



Oracle Java Programming II 5 Days

Overview

Official Oracle Curriculum

This is a second-level course for programmers learning the Java language. It rounds-out the topics that were taught in the previous course, *Java SE: Programming I*, and provides additional key skills for Java programmers, such as secure coding, modular programming, and database access.

This course uses today's main Java version for its practices (currently Java 11).

Target Audience

- Data Scientist
- Developer
- Implementer

Prerequisites

[Java SE: Programming I](#)

Delegates will learn how to

- Create Java applications that leverage the object-oriented features of the Java language, such as encapsulation, inheritance, and polymorphism
- Execute a Java application from the command line
- Create applications that use the Java Collections framework
- Search and filter collections using Lambda Expressions
- Implement error-handling techniques using exception handling
- Implement input/output (I/O) functionality to read from and write to data and text files
- Manipulate files, directories and file systems using the JDK NIO.2 specification
- Perform multiple operations on database tables, including creating, reading, updating and deleting using both JDBC and JPA technology

- Use Lambda Expression concurrency features
- Apply modular programming practices and services to applications
- Create high-performing multi-threaded applications

Outline

Functional Programming

- Functional Interfaces and Lambda Expressions
- Collections Streams, and Filters
- Built-in Functional Interfaces

Modular Programming

- Introduction to Modular Programming in Java
- Services in a Modular Application
- Migration to a Modular Application

Streams and Parallel Streams

- Concurrency
- Parallel Streams
- Terminal Operations: Collectors
- Creating Custom Streams

Java API Programming and Secure Coding Concepts

- I/O (Fundamentals and NIO2)
- Secure Coding
- Database Applications with JDBC
- Localization

Collections and Generics

- What is the Collections Framework?
- Iterators
- Type-Wrapper Classes
- What are Generics?

Exception Handling and Assertions

- Exceptions
- Error handling
- Assertions