



Migrating Java 7 to Java 11 | Java 11 / 13 New Features and Skills

Transition to Java 11 | Explore New features, Versioning, Jigsaw, JShell, Concurrency, Performance Enhancements and More

www.triveratech.com

Course Snapshot

- **Course: Migrating Java 7 to Java 11 | Java 11 New Features and Skills (TT2134)**
- **Duration:** 4 days
- **Skill-Level & Audience :** Introduction to Java 11 new features for experienced Java 7 developers.
- **Hands-on Learning:** This course is approximately **50% hands-on**, combining expert lecture, real-world demonstrations and group discussions with machine-based practical programming labs and exercises. Student machines are required.
- **Delivery Options:** This course is available for **onsite private classroom presentation, live online virtual presentation**, or can be presented in a **flexible blended learning format** for combined onsite and remote attendees. Please also ask about our **Self-Paced / Video / QuickSkills or Mini-Camp Flex Hours / Short Course** options.
- **Public Schedule:** This course has active dates on our live-online open enrollment **Public Schedule**.
- **Customizable:** This course agenda, topics and labs can be further adjusted to target your specific training skills objectives, tools and learning goals. Please inquire for details.

Overview

Migrating from Java 7 to Java 11 is a four-day, hands-on Java transition fast-track course geared for developers who have prior hands-on experience working with Java 7, 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-pace, 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.

Learning Objectives

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:

- What functional programming means and what it brings to the table
- How Lambda expressions and functional interfaces can greatly enhance other aspects of Java
- Explore the new 'features' of the Collection API
- To use the new Stream constructs to work with Collections and Collectors
- How to work with default methods and pass methods as arguments
- How to use the new features to support concurrency in multi-code systems
- To work with the new Date/Time API and other new features
- 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

Audience & Pre-Requisites

This is an intermediate- level Java programming course, designed for experienced Java 7 developers who wish to get up and running with Java 11 immediately. Attendees should have a working knowledge of developing Java 7 applications.

Follow On Courses / Take After: Our core Java training courses provide students with a solid foundation for continued learning based on role, goals, or their areas of specialty. Our object oriented, Java developer learning paths offer a wide variety of follow-on courses such as:

- Continued Java & JEE training: Intermediate to Advanced Java, JEE Essentials, Java for Web, Spring / Spring Boot, Microservices / Web Services / REST, RxJava, Tuning, Patterns & More
- TDD / Test Driven Development, JUnit / Unit Testing, Agile development training
- Secure Java Coding / Java Security and secure application development training
- Mobile developer / Android training
- Please contact us for recommended next steps tailored to your longer-term education, project, role or development objectives.

Enhanced Learning Services: Please also ask about our **Pre-Training Class OnRamp & Prep / Primer** offerings, **Skills Gap Assessment Services, Case Studies, Knowledge Check Quizzes, Skills Immersion Programs & Camps, Collaborative Mentoring Services and Extended Learning Support & Post Training** services.

Course Topics / Agenda

Please note that this list of topics is based on our standard course offering, evolved from typical industry uses and trends. We will work with you to tune this course and level of coverage to target the skills you need most. Course agenda, topics and labs are subject to adjust during live delivery in response to student skill level, interests and participation.

Session: Introduction

Lesson: Java Versioning

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

Lesson: Evolving Interfaces

- Interfaces in Java 8
- Default methods
- Static methods
- Multiple Inheritance?

Session: Introduction to Lambda Expressions

Lesson: Functional Programming

- Functional vs OO Programming
- Anonymous Inner-classes
- Utility Methods
- Lambda Expressions

Lesson: Lambda Expressions and Functional Interface

- Lambda Expression Syntax
- Functional Interfaces
- Type Inference in Java 8
- Method references

Session: Collection Updates

Lesson: Java 8 Collection Updates

- Introduce the ConcurrentHashMap
- Lambda expressions and Collections

Session: Streams

Lesson: Streams

- Processing Collections of data
- The Stream interface
- Reduction and Parallelism
- Filtering collection data
- Sorting Collection data
- Map collection data
- Find elements in Stream
- Numeric Streams

