

Java 8 Concurrency and Reactive Programming using RxJava2

Next-Level Java Skills | Explore Parallel Programming in Java, Reactive Programming, Reactive Events, Backpressure and More

Course Snapshot

www.triveratech.com

- **Course:** TT3132: Java 8 Concurrency and Reactive Programming using RxJava
- **Duration:** 2 days
- **Audience & Skill-Level:** This is an **intermediate level** Java programming course geared for experienced Java 8 developers. This course is also available for Project Reactor.
- **Hands-on Learning:** This course is approximately **50% hands-on**, combining expert lecture, real-world demonstrations and group discussions with machine-based practical 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

Java 8 Concurrency and Reactive Programming using RxJava 2 is a two-day, **fast-paced** coding course geared for **experienced** developers who have prior working knowledge of Java. Throughout the course students learn the best practices for writing non-blocking applications in Java 8 using both the Java Concurrency Framework and the RxJava 2 API. The course provides an in-depth view of the enhancements made to the Java Concurrency API in Java 8, allowing for the development of non-blocking asynchronous processes.

This course also covers the RxJava 2 API, an implementation of the Reactive-Streams specification, allowing for the development of asynchronous and event-based programs by using observable sequences. Please note that this course is also available for Project Reactor.

Learning Objectives

This “skills-centric” course is about **50% hands-on lab and 50% lecture**, designed to train attendees in 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 within in a hands-on learning environment, guided by our expert team, attendees will explore:

- Explore Parallel programming in Java including coverage of multithreading and concurrency
- The asynchronous, non-blocking concurrency API introduced in Java 8
- Reactive Programming in Java using Project Reactor
- Understand event publishers and Subscribers
- Apply operators to Reactive events
- Understand the concept of Backpressure
- Properly handle exceptions in the reactive process

Need different skills or topics? If your team requires different topics or tools, additional skills or custom approach, this course may be further adjusted to accommodate. We offer additional Java, JEE and Java for Web application development, design, testing, services, application security and other related topics that may be blended with this course for a track that best suits your needs. Our team will collaborate with you to understand your needs and will target the course to focus on your specific learning objectives and goals.

Audience & Pre-Requisites

This is an **intermediate** level Java SE (JSE) developer course, designed for **experienced Java 8 developers**. Attendees should have current hands-on experience in developing basic Java 8 applications. This course is NOT for new developers or non-developers.

Take Before: Students should have practical skills equivalent to or should have received training in the following topic(s) as a pre-requisite:

- TT2100-J8: Introduction to Java 8 Programming for OO Experienced Developers

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'll work with you to tune this course and level of coverage to target the skills you need most. Topics, agenda and labs are subject to change, and may adjust during live delivery based on audience skill level, interests and participation.

Session: Multithreading and Concurrency

- **Lesson: Introduction to Multithreading and Concurrency**
- Principles of Multithreading
- The Thread class and Runnable interface
- Explore thread synchronization
- Introduce the Java Concurrency API

Lesson: Concurrent Java

- Thread management using Executors
- The common thread-pool
- Submitting and controlling asynchronous tasks
- Explore the locking API
- [Lab: Working with Concurrent Java](#)

Lesson: Non-blocking asynchronous tasks

- The CompletableFuture
- Define non-blocking processes
- [Lab: CompletableFuture](#)

Session: Introduction to RxJava2

Lesson: Introduction

- Introduce ReactiveX and RxJava 2
- The Observer and Iterator pattern and functional programming
- Introduce the RxJava 2 Building blocks
- Discuss hot and cold observables
- Registering operators
- Subscribing to an Observable
- [Lab: RxJava Introduction](#)

Lesson: Testing Event Sources (Introduction)

- Introduce the TestObserver and TestSubscriber
- Testing Observable implementations
- Introduce some commonly used assertions
- [Lab: Testing the Observable](#)

Lesson: Operators

- Introduce RxJava 2 Operators
- Show the use of RxMarbles

(marble diagrams)

- Explain some commonly used operators
- Callback operators
- [Lab: RxJava Operations](#)

Lesson: Creating the event publisher

- Implementing the event source
- Introduce the different types of event publishers
- Convenience methods for creation of publishers
- Creating an event source using an Emitter
- Pre-defined observables
- The defer operator
- ConnectableObservable
- [Lab: RxJava Observables](#)

Session: Working with RxJava2

Lesson: Subjects

- RxJava subjects
- Enable multicasting using Subjects
- Introduce most commonly used Subject implementations

- [Lab: RxJava Subjects](#)

Lesson: RxJava Schedulers (Multithreading)

- Thread usage of Observable, Operator and Observer
- Using the subscribeOn method
- Introduce the Scheduler implementations
- Using the observeOn method

Lesson: Testing (multi-threaded) Event Sources

- Testing Asynchronous code
- Discuss strategies for testing

event sources

- Waiting for source to complete (await methods)
- Use RxJava extension mechanism to Scheduler instances used during test
- The JUnit 5 @ExtendsWith and @RegisterExtension annotations
- Using the TestScheduler to advance time during test
- [Lab: Testing Asynchronous Event sources](#)

Lesson: Backpressure

- Strategies for dealing with

Backpressure

- Reactive Streams
- Flowable and Processor
- “reactive pull” backpressure
- [Lab: RxJava Backpressure](#)

Lesson: Exception Handling

- Handling errors in onError
- Exception handling strategies
- Using onErrorReturn or onErrorNext operators
- Using the retry operators
- The Global Error Handler
- [Lab: RxJava Exceptions](#)

Course Materials: Each participant will receive a **Student Guide** with course notes, code samples, software tutorials, step-by-step written lab instructions, diagrams and related reference materials and resource links. 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 our ‘easy-access’ cloud-based course environment is accessible, fully-tested and verified as ready to go well in advance of the course start date, ensuring a smooth start to class and effective learning experience for all participants. Please inquire for details and options.

[For More Information](#)

For more information about our dedicated training services, collaborative mentoring services, courseware licensing options, courseware development services, public course schedule, training management services, partner and reseller 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 & Courseware 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

Explore Trivera’s Ways to Learn...

