DevOpsSchool

Course curriculum · November 2026

LEVEL 04 OF 04 · ADVANCED

Java Advanced Training

Tune JVM performance, build reactive and event-driven Java services, secure them with OAuth2 and run them on Kubernetes with full observability.

LIVE ONLINE **Max 8-10 hours**	CORPORATE (DAY-WISE) **5 days · Days 1-5**	SELF-PACED VIDEO **1-year LMS access**
BATCH SIZE **Max 10 learners**	NEXT BATCHES **1st of every month · weekends: first Saturday**	CERTIFICATE **Verifiable certificate of completion**

Foundation	Essential	Intermediate	Advanced
1 day · Included	2 days · Included	3 days · Included	5 days · This curriculum

INSIDE THIS CURRICULUM

What you will achieve	Outcomes and level focus
Who should attend	Audience and prerequisites
Agenda	53 topics · Foundation to Advanced
Hands-on labs	4 labs
Formats and fees	Live, 1:1, self-paced, corporate
Trainer and reviews	Rajesh Kumar
Enrolment	Batches, payment, contacts

WHY DEVOPSSCHOOL

- Rated 4.6 on Google
- 1,500+ verifiable certificates issued in the public registry
- Trainer Rajesh Kumar: 20 years in industry
- Small batches: max 10 learners, every session recorded

Course page: www.devopsschool.com/courses/java/java-advance-training.html
Talk to us: WhatsApp +91 99057 40781 · +1 (469) 756-6329 · contact@DevOpsSchool.com · reply within 1 hour

OUTCOMES

What you will achieve

By the end of the Java course path, you will be able to:

1

Write solid Java code

Use core and modern Java features, collections and OOP to write clean, tested programs.

2

Build Spring Boot services

Create database-backed REST services with Spring Boot, tests and containers.

3

Run Java in production

Tune, secure, observe and scale Java services on the JVM and Kubernetes.

WHAT THE ADVANCED LEVEL ADDS

JVM tuning, reactive and event-driven systems, security

Hands-on: Capstone: event-driven Java microservices. Connect two Spring Boot services through Kafka with OAuth2 security and Resilience4j, profile and tune them, and deploy them to Kubernetes with tracing.

WHO SHOULD ATTEND

- Experienced Java practitioners responsible for quality, security and scale.
- Senior engineers, architects and leads who design and troubleshoot Java solutions.
- Teams making Java an organisation-wide standard.

BEFORE YOU START

- The Intermediate level, or equivalent hands-on experience.
- You can already: Build a tested Spring Boot REST service backed by PostgreSQL, use concurrency safely and package the service as a container.
- Basic computer skills and a laptop with a stable internet connection for the live sessions and labs.
- This curriculum includes every earlier level, so it is complete on its own.

AGENDA · FOUNDATION TO ADVANCED

Java Advanced Training agenda

53 topics. Each level builds on the one before it, so this curriculum lists every earlier level in full. Days are for corporate (day-wise) delivery; live online batches cover the same agenda in max 8-10 hours.

DAY 1 — Foundation: Java basics, OOP, collections, files and sockets

Included from Foundation

1. Introduction to Java: the language, JDK, JRE and JVM, LTS releases and setting up the development environment
2. First Java program: writing, compiling with javac and running with java, the main method and an IDE project
3. Syntax and basic data types: primitives, String, literals, variables and operators
4. Control flow and decision making: if, if-else and switch conditional statements
5. Looping statements: for, while and do-while loops with break and continue
6. Arrays and methods: array manipulation, introduction to methods and functions, parameters and return values
7. OOP basics: principles of OOP, classes and objects, constructors and methods
8. Encapsulation, inheritance and polymorphism: data hiding, extends, overriding and dynamic dispatch
9. Classes and packages: creating and using objects, public, private and protected access, packages and imports
10. Exception handling and API docs: try, catch, finally, checked versus unchecked, reading the Java API documentation
11. Collections framework: List, Set and Map interfaces with ArrayList, LinkedList, HashSet and HashMap
12. Collection operations and generics: iterating, common operations and an introduction to generics
13. File I/O: reading and writing files, handling I/O exceptions, input and output streams and serialization
14. Networking basics and practice: networking concepts, socket programming, coding challenges and mini-projects

