions and Exercises: Creating and Using Packages

Questions

Table of Contents

Exercises

Answers

Chapter 9 Numbers and Strings

Numbers

The Numbers Classes

Formatting Numeric Print Output

Beyond Basic Arithmetic

Autoboxing and Unboxing

Summary of Numbers

Questions and Exercises: Numbers

Characters

Escape Sequences

Strings

Creating Strings

String Length

Concatenating Strings

Creating Format Strings

Converting between Numbers and Strings

Manipulating Characters in a String

Comparing Strings and Portions of Strings

The StringBuilder Class

Summary of Characters and Strings

Questions and Exercises: Characters and Strings

Chapter 10 Exceptions

What Is an Exception?

The Catch or Specify Requirement

The Three Kinds of Exceptions

Bypassing Catch or Specify

Catching and Handling Exceptions

Table of Contents

The try Block

The catch Blocks

The finally Block

The try-with-resources Statement

Putting It All Together

Specifying the Exceptions Thrown by a Method

How to Throw Exceptions

The throw Statement

Throwable Class and Its Subclasses

Error Class

Exception Class

Chained Exceptions

Creating Exception Classes

Unchecked Exceptions: The Controversy

Advantages of Exceptions

Advantage 1: Separating Error-Handling Code from "Regular" Code

Advantage 2: Propagating Errors Up the Call Stack

Advantage 3: Grouping and Differentiating Error Types

Summary

Questions and Exercises: Exceptions

Questions

Exercises

Answers

Chapter 11 Basic I/O and NIO.2

I/O Streams

Byte Streams

Character Streams

Buffered Streams

Scanning and Formatting

Table of Contents

I/O from the Command Line

Data Streams

Object Streams

File I/O (Featuring NIO.2)

What Is a Path? (And Other File System Facts)

The Path Class

File Operations

Checking a File or Directory

Deleting a File or Directory

Copying a File or Directory

Moving a File or Directory

Managing Metadata (File and File Store Attributes)

Reading, Writing, and Creating Files

Random Access Files

Creating and Reading Directories

Links, Symbolic or Otherwise

Walking the File Tree

Finding Files

Watching a Directory for Changes

Other Useful Methods

Legacy File I/O Code

Summary

Questions and Exercises: Basic I/O

Questions

Exercises

Answers

Chapter 12 Collections

Introduction to Collections

What Is a Collections Framework?

Table of Contents

- Benefits of the Java Collections Framework

Interfaces

- The Collection Interface
- Traversing Collections
- Collection Interface Bulk Operations
- Collection Interface Array Operations
- The Set Interface
- The List Interface
- The Queue Interface
- The Deque Interface
- The Map Interface
- Object Ordering
- The SortedSet Interface
- The SortedMap Interface
- Summary of Interfaces
- Questions and Exercises: Interfaces

Aggregate Operations

- Pipelines and Streams
- Differences between Aggregate Operations and Iterators
- Reduction
- Parallelism
- Side Effects
- Questions and Exercises: Aggregate Operations

Implementations

- Set Implementations
- List Implementations
- Map Implementations
- Queue Implementations
- Deque Implementations

Table of Contents

    - Wrapper Implementations
    - Convenience Implementations
    - Summary of Implementations
    - Questions and Exercises: Implementations
  - Algorithms
    - Sorting
    - Shuffling
    - Routine Data Manipulation
    - Searching
    - Composition
    - Finding Extreme Values
  - Custom Collection Implementations
    - Reasons to Write an Implementation
    - How to Write a Custom Implementation
  - Interoperability
    - Compatibility
    - API Design
- Chapter 13 Concurrency
  - Processes and Threads
    - Processes
    - Threads
  - Thread Objects
    - Defining and Starting a Thread
    - Pausing Execution with Sleep
    - Interrupts
    - Joins
    - The SimpleThreads Example
  - Synchronization
    - Thread Interference

Table of Contents

Memory Consistency Errors

Synchronized Methods

Intrinsic Locks and Synchronization

Atomic Access

Liveness

Deadlock

Starvation and Livelock

Guarded Blocks

Immutable Objects

A Synchronized Class Example

A Strategy for Defining Immutable Objects

High-Level Concurrency Objects

Lock Objects

Executors

Concurrent Collections

Atomic Variables

Concurrent Random Numbers

Questions and Exercises: Concurrency

Question

Exercises

Answers

Chapter 14 Regular Expressions

Introduction

What Are Regular Expressions?