- Create infinite Streams
- Sources for using Streams

Lesson: Collectors

- Creating Collections from a Stream
- Group elements in the Stream
- Multi-level grouping of elements
- Partitioning Streams

Session: Additional Java 8 Enhancements

Lesson: The new Date/Time API

- Introduce the new Date/Time API
- LocalDate, LocalDateTime, etc.
- Formatting Dates
- Working with time zones
- Manipulate date/time values

Lesson: Optional

- Introduce Optional
- Implement Optional attributes
- Lambda expressions and Optional

Lesson: Other new Java 8 features

- StringJoiner
- Repeating and Type Annotations
- Parallel Array Sorting
- Improved type inference
- Method parameter reflection
- Updated command line tooling

Lesson: Java 8 Concurrency Updates

- The common thread pool
- Atomic variables
- LongAdder and LongAccumulator
- CompletableFuture
- Non-blocking asynchronous tasks

Lesson: Nashorn JavaScript Engine (Optional)

- Working with JavaScript and Java
- Accessing Nashorn
- Executing JavaScript from Java
- Executing Java from JavaScript
- Implementing a Java Interface
- Alternatives to Nashorn

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 jdepscan 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

Course Materials: Each student will receive a **Student Guide** with course notes, code samples, software tutorials, step-by-step written lab instructions, diagrams and related reference materials and links (as applicable). Students will also receive the project files (or code, if applicable) and solutions required for the hands-on work.

Hands-on Setup Made Simple! Our dedicated tech team will work with you to **ensure your student machines and learning environment is setup, tested and ready to go** well in advance of the course delivery date, ensuring a smooth start to class and seamless hands-on experience for your students. We offer several flexible student machine setup options including **guided manual set up** for simple installation directly on student machines, or **cloud based / remote hosted lab solutions** where students can log in to a complete separate lab environment minus any installations, or we can supply **complete turn-key, pre-loaded equipment** to bring ready-to-go student machines to your students or in-person facility. Please inquire for details.

For More Information

All courses can be presented **onsite** or **online**, or in a **combined / flex / blended learning format**, tailored to target your specific audience, needs and learning goals. We also offer focused, flexible **short courses, self-paced learning options, recorded sessions** and more. We train beginner to advanced skills in all areas we cover, and offer **New Hire / Cohort Training, Boot Camps, Skills Immersion Programs, Reskilling Programs, Skills Migration & Transition Programs**, and more. We collaborate with you to ensure all courses are truly targeted to meet your specific needs and learning skills, maximizing your valuable training time, as well as your important budget.

Please also visit our extensive **Public Training Schedule** for training for smaller groups or individuals. Please contact us for course details, **Corporate Rates** and **Special Discount Offers**.

For more information about our dedicated training services, collaborative coaching services, courseware licensing and development services, public course schedule, training management services, partner programs, or to see our complete list of course offerings and special offers please visit us at www.triveratech.com, email Info@triveratech.com or call us toll free at **844-475-4559**. Our pricing and services are always satisfaction guaranteed.

TRIVERA TECHNOLOGIES • Collaborative IT Training, Coaching & Skills Development Solutions
www.triveratech.com • toll free +1-844-475-4559 • Info@triveratech.com • Twitter TriveraTech

ONSITE, ONLINE & BLENDED TRAINING SOLUTIONS • PUBLIC / OPEN ENROLLMENT COURSES • COURSEWARE LICENSING & DEVELOPMENT
MENTORING • ASSESSMENTS • LEARNING PLAN DEVELOPMENT • SKILLS IMMERSION PROGRAMS / RESKILLING / NEW HIRE / BOOT CAMPS
PARTNER & RESELLER PROGRAMS • CORPORATE TRAINING MANAGEMENT • VENDOR MANAGEMENT SERVICES

Trivera Technologies is a Woman-Owned Small-Business Firm

TECHNOLOGY
MiniCamps

every**Course**
EXTRAS

quickSkills
FREE COURSES AND WEBINARS

afterCourse
COACHING AND SUPPORT