incusdata (pty) ltd. 1995/04149/07
Coding · Matters
PO Box 94
Raslouw, 0109
tel : +27 12 666 2020
fax: +27 12 666 2021
www.incusdata.com

Price: R4,500.00 excl. VAT
Duration: 2 days
Code: JFUNC

Java Functional Programming with Lambdas & Streams

Description

Functional programming is a way of developing software, using side-effect-free functions as basic building blocks. Functional code has some advantages: it can be more concise, more predictable, and easier to test than imperative or object-oriented code. Lambdas, which are anonymous functions, are an integral part of functional programming and were introduced in Java 8. This workshop will help Java programmers understand the new terminology and core concepts of functional programming, in particular lambdas and streams. It will teach you how to write even better Java code.

Objectives

After you have completed the Java Functional Programming workshop, you will be able to:

- Understand the terminology and core concepts of functional programming.
- Understand how lambdas are implemented and used in Java from version 8.
- Read, write and maintain Java code that uses lambdas.
- Read, write and maintain Java code that uses the Stream APIs.

Intended Audience

You should attend the Java Functional Programming course if:

- You are a Java programmer and you want to learn about the functional programming in Java.
- You are a Java programmer and you need to write or maintain Java code that uses lambdas.
- You are a Java programmer and you need to write or maintain Java code that uses the Stream API.

Prerequisites

Before you attend the Java Functional Programming course:

- You must have attended our Java Programming course or already be comfortable with the fundamentals of the Java programming language.
- You should have at least 6 months practical experience programming in Java.

Course Contents

Functional Programming Concepts

- Origins of functional programming and the lambda calculas.
- Functional programming vs object-oriented programming.
- Functions as first class citizens.
- Advantages and disadvantages of functional programming.
- Pure functions, higher-level functions, anonymous functions.
- Shared state, side effects and immutability.

Lambda Expressions

- Definition of a lambda expression.
- Use cases for lambdas.
- Lambda syntax.
- Target typing and inference.
- Closures and effectively final local variables.

PO Box 94
Raslouw, 0109
tel : +27 12 666 2020
fax: +27 12 666 2021
www.incusdata.com

- Higher order functions in Java.

Functional Interfaces

- Single Abstract Method (SAM) interfaces and the @FunctionalInterface annotation.
- The java.util.function package.
- Choosing a functional interface.
- Functional interfaces as targets for lambda expressions.
- Default and static methods in interfaces.
- Method and constructor references.
- The Optional class as a better alternative to null.

Streams and Collections

- Introduction to streams.
- External vs internal iteration.
- Concurrency vs parallelism.
- Using lambdas in collections.
- Using comparators and filters.
- Common stream operations: collect, flatMap, map, filter, reduce.
- Transformations and reductions.
- Grouping and partitioning.

Other Topics

- Factors influencing parallel performance.
- Designing with lambdas.
- Implementing GoF design patterns using lambdas.
- Best practices in Java.

*** The lecturer reserves the right to modify the contents of the course to suit the needs of the delegates.*