How Are Regular Expressions Represented in This Package?

Test Harness

String Literals

Metacharacters

Table of Contents

Character Classes

- Simple Classes

Predefined Character Classes

Quantifiers

- Zero-Length Matches
- Capturing Groups and Character Classes with Quantifiers
- Differences among Greedy, Reluctant, and Possessive Quantifiers

Capturing Groups

- Numbering
- Backreferences

Boundary Matchers

Methods of the Pattern Class

- Creating a Pattern with Flags
- Embedded Flag Expressions
- Using the matches(String,CharSequence) Method
- Using the split(String) Method
- Other Utility Methods
- Pattern Method Equivalents in java.lang.String

Methods of the Matcher Class

- Index Methods
- Study Methods
- Replacement Methods
- Using the start and end Methods
- Using the matches and lookingAt Methods
- Using replaceFirst(String) and replaceAll(String)
- Using appendReplacement(StringBuffer,String) and appendTail(StringBuffer)
- Matcher Method Equivalents in java.lang.String

Methods of the PatternSyntaxException Class

Table of Contents

Unicode Support

Matching a Specific Code Point

Unicode Character Properties

Questions and Exercises: Regular Expressions

Questions

Exercise

Answers

Chapter 15 The Platform Environment

Configuration Utilities

Properties

Command-Line Arguments

Environment Variables

Other Configuration Utilities

System Utilities

Command-Line I/O Objects

System Properties

The Security Manager

Miscellaneous Methods in System

PATH and CLASSPATH

Update the PATH Environment Variable (Microsoft Windows)

Update the PATH Variable (Solaris, Linux, and OS X)

Checking the CLASSPATH Variable (All Platforms)

Questions and Exercises: The Platform Environment

Question

Exercise

Answers

Chapter 16 Packaging Programs in JAR Files

Using JAR Files: The Basics

Table of Contents

    - Creating a JAR File
    - Viewing the Contents of a JAR File
    - Extracting the Contents of a JAR File
    - Updating a JAR File
    - Running JAR-Packaged Software
  - Working with Manifest Files: The Basics
    - Understanding the Default Manifest
    - Modifying a Manifest File
    - Setting an Application's Entry Point
    - Adding Classes to the JAR File's Class Path
    - Setting Package Version Information
    - Sealing Packages within a JAR File
    - Enhancing Security with Manifest Attributes
  - Signing and Verifying JAR Files
    - Understanding Signing and Verification
    - Signing JAR Files
    - Verifying Signed JAR Files
  - Using JAR- Related APIs
    - An Example: The JarRunner Application
    - The JarClassLoader Class
    - The JarRunner Class
  - Questions and Exercises: Packaging Programs in JAR Files
    - Questions
    - Answers
- Chapter 17 Java Web Start
  - Additional References
  - Developing a Java Web Start Application
    - Creating the Top JPanel Class
    - Creating the Application

Table of Contents

Benefits of Separating Core Functionality from the Final Deployment Mechanism

Retrieving Resources

Deploying a Java Web Start Application

Setting Up a Web Server

Displaying a Customized Loading Progress Indicator

Developing a Customized Loading Progress Indicator

Specifying a Customized Loading Progress Indicator for a Java Web Start Application

Running a Java Web Start Application

Running a Java Web Start Application from a Browser

Running a Java Web Start Application from the Java Cache Viewer

Running a Java Web Start Application from the Desktop

Java Web Start and Security

Dynamic Downloading of HTTPS Certificates

Common Java Web Start Problems

"My Browser Shows the Java Network Launch Protocol (JNLP) File for My Application as Plain Text"

"When I Try to Launch My JNLP File, I Get the Following Error"

Questions and Exercises: Java Web Start

Questions

Exercises

Answers

Chapter 18 Applets

Getting Started with Applets

Defining an Applet Subclass

Methods for Milestones

Life Cycle of an Applet

Applet's Execution Environment

Table of Contents

Developing an Applet

Deploying an Applet

Doing More with Applets

Finding and Loading Data Files

Defining and Using Applet Parameters

Displaying Short Status Strings

Displaying Documents in the Browser

Invoking JavaScript Code from an Applet

Invoking Applet Methods from JavaScript Code

Handling Initialization Status with Event Handlers

Manipulating DOM of Applet's Web Page

Displaying a Customized Loading Progress Indicator

Writing Diagnostics to Standard Output and Error Streams

Developing Draggable Applets

Communicating with Other Applets

Working with a Server-Side Application

What Applets Can and Cannot Do

Solving Common Applet Problems

"My Applet Does Not Display"