DAY 2 — Essential: Deeper OOP, modern language features and tooling

Included from Essential

15. Interfaces and abstract classes: default and static methods, when to choose each and the Comparable contract
16. Nested and anonymous classes: static nested, inner and anonymous classes and when lambdas replace them
17. Generics in depth: bounded types, wildcards with extends and super, generic methods and type erasure
18. More collections: Deque, TreeMap, LinkedHashMap, Comparator chains and unmodifiable collections
19. Lambdas and functional interfaces: Function, Predicate, Supplier, Consumer and method references
20. Streams: filter, map, flatMap, reduce, collectors such as groupingBy and partitioningBy
21. Optional and null safety: creating, chaining and avoiding common misuse
22. Records, enums and sealed types: immutable data carriers, enum methods and closed hierarchies
23. Modern syntax: switch expressions, pattern matching for instanceof and switch, text blocks and var
24. Strings and dates: StringBuilder, String.format, regular expressions and the java.time API
25. Gradle builds: project structure, dependencies, tasks and the Gradle wrapper
26. Testing with JUnit 5 and AssertJ: assertions, parameterised tests and Mockito stubs
27. IDE debugging and Git: breakpoints, watches, conditional breakpoints and a Git branch workflow

DAY 3 — Intermediate: Concurrency, persistence, Spring Boot APIs and containers

Included from Intermediate

28. Multithreading: threads, ExecutorService, Future, synchronisation and the Java memory model basics
29. Virtual threads: running blocking code at scale with Java 21 virtual threads and avoiding pinning
30. Concurrent utilities: ConcurrentHashMap, atomics, locks, CompletableFuture composition

31. JDBC with PostgreSQL: DataSource, PreparedStatement, transactions and HikariCP pooling
32. JPA and Hibernate: entities, relationships, fetching strategies, N+1 queries and Spring Data repositories
33. Spring Boot REST services: controllers, request validation with Jakarta Validation and error handling with @ControllerAdvice
34. Configuration: application.yml, profiles, @ConfigurationProperties and externalised secrets
35. API documentation: OpenAPI specs generated with springdoc and Swagger UI
36. Database migrations: Flyway or Liquibase versioned scripts in the build
37. Integration testing: @SpringBootTest, MockMvc and Testcontainers with a real PostgreSQL
38. Logging: SLF4J with Logback, MDC correlation IDs and log levels per package
39. Containers: layered JARs, Dockerfile or Jib images and running with Docker Compose
40. Troubleshooting: stack traces, thread dumps with jcmd, connection leaks and slow queries

DAYS 4-5 **Advanced: JVM tuning, reactive and event-driven systems, security** **New at this level**

41. JVM internals and GC tuning: heap regions, G1 and generational ZGC, GC logs and container-aware memory settings
42. Profiling: JDK Flight Recorder, Mission Control, async-profiler and heap dump analysis with Eclipse MAT
43. Structured concurrency and scoped values: newer concurrency APIs in Java 21 to 25 and when to adopt them
44. Reactive programming: Project Reactor, Spring WebFlux, backpressure and when reactive beats virtual threads
45. Event-driven services: Apache Kafka producers and consumers with Spring for Apache Kafka, idempotency and retries
46. Caching: Spring Cache with Redis or Caffeine, eviction and cache stampede protection
47. Security: Spring Security, OAuth2 and OIDC resource servers, JWT validation and method-level authorisation
48. Resilience: Resilience4j circuit breakers, retries, rate limiters and bulkheads
49. Observability: Micrometer metrics, OpenTelemetry tracing, Actuator health probes and structured logs
50. Kubernetes deployment: health probes, resource limits, graceful shutdown and config via ConfigMaps
51. Startup and footprint: GraalVM native image, CDS and AOT cache options
52. Modules and migration: JPMS basics, jlink runtimes and upgrading Java 8 or 11 code to 21 or 25 with OpenRewrite
53. Capstone: an event-driven, secured, observable microservice pair deployed on Kubernetes

PRACTICE

Hands-on labs

You build these yourself during the course, guided by the trainer.

