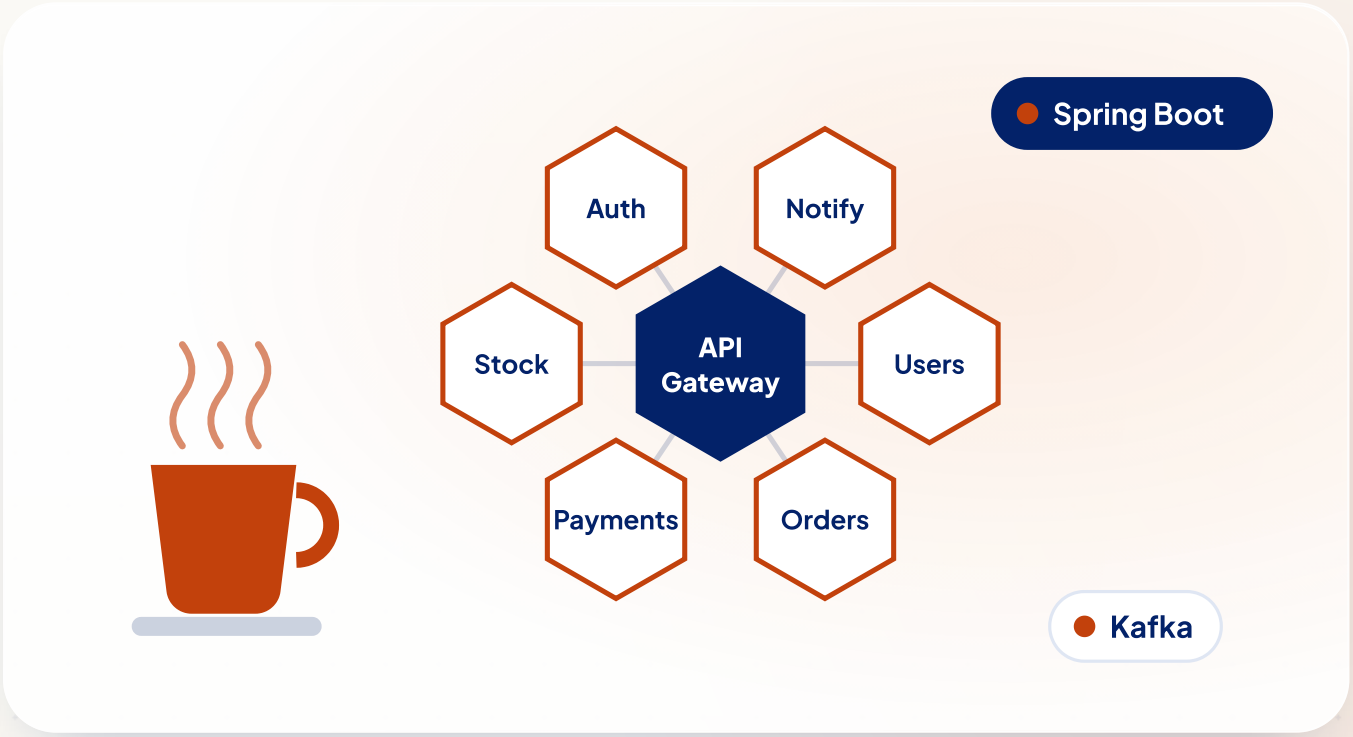
TECH DRONA
DEBUG YOUR CAREER. DEPLOY YOUR DREAMS.

Coming soon · 2026

FULL-STACK DEVELOPMENT COURSE

Java Full Stack Development with Spring Boot and Microservices

From core Java to secure Spring Boot microservices, deployed on AWS.



TECHNOLOGIES COVERED

Java 17+ · Spring Boot · Spring Cloud · Hibernate · JPA · MySQL · Redis · Kafka · AWS

56 hours	11	Labs + capstone
Duration	Modules	Projects

01 ABOUT THE COURSE

Course overview

A 56-hour course that takes learners from core Java to a set of Spring Boot microservices running on AWS. Most of the time goes to the back end: Spring Boot REST APIs, JPA, security, testing, microservices, Redis caching, Kafka messaging and calling AI models with Spring AI. A focused front-end module covers what a Java developer needs to build and connect a working UI.

