# Java Programming 2 / Java Advanced Developer

## Duration: 5 Days  Course Code: JAVA2

### Overview:

Modern Java code extends well beyond object orientation and the associated principles. This hands-on course, delivered using Java 11, endeavours to provide a grounding in advanced topics including generics, functional programming, concurrency, annotations, reflection and security (among others).

This course assumes you are familiar with the fundamentals of the Java language including object oriented principles and that you can build simple applications. If not you should consider our Java Programming 1 / Java Developer course instead.

Exercises and examples are used throughout the course to give practical hands-on experience with the techniques covered.

### Target Audience:

The Java Programming 2 / Java Advanced Developer course is aimed at existing Java developers who are seeking to build on their knowledge of the fundamentals so as to make full use of the language's advanced features.

### Objectives:

- **After you complete this course you will be able to:**
- Build and/or maintain Java applications that exploit advanced features. It also serves as good preparation for developers seeking to contribute to enterprise level applications built with Java EE/Jakarta EE/Spring or similar.

### Prerequisites:

**Attendees should meet the following prerequisites:**

Delegates attending this course should be familiar with the fundamentals of the Java language including object oriented principles,and should be able to build simple applications. This knowledge can be obtained by attendance on the pre-requisite Java Programming 1 / Java Developer course.

### Follow-on-Courses:

**The following courses are recommended for further study:**

- Developing Applications with Java EE
- Unit Testing using JUnit
- Core Spring
- Java Web Services

## Content:

Day 1

Course Introduction

- Administration and Course Materials
- Course Structure and Agenda
- Delegate and Trainer Introductions

Session 1: Generics

- Generic types
- Generic methods
- Bounded type parameters
- Generics and inheritance
- Type inference
- Wildcards
- Type erasure
- Restrictions

Session 2: Collections

- The Collection interface hierarchy
- Iterable and Iterator
- Collection operations and traversal
- The Collections class
- Collection types (characteristics and implementations) incl. List,Set and Queue
- The Map interface hierarchy
- Map operations and traversal
- Map types (characteristics and implementations)

Session 3: hashCode,equals ; compareTo/compare

- The Object class
- Overriding toString
- Equality and membership testing
- Hashing collections
- Overriding hashCode and equals
- The Comparable and Comparator interfaces
- Searching and sorting
- Overriding compareTo and compare

Day 2

Session 4: Nested Classes

- Static nested classes
- Inner classes
- Shadowing
- Local classes
- Anonymous classes
- Variable capture and effectively final

Session 5: Lambda Expressions ; Method References

- What is a lambda expression?
- Lambda use case: collection filter
- Lambda syntax

Session 6: Functional Programming

- What is functional programming?
- Functions as first class objects
- Pure functions
- Higher order functions
- Immutable objects
- Java's functional interfaces
- Functional composition

Session 7: Streams

- What is a stream?
- Obtaining/creating a stream
- Intermediate (non-terminal) operations
- Terminal operations
- The Optional class
- The Collectors class
- Stream concatenation
- Primitive streams

Day 3

Session 8: Exceptions, Assertions and Localisations

- Exceptions (a review)
- Custom exceptions
- Try with resources
- Assertions
- Dates and times
- Internationalisation and localisation
- Resource bundles

Session 9: Module Applications

- Modules (a review)
- Module types
- Analysing dependencies
- Migrating an existing application
- Creating a service

Session 10: Concurrency

- Low-level threading incl. Runnable,sleep,join
- The Java memory model
- Threading problems incl. race conditions,deadlock,and starvation
- Immutable objects and ThreadLocal
- Thread synchronization incl. volatile,synchronized,atomic data types,and locks

Day 4

Session 10 (Continued): Concurrency

- ExecutorService,thread pools,Callable,and Future
- Concurrent and immutable collections

Session 11: I/O

- Files and directories
- I/O streams
- Common operations
- Binary data,character data,and serialisation
- Console I/O

Session 12: NIO.2

- What is NIO?
- Paths
- Manipulating the filesystem
- File attributes
- Listing,traversing,and searching a directory

Day 5

Session 13: Annotation

- What is an annotation?
- Form and application
- Standard annotations
- Declaring an annotation

Session 14: Reflection

- What is reflection?
- Modules,classes,constructors,fields,and methods
- Private members
- Annotations
- Generics and arrays
- Dynamic proxies
- Dynamic class loading

Session 15: Security

- Securing objects
- Injection and input validation
- Handling confidential information
- Serialising and deserialising objects
- Sensitive objects
- Preventing denial of service attacks

- What is a method reference?
- Method reference use case: collection sort
- Method reference types x 4
- Parallel streams

## Further Information:

For More information, or to book your course, please call us on Head Office 01189 123456 / Northern Office 0113 242 5931

info@globalknowledge.co.uk

www.globalknowledge.com/en-gb/

Global Knowledge, Mulberry Business Park, Fishponds Road, Wokingham Berkshire RG41 2GY UK