"The Java Console Log Displays java.lang.ClassNotFoundException"

"I Was Able to Build the Code Once, but Now the Build Fails Even Though There Are No Compilation Errors"

"When I Try to Load a Web Page That Has an Applet, My Browser Redirects Me to www.java.com without Any Warning"

"I Fixed Some Bugs and Rebuilt My Applet's Source Code. When I Reload the Applet's Web Page, My Fixes Are Not Showing Up"

Questions and Exercises: Applets

Questions

Exercises

Answers

Table of Contents

Chapter 19 Doing More with Java Rich Internet Applications

- Setting Trusted Arguments and Secure Properties
  - System Properties
- JNLP API
  - Accessing the Client Using the JNLP API
- Cookies
  - Types of Cookies
  - Cookie Support in RIAs
  - Accessing Cookies
- Customizing the Loading Experience
- Security in Rich Internet Applications
- Guidelines for Securing RIAs
  - Follow Secure Coding Guidelines
  - Test with the Latest Version of the JRE
  - Include Manifest Attributes
  - Use a Signed JNLP File
  - Sign and Time Stamp JAR Files
  - Use the HTTPS Protocol
  - Avoid Local RIAs
- Questions and Exercises: Doing More with Rich Internet Applications
  - Questions
  - Exercise
  - Answers

Chapter 20 Deployment in Depth

- User Acceptance of RIAs
- Deployment Toolkit
  - Location of Deployment Toolkit Script
  - Deploying an Applet

Table of Contents

        - Deploying a Java Web Start Application
        - Checking the Client JRE Software Version
    - Java Network Launch Protocol
        - Structure of the JNLP File
    - Deployment Best Practices
        - Reducing the Download Time
        - Avoiding Unnecessary Update Checks
        - Ensuring the Presence of the JRE Software
    - Questions and Exercises: Deployment in Depth
        - Questions
        - Exercise
        - Answers
- Chapter 21 Date- Time
    - Date-Time Overview
    - Date-Time Design Principles
        - Clear
        - Fluent
        - Immutable
        - Extensible
    - The Date-Time Packages
    - Method Naming Conventions
    - Standard Calendar
    - Overview
    - DayOfWeek and Month Enums
        - DayOfWeek
        - Month
    - Date Classes
        - LocalDate

Table of Contents

- YearMonth
- MonthDay
- Year

Date and Time Classes

- LocalTime
- LocalDateTime

Time Zone and Offset Classes

- ZoneId and ZoneOffset
- The Date-Time Classes

Instant Class

Parsing and Formatting

- Parsing
- Formatting

The Temporal Package

- Temporal and TemporalAccessor
- ChronoField and IsoFields
- ChronoUnit
- Temporal Adjuster
- Temporal Query

Period and Duration

- Duration
- ChronoUnit
- Period

Clock

Non-ISO Date Conversion

- Converting to a Non-ISO-Based Date
- Converting to an ISO-Based Date

Legacy Date-Time Code

Table of Contents

Interoperability with Legacy Code
Mapping java.util Date and Time Functionality to java.time
Date and Time Formatting
Summary
Questions and Exercises: Date- Time
Questions
Exercises
Answers
Chapter 22 Introduction to JavaFX
Appendix: Preparation for Java Programming Language Certification
Programmer Level I: Exam
Section 1: Java Basics
Section 2: Working with Java Data Types
Section 3: Using Operators and Decision Constructs
Section 4: Creating and Using Arrays
Section 5: Using Loop Constructs
Section 6: Working with Methods and Encapsulation
Section 7: Working with Inheritance
Section 8: Handling Exceptions
Section 9: Working with Selected Classes from the Java API
Programmer Level II: Exam
Java SE 8 Upgrade Exam
Section 1: Lambda Expressions
Section 2: Using Built-In Lambda Types
Section 3: Filtering Collections with Lambdas
Section 4: Collection Operations with Lambda
Section 5: Parallel Streams
Section 6: Lambda Cookbook

Pearson

Table of Contents

Section 7: Method Enhancements

Section 8: Use Java SE 8 Date/Time API

Section 9: JavaScript on Java with Nashorn

Index

Pearson