DURATION
56 hours
Training hours, including guided labs

MODULES
11
Each module ends with hands-on practice

PROJECTS
Labs + capstone
Built, reviewed and demonstrated

LEVEL
Beginner to intermediate

What you will learn

Core and advanced Java
OOP · Collections · Streams · Exceptions · JDBC

Spring Boot
REST APIs · Spring Data JPA · Security · JWT · JUnit · Mockito

Microservices
Eureka · Gateway · Config Server · Resilience4j · Redis · Kafka

Cloud and AI
AWS S3 · EC2 deployment · Spring AI

02 WHO IT IS FOR

Goals & outcomes

Who it is for

CS/IT graduates, final-year students and working developers who want Java back-end or full-stack roles.

Prerequisites

- Programming basics in any language: variables, loops and functions
- Basic SQL helps, but it is covered in the course

Learning objectives

- Write idiomatic modern Java with OOP, collections, streams and lambdas.
- Build layered REST services with Spring Boot, Spring Data JPA and validation.
- Secure APIs with Spring Security and JWT, and test them with JUnit and Mockito.
- Split an application into microservices with discovery, a gateway, central config and circuit breakers.
- Add Redis caching, Kafka messaging and an AI feature, then deploy to AWS.

Key highlights

- Back-end depth that service and product companies hire for.
- Covers the full path from a single API to a set of microservices.
- Includes Redis, Kafka and Spring AI, beyond basic CRUD.
- Deployed on AWS with S3 storage and EC2 hosting.

Expected outcomes

Build back-end systems

Design, build and test Spring Boot services with clean layers and secure APIs.

Work with microservices

Connect services with discovery, a gateway, Kafka events and circuit breakers.

Deploy to the cloud

Package and run services on AWS, with files stored in S3.

Interview ready

Present a working multi-service project and explain its design choices.

SKILLS YOU BUILD

Java · Spring Boot · REST APIs · JPA · Spring Security · Microservices · Redis · Kafka · AWS

03

ELEVEN MODULES, FROM JAVA TO MICROSERVICES

Course curriculum

01 MODULE 1 Core Java

6 hrs

1.1 **Java basics**
JDK and IntelliJ setup · data types and operators · control flow · arrays and strings

1.2 **Object-oriented Java**
classes and objects · constructors · encapsulation · inheritance · polymorphism · abstraction and interfaces

1.3 **Modern Java features**
records · enums · var · switch expressions · text blocks

02 MODULE 2 Advanced Java

6 hrs

2.1 **Collections and generics**
List, Set and Map · Comparable and Comparator · generic classes and methods

2.2 **Functional Java**
lambdas · functional interfaces · Streams API · Optional

2.3 **Exceptions and concurrency**
checked and unchecked exceptions · custom exceptions · threads · ExecutorService basics

2.4 **JDBC**
DriverManager and connections · PreparedStatement · ResultSet · CRUD with JDBC

03 MODULE 3 SQL, Hibernate and JPA

5 hrs

3.1 **SQL with MySQL**
DDL, DML and DQL · constraints and keys · joins · aggregate functions · commit and rollback

3.2 **ORM with Hibernate**
entities and IDs · id generation · cascade types · eager vs lazy loading

03

MODULE 3 · CONTINUED

SQL, Hibernate and JPA

3.3 **Relationships**
one-to-one · one-to-many and many-to-one · many-to-many

04

MODULE 4

Spring and Spring Boot

6 hrs

4.1 **Spring core**
beans and the container · IoC and dependency injection · bean scopes and lifecycle · annotation and Java configuration

4.2 **Spring Boot**
auto-configuration · starters · application.properties and YAML · profiles

4.3 **Cross-cutting features**
AOP for logging · Actuator and metrics · scheduling with cron expressions

05

MODULE 5

REST APIs and Spring Data JPA

6 hrs

5.1 **Building REST APIs**
@RestController vs @Controller · HTTP methods · ResponseEntity and status codes · DTOs and validation

5.2 **Spring Data JPA**
JpaRepository CRUD · finder methods · JPQL and custom queries · @Transactional · pagination and sorting

