

MIGRATING FROM JAVA 8 TO JAVA 11 | JAVA 11 NEW FEATURES & SKILLS

Course Code: 101023

Jumpstart Your Java 11 Skills | Move from Java 8 to Java 11 Exploring New features, Versioning, Jigsaw, JShell, Concurrency, Performance Enhancements and More.

Migrating from Java 8 to Java 11 is a three-day, hands-on fast-track course geared for developers who have prior hands-on experience working with Java 8, who need to quickly get up and running the latest features introduced in Java 11. Throughout the course students learn the best practices for taking advantage of the new Java Module system as well as other new features in this major update to the Java programming language.

The Java 9 update introduced major changes to the core language, including new features such as the Java Module system, the introduction of JShell, as well as several small enhancements to the language as part of the 'Milling Project Coin' project. 'Project Coin' was introduced during the development of Java 7 to introduce small language changes. Milling Project Coin introduced several enhancements that did not make it into Java 7. This course provides a fast-paced, high-level overview of some of the lesser-known language changes that were introduced over the years. Several of these small changes have laid the foundation for the enhancements made in Java 9, 10 and 11.

What You'll Learn

This "skills-centric" course is about 50% hands-on lab and 50% lecture, designed to train attendees in core next-level Java development skills, coupling the most current, effective techniques with the soundest industry practices. Throughout the course students will be led through a series of progressively advanced topics, where each topic consists of lecture, group discussion, comprehensive hands-on lab exercises, and lab review.

Our engaging instructors and mentors are highly experienced practitioners who bring years of current "on-the-job" experience into every classroom. Working in a hands-on learning environment, guided by our expert team, attendees will learn to:

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

- Develop modular applications in Java
- Migrate existing Java applications to the Java 11 platform
- Utilize the tooling that is provided in Java 11 to migrate, monitor and optimize applications
- Use the new JShell tool to quickly test java constructs
- Improve implementations already using Java 8's Stream API by utilizing the methods new in Java 11
- Understand how the implementation of the String class has been updated to decrease the memory footprint
- Use the enhancements made to the Concurrency API, working with CompletableFuture instance of Thread pools
- Specific Java 11 features covered include: The Java Module System (project Jigsaw); JShell; Updated try-with-resources; Performance enhancements since Java 9; Updates to Collection and Stream API; Using the Local Variable Types; Updates made to the String API; The HTTPClient API

Who Needs to Attend

Java Developers, Application Developers, Full Stack Developers.

Prerequisites

This is an intermediate- level Java programming course, designed for experienced Java 8 developers who wish to get up and running with Java 11 immediately. Attendees should have a working knowledge of developing Java 8 applications.

MIGRATING FROM JAVA 8 TO JAVA 11 | JAVA 11 NEW FEATURES & SKILLS

Course Code: 101023

CLASSROOM LIVE

\$2,854 CAD

3 Day

Classroom Live Outline

Session: Introduction

Lesson: Java Versioning

- Introduce the new release cycle of Java versions
- Explain LTS versions

Lesson: Milling Project Coin

- Overview of changes made to the language throughout the years
- Multi-catch
- Using effectively final variables in try-with-resources
- Suppressed Exceptions
- Binary literals
- Reserved underscore (Java 9)
- Type inference in anonymous classes (Java 9)
- @SafeVargs (updates in Java 9)
- Default and static methods in interfaces (Java 8)
- Private methods in interfaces (Java 9)

Session: Survey of Java 9 Updates

Lesson: Survey of Java 9 Changes

- Introduce some of the changes in Java
- Java versioning
- The JDK/JRE file structure
- Deprecation
- The jdeprscan tool
- Multi-Release JAR Files
- HTML 5 compliant Javadoc

Lesson: Collection and Stream Updates

- Factory methods for Immutable Collection types
- The takeWhile and dropWhile methods
- The Stream Iterate and ofNullable methods

Lesson: Java 9 Concurrency Updates

- Brief overview of Concurrency in Java
- Overview of CompletableFuture (Java 8)
- Subclassing the CompletableFuture
- The default Executor
- New Factory methods
- Dealing with time-outs

Lesson: Other New Java Features

- Enhancements on the Optional class
- Improvements made in the Process API
- The Stack-Walking API
- The HTTP2 Client
- The Multi-Resolution API

Session: The Java Module system (Jigsaw)

Lesson: Why JigSaw?

