

Java SE 11: Programming Complete

1 Introduction to Java

Course Goals

Audience

Course Schedule

Course Practices

Lesson Objectives

What Is Java?

How Java Works?

Classes

Objects

Inheritance

Java APIs

Java Keywords, Reserved Words, and a Special Identifier

Java Naming Conventions

Java Basic Syntax Rules

Define Java Class

Access Classes Across Packages

Use Access Modifiers

Create Main Application Class

Compile Java Program

Execute Java Program

Comments and Documentation

Summary

Practices

2 Primitive Types, Operators, and Flow Control Statements

Objectives

Declare and Initialize Primitive Variables

Restrictions on Primitive Declarations and Initializations

Java Operators

Assignment and Arithmetic Operators

Arithmetic Operations and Type Casting

More Mathematical Operations

Binary Number Representation

Bitwise Operators

Equality, Relational, and Conditional Operators

Short-Circuit Evaluation

Flow Control Using if/else Construct
Ternary Operator
Flow Control Using switch Construct
JShell
Summary
Practices

3 Text, Date, Time, and Numeric Objects

Objectives
String Initialization
String Operations
String Indexing
StringBuilder: Introduction
Wrapper Classes for Primitives
Representing Numbers Using BigDecimal Class
Method Chaining
Local Date and Time
More Local Date and Time Operations
Instants, Durations, and Periods
Zoned Date and Time
Represent Languages and Countries
Format and Parse Numeric Values
Format and Parse Date and Time Values
Localizable Resources
Format Message Patterns
Formatting and Localization: Example
Summary
Practices

4 Classes and Objects

Objectives
UML: Introduction
Modeling Classes
Modeling Interactions and Activities
Designing Classes
Creating Objects
Define Instance Variables
Define Instance Methods
Object Creation and Access: Example
Local Variables and Recursive Object Reference
Local Variable Type Inference

- Define Constants
- Static Context
- Accessing Static Context
- Combining Static and Final
- Other Static Context Use Cases
- NetBeans IDE: Introduction
- Summary
- Practices

5 Improved Class Design

- Objectives
- Overload Methods
- Variable Number of Arguments
- Define Constructors
- Reuse Constructors
- Access Modifiers Summary
- Define Encapsulation
- Define Immutability
- Constants and Immutability
- Enumerations
- Complex Enumerations
- Java Memory Allocation
- Parameter Passing
- Java Memory Cleanup
- Summary
- Practices

6 Inheritance

- Objectives
- Extend Classes
- Object Class
- Reuse Parent Class Code Through Inheritance
- Instantiating Classes and Accessing Objects
- Rules of Reference Type Casting
- Verify Object Type Before Casting the Reference
- Reference Code Within the Current or Parent Object
- Define Subclass Constructors
- Class and Object Initialization Summary
- Override Methods and Use Polymorphism
- Reuse Parent Class Logic in Overwritten Method
- Define Abstract Classes and Methods

- Define Final Classes and Methods
- Override Object Class Operations: toString
- Override Object Class Operations: equals
- Override Object Class Operations: hashCode
- Compare String Objects
- Factory Method Pattern
- Summary
- Practices

7 Interfaces

- Objectives
- Java Interfaces
- Multiple Inheritance Problem
- Implement Interfaces
- Default, Private, and Static Methods in Interfaces
- Interface Hierarchy
- Interface Is a Type
- Functional Interfaces
- Generics
- Use Generics
- Examples of Java Interfaces: java.lang.Comparable
- Examples of Java Interfaces: java.util.Comparator
- Examples of Java Interfaces: java.lang.Cloneable
- Composition Pattern
- Summary
- Practices

8 Arrays and Loops

- Objectives
- Arrays
- Combined Declaration, Creation, and Initialization of Arrays
- Multidimensional Arrays
- Copying Array Content
- Arrays Class
- Loops
- Processing Arrays by Using Loops
- Complex for Loops
- Embedded Loops
- Break and Continue
- Summary
- Practices

9 Collections

Objectives

Introduction to Java Collection API

Java Collection API Interfaces and Implementation Classes

Create List Object

Manage List Contents

Create Set Object

Manage Set Contents

Create Deque Object

Manage Deque Contents

Create HashMap Object

Manage HashMap Contents

Iterate through Collections

Other Collection Behaviors

Use java.util.Collections Class

Access Collections Concurrently

Prevent Collections Corruption

Legacy Collection Classes

Summary

Practices

10 Nested Classes and Lambda Expressions

Objectives

Types of Nested Classes

Static Nested Classes

Member Inner Classes

Local Inner Classes

Anonymous Inner Classes

Anonymous Inner Classes and Functional Interfaces

Understand Lambda Expressions

Define Lambda Expression Parameters and Body

Use Method References

Default and Static Methods in Functional Interfaces

Use Default and Static Methods of the Comparator Interface

Use Default and Static Methods of the Predicate Interface

Summary

Practices