5.3 **Error handling**
@ControllerAdvice and @RestControllerAdvice · exception handlers · consistent error responses

06 MODULE 6 — Security, testing and Git

5 hrs

6.1 **Spring Security**
security filter chain · JWT login · role-based access · password hashing · CORS

6.2 **Unit testing**
JUnit 5 annotations and assertions · Mockito · testing the service layer

6.3 **Git and GitHub**
branching · pull requests · code review · merge conflicts

07 MODULE 7 — Front end for Java developers

3 hrs

7.1 **HTML, CSS and Bootstrap**
page structure · forms · responsive layout · Bootstrap components

7.2 **JavaScript and the API**
DOM and events · fetch and JSON · calling Spring Boot APIs

7.3 **Server-side views**
Thymeleaf templates · forms and validation messages

08 MODULE 8 — Microservices with Spring Cloud

6 hrs

8.1 **Microservices design**
monolith vs microservices · service boundaries · synchronous vs asynchronous communication

8.2 **Service discovery and calls**
Eureka server and clients · RestTemplate and OpenFeign

8.3 **Gateway and configuration**
Spring Cloud Gateway routing · Config Server backed by GitHub

8.4 **Resilience**
Resilience4j circuit breaker · fallback methods · thresholds and retries

09 MODULE 9 — Caching, messaging and AI integration

5 hrs

9.1	**Redis caching**	why cache · Redis setup and CLI · @Cacheable, @CachePut and @CacheEvict · TTL and expiry
9.2	**Apache Kafka**	brokers, topics, partitions and offsets · producers and consumers · KafkaTemplate and listeners · JSON serialisation
9.3	**Spring AI**	calling an LLM from Spring Boot · prompt templates · structured output

10 MODULE 10 — Deployment on AWS

3 hrs

10.1	**AWS essentials**	regions and availability zones · free tier · security groups and ports
10.2	**Amazon S3**	buckets and permissions · upload and download from Spring Boot · pre-signed URLs
10.3	**Going live**	packaging a JAR · running on EC2 · public DNS · front end on Netlify

11 MODULE 11 — Capstone project

5 hrs

11.1	**Design the system**	services and data · API contracts
11.2	**Build and deploy**	two or more services · gateway and security · simple UI · AWS deployment
11.3	**Demo and code review**	live demonstration · review feedback — PROJECT

Total duration · 56 hours

Training hours across eleven modules, including guided labs and the capstone project.

04 LEARN BY BUILDING

Practical learning

Hands-on learning

- Core Java labs: a bank-account simulator and a stream-based sales report.
- An employee-management REST API with Spring Data JPA, validation, error handling and unit tests.
- Securing that API with JWT roles and adding a Thymeleaf admin page.
- Splitting it into user, order and inventory services behind a gateway with a circuit breaker.
- Adding Redis caching, a Kafka order-event pipeline and S3 file uploads.

Tools & technologies

Language and frameworks

Java 17+ · Spring Boot · Spring Cloud · Spring Security · Spring AI

Data

MySQL · Hibernate · Spring Data JPA · Redis

Messaging and cloud

Apache Kafka · AWS S3 · AWS EC2 · Netlify

Workflow

IntelliJ IDEA · Maven · JUnit 5 · Mockito · Postman · Git and GitHub

Projects

Mini project · Online food ordering

Menu, cart and order APIs with Spring Boot and JPA, secured with JWT.

Mini project · AI resume screening

Upload resumes, extract key details with Spring AI and rank candidates against a job description.

Capstone · choose one

A hospital-management, accommodation-booking or banking system built as microservices behind a gateway and deployed on AWS.

TECH DRONA

DEBUG YOUR CAREER. DEPLOY YOUR DREAMS.

One technology partner. From learning to building to scaling.

Software engineering, AI and cloud, practical training and internships, recruitment and technology procurement, from one accountable team in Vijayawada.

NEXT STEP

Coming soon: join the waitlist

techdrona.in/contact

WEBSITE
techdrona.in

EMAIL
info@techdrona.com

LOCATION
Vijayawada, Andhra Pradesh, India

MSME registered

AICTE partner

Course details, schedules and fees are confirmed on enquiry and may change.