- Problems with Classpath
- Encapsulation and the public access modifier
- Application memory footprint
- Java 8's compact profile
- Using internal JDK APIs

Lesson: Introduction to the Module System

- Introduce Project Jigsaw
- Classpath and Encapsulation
- The JDK internal APIs
- Java 9 Platform modules
- Defining application modules
- Define module dependencies
- Implicit dependencies
- Implied Readability
- Exporting packages

Lesson: The Module Descriptor

- Define module requirements
- Explain qualified exports
- Open modules for reflection
- Use ServiceLoader
- The provides and uses keywords

Lesson: Working With Modules

- Being backwards compatible

- The ModulePath and ClassPath
- Unnamed Modules
- Automatic Modules
- The JLink tool

Session: JShell

Lesson: JShell

- Introduction to JShell
- Running Expressions in JShell
- Importing packages
- Defining methods and types
- Using the JShell editor
- Save and loading state

Session: Survey of Java 10 updates

Lesson: Survey of Java 10 Changes

- Docker Awareness and Support
- Unmodifiable Collections
- Garbage Collection Enhancements
- Application Class Data Sharing
- Ahead-of-Time Compilation

Lesson: Local-Variable Type Inference

- Type inference
- Inferring Types of Local Variables
- The var Reserved Type name
- Benefits of Using var
- Backward Compatibility

Session: Survey of Java 11 updates

Lesson: Survey of Java 11 Changes

- Provide an overview of some of the new features introduced in Java 11
- The Java versioning scheme
- Local-Variable Syntax in Lambdas
- Using Regular expression patterns with Predicate
- The Collection toArray Method
- Unicode 10 Standard
- HTTP Client API
- Launch Single-File Source-Code Programs

Lesson: Using Strings in Java 11

- Working with Strings
- Discuss the definition of whitespace in Java
- Introduce the new strip() methods of the String class
- The isBlank() and repeat() methods introduced in Java 11
- Using the lines() method to construct a Stream instance using a String

Lesson: Java 11: Removed Features and Options

- Provide an overview of tools and APIs removed
- Java EE modules are no longer available in Java 11

Session: HTTP Client API

Lesson: The HTTP Client API

- Making HTTP (Hypertext Transfer Protocol) requests
- Explain Incubator Modules
- HTTP2 Client API
- Introduce WebSockets
- Communicate with WebSocket endpoints

Session: Additional Topics

Lesson: Memory Management

- Understand memory management in Java
- Discuss the various garbage collectors
- The Garbage-First (G1) Garbage Collector
- The No-Op and ZGS Garbage Collectors

Lesson: Performance Optimizations

- Ahead-Of-Time Compilation
- Hotspot Diagnostic commands

MIGRATING FROM JAVA 8 TO JAVA 11 | JAVA 11 NEW FEATURES & SKILLS

Course Code: 101023

VIRTUAL CLASSROOM LIVE

\$2,854 CAD

3 Day

Virtual Classroom Live Outline

Session: Introduction

Lesson: Java Versioning

- Introduce the new release cycle of Java versions
- Explain LTS versions

Lesson: Milling Project Coin

- Overview of changes made to the language throughout the years
- Multi-catch
- Using effectively final variables in try-with-resources
- Suppressed Exceptions
- Binary literals
- Reserved underscore (Java 9)
- Type inference in anonymous classes (Java 9)
- @SafeVargs (updates in Java 9)
- Default and static methods in interfaces (Java 8)
- Private methods in interfaces (Java 9)

