
Basic Java 11 and OO Programming for Developers New to OO (TT2120-J11)

Overview

This course provides hands-on Java 11 training for developers who have little or no prior working knowledge of object-oriented programming languages such as C, COBOL, and 4GL. You will learn the best practices for writing great object-oriented programs in Java 11, using sound development techniques, new improved features for better performance, and new capabilities for addressing rapid application development. Special emphasis is placed on object oriented concepts and best practices.

Prerequisite Comments

Continued Java & JEE training: Intermediate to Advanced Java, JEE Essentials, Java for Web, Spring / Spring Boot, Microservices / Web Services / REST, RXJava, Tuning, Patterns & More
TDD / Test Driven Development, JUnit / Unit Testing, Agile development training
Secure Java Coding / Java Security and secure application development training
Mobile developer / Android training
Please contact us for recommended next steps tailored to your longer-term education, project, role or development objectives.

Target Audience

This is a basic-level programming course designed for attendees with prior development experience in another language, such as COBOL, 4GL, Mainframe or other non-object oriented languages. This course is not geared for non-developers.

Course Objectives

This "skills-centric" course is about 50% hands-on lab and 50% lecture, designed to train attendees in core OO coding and Java development skills, coupling the most current, effective techniques with the soundest industry practices. Our engaging instructors and mentors are highly experienced practitioners who bring years of current "on-the-job" experience into every classroom.

Working within in a hands-on learning environment, guided by our expert team, attendees will learn to:

Understand what OO programming is and what the advantages of OO are in today's world

Work with objects, classes, and OO implementations

Understand the basic concepts of OO such as encapsulation, inheritance, polymorphism, and abstraction

Understand not only the fundamentals of the Java language, but also its importance, uses, strengths and weaknesses

Understand the basics of the Java language and how it relates to OO programming and the Object Model

Work with the Modular system (Project Jigsaw)

Understand and use classes, inheritance and polymorphism

Understand and use collections, generics, autoboxing, and enumerations

Process large amount of data using Lambda expressions and the Stream API

Abstract, static and private methods in interfaces

Take advantage of the Java tooling that is available with the programming environment being used in the class

Java 11 features covered: Using the Local Variable Type in Lambda expressions; Updates made to the String API

Course Outline

1 - A First Look

The Java Platform
Using the JDK
The Eclipse Paradigm

2 - Getting Started with Java

Writing a Simple Class
Adding Methods to the Class

3 - OO Concepts

Object-Oriented Programming
Inheritance, Abstraction, and Polymorphism

4 - Essential Java Programming

Language Statements
Using Strings
Specializing in a Subclass
Fields and Variables
Using Arrays
Local-Variable Type Inference
Java Packages and Visibility

5 - Object Oriented Development

Inheritance and Polymorphism
Interfaces and Abstract Classes
Introduction to Exception Handling
Exceptions

6 - Java Developer's Toolbox

Utility Classes
Java Date/Time

7 - Advanced Java Programming

Introduction to Generics
Lambda Expressions and Functional Interface

8 - Working with Collections

Collections
Using Collections

9 - Stream API

Streams
Collectors

10 - The Java Module System

Introduction to the Module System

11 - Time Permitting

Formatting Strings
Introduction to Annotations
Java 12 and beyond
