

**Price:** R9,800.00 excl. VAT  
**Duration:** 4.5 days  
**Code:** JAVAP

# Java Programming

## Description

Java is one of the most popular programming languages in the world, and is used by thousands of companies. This course will teach you the fundamentals of the Java language, so that you can write Java programs or maintain existing Java code. It is also the foundation for learning other Java technologies, like EJB, JSP, Android and Spring.

## Objectives

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

- Understand basic concepts of object orientation and how they apply to Java.
- Write your own Java programs.
- Read and maintain Java programs.
- Debug Java code to find and correct mistakes.

## Intended Audience

You should attend the Java Programming course if:

- You are a programmer and you want to learn the Java language.
- You need to support existing Java systems.
- You need to understand Java so that you can learn other Java-based technologies.
- You have already learnt some Java - perhaps on your own or at university - but struggle with some concepts or have gaps in your knowledge.

## Prerequisites

Before you attend the Java Programming course:

- You must already be a programmer and have experience in programming.
- If you are a mainframe programmer, please consider first attending our Object-Oriented Analysis and Design Course. It will make your move to Java faster and easier.
- You should know how to use the command line for simple instructions.

## Course Contents

### **Overview**

- Java language history.
- Portability and the Java Virtual Machine.
- The Java development environment and tools.

### **OO Concepts**

- Object-oriented vs structured programming.
- Classes, attributes, behaviours.
- Data encapsulation.
- Inheritance and code reuse.
- Polymorphism.
- Abstract classes and interfaces.

### **Applets and Applications**

- Applets vs applications.

- Creating console-based Java applications and simple applets.

### ***Fundamentals***

- Keywords, identifiers and literals.
- Primitive and object reference data types.
- Variable declarations, initialization and scope.
- Expressions, operators and operator precedence.
- Selection and iteration statements.
- Exception handling.
- Modifiers.

### ***Classes, Interfaces and Packages***

- Creating and using classes.
- Encapsulation - fields and methods.
- Constructors and initializers.
- Polymorphism - overloading and overriding.
- Inheritance - subclasses and superclasses.
- Abstract classes and interfaces.
- Canonical classes and JavaBeans.
- Inner classes.
- Packages.

### ***Strings, Arrays and Collections***

- String methods and concatenation.
- Strings versus StringBuffer.
- Creating, initializing and using arrays.
- Collections API overview.

### ***Threads and Multi-tasking***

- Multi-tasking overview.
- Using and creating threads.
- The Runnable interface.
- Thread attributes, priorities and synchronization.

### ***File Input/Output and JDBC***

- Standard System streams.
- Files, Streams, Readers and Writers.
- The Serializable interface and the transient modifier.
- JDBC Overview.

### ***New Language Features in Java 5 and Java 7***

- Java 5 features - annotations, generics, enums, enhanced for loop, auto-boxing/unboxing, varargs, static imports.
- Java 7 features - binary literals, try-with-resources, type inference, strings in switches.

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