Session: Survey of Java 9 Updates

Lesson: Survey of Java 9 Changes

- Introduce some of the changes in Java
- Java versioning
- The JDK/JRE file structure
- Deprecation
- The jdeprscan tool
- Multi-Release JAR Files
- HTML 5 compliant Javadoc

Lesson: Collection and Stream Updates

- Factory methods for Immutable Collection types
- The takeWhile and dropWhile methods
- The Stream Iterate and ofNullable methods

Lesson: Java 9 Concurrency Updates

- Brief overview of Concurrency in Java
- Overview of CompletableFuture (Java 8)
- Subclassing the CompletableFuture
- The default Executor
- New Factory methods
- Dealing with time-outs

Lesson: Other New Java Features

- Enhancements on the Optional class
- Improvements made in the Process API
- The Stack-Walking API
- The HTTP2 Client
- The Multi-Resolution API

Session: The Java Module system (Jigsaw)

Lesson: Why JigSaw?

- Problems with Classpath
- Encapsulation and the public access modifier
- Application memory footprint
- Java 8's compact profile
- Using internal JDK APIs

Lesson: Introduction to the Module System

- Introduce Project Jigsaw
- Classpath and Encapsulation
- The JDK internal APIs
- Java 9 Platform modules
- Defining application modules
- Define module dependencies
- Implicit dependencies
- Implied Readability
- Exporting packages

Lesson: The Module Descriptor

- Define module requirements
- Explain qualified exports
- Open modules for reflection
- Use ServiceLoader
- The provides and uses keywords

Lesson: Working With Modules

- Being backwards compatible

- The ModulePath and ClassPath
- Unnamed Modules
- Automatic Modules
- The JLink tool

Session: JShell

Lesson: JShell

- Introduction to JShell
- Running Expressions in JShell
- Importing packages
- Defining methods and types
- Using the JShell editor
- Save and loading state

Session: Survey of Java 10 updates

Lesson: Survey of Java 10 Changes

- Docker Awareness and Support
- Unmodifiable Collections
- Garbage Collection Enhancements
- Application Class Data Sharing
- Ahead-of-Time Compilation

Lesson: Local-Variable Type Inference

- Type inference
- Inferring Types of Local Variables
- The var Reserved Type name
- Benefits of Using var
- Backward Compatibility

Session: Survey of Java 11 updates

Lesson: Survey of Java 11 Changes

- Provide an overview of some of the new features introduced in Java 11
- The Java versioning scheme
- Local-Variable Syntax in Lambdas
- Using Regular expression patterns with Predicate
- The Collection toArray Method
- Unicode 10 Standard
- HTTP Client API
- Launch Single-File Source-Code Programs

Lesson: Using Strings in Java 11

- Working with Strings
- Discuss the definition of whitespace in Java
- Introduce the new strip() methods of the String class
- The isBlank() and repeat() methods introduced in Java 11
- Using the lines() method to construct a Stream instance using a String

Lesson: Java 11: Removed Features and Options

- Provide an overview of tools and APIs removed
- Java EE modules are no longer available in Java 11

Session: HTTP Client API

Lesson: The HTTP Client API

- Making HTTP (Hypertext Transfer Protocol) requests
- Explain Incubator Modules
- HTTP2 Client API
- Introduce WebSockets
- Communicate with WebSocket endpoints

Session: Additional Topics

Lesson: Memory Management

- Understand memory management in Java
- Discuss the various garbage collectors
- The Garbage-First (G1) Garbage Collector
- The No-Op and ZGS Garbage Collectors

Lesson: Performance Optimizations

- Ahead-Of-Time Compilation
- Hotspot Diagnostic commands



MIGRATING FROM JAVA 8 TO JAVA 11 | JAVA 11 NEW FEATURES & SKILLS

Course Code: 101023

PRIVATE GROUP TRAINING

3 Day

Visit us at www.globalknowledge.com or call us at 1-866-716-6688.

Date created: 4/24/2025 11:22:24 PM

Copyright © 2025 Global Knowledge Training LLC. All Rights Reserved.