FOUNDATION

Console app with files and collections

Build a small inventory console application with classes, a HashMap store, exception handling and save and load to a file, then expose it over a simple socket.

ESSENTIAL

Refactor to modern Java

Refactor the inventory app to records, streams and sealed types, build it with Gradle and cover its logic with JUnit 5 tests and Mockito.

INTERMEDIATE

Containerised Spring Boot inventory API

Build an inventory REST API with Spring Boot, JPA and Flyway on PostgreSQL, test it with Testcontainers and run it with Docker Compose.

ADVANCED

Capstone: event-driven Java microservices

Connect two Spring Boot services through Kafka with OAuth2 security and Resilience4j, profile and tune them, and deploy them to Kubernetes with tracing.

HOW YOU CAN LEARN

Formats and fees

The same agenda in four formats. Fees are in Indian rupees as of November 2026; the course page shows them in your currency.

FORMAT	DURATION	WHAT YOU GET	FEE
Live & interactive (online)	Max 8-10 hours	Instructor-led public batch of max 10 learners with demos, labs, Q&A and assignment review. Weekday batches start on the 1st of every month, weekend batches on the first Saturday of every month.	**INR 34,999**
1:1 live training (online)	Max 8-10 hours, one-to-one	The same agenda, taught one-to-one by the trainer. Starts immediately, without waiting for the next batch; sessions planned around your working hours, at your pace.	**INR 99,999**
Self-paced video	1-year LMS access	All four Java levels, plus 20+ more courses and 50+ tools, at your own pace.	**INR 833/month** billed yearly at INR 9,996
Corporate (day-wise)	5 days for this level	A private batch for your team, on demand. Foundation 1 day, Essential 2, Intermediate 3, Advanced 5 (cumulative).	**Quote on request**

ASSESSMENT AND CERTIFICATE

Your progress is checked through hands-on labs and assignments at every level. On completion you receive a DevOpsSchool certificate of completion with a unique credential ID that anyone can verify at devopsschool.com/certificates.

INCLUDED WITH EVERY LIVE BATCH

- Every live session is recorded and stays in your LMS for a year.
- Can't attend your batch? Move to next month's batch.
- Free demo class before you pay.
- Free 10-15 minute call with an advisor or the trainer to check the course fits you.

YOUR TRAINER

Learn from Rajesh Kumar

Principal Architect, trainer and mentor in DevOps, SRE, DevSecOps and platform engineering, with 20 years across 8+ software MNCs, including Staff DevOps/SRE Engineer for a leading Japanese mobile-payment platform and Principal DevOps Architect and Manager in Europe.

Profile: www.rajeshkumar.xyz · Videos: youtube.com/@TheDevOpsSchool

20 years in industry	**200+** corporate trainings
10,000+ engineers mentored	**70+** organisations coached

REVIEWS

What learners say

Rated 4.6 on Google · more reviews at devopsschool.com/testimonials and in our video testimonials.

> “Rajesh is very good trainer. Rajesh was able to resolve our queries and question effectively. We really liked the hands-on examples covered during this training program.”
>
> Indrayani, India

> “The training was very useful and interactive. Rajesh helped develop the confidence of all.”
>
> Abhinav Gupta, Pune

START YOUR JAVA TRAINING

Join the next batch

Weekday batches start on the 1st of every month and weekend batches on the first Saturday of every month, with max 10 learners each. 1:1 training can start immediately; corporate batches run on demand. We reply within 1 hour.

Pay by UPI, credit and debit cards, net banking or PayPal. Before you pay, read our refund policy. All levels of this course: www.devopsschool.com/courses/java/

ENROL OR RESERVE A SEAT
Java Advanced Training

WHATSAPP · INDIA
+91 99057 40781

WHATSAPP · USA
+1 (469) 756-6329

EMAIL
contact@DevOpsSchool.com