Where trainees meet real Support

Java Full-Stack Developer Program

Spring Boot • React • Microservices • Cloud Deployment

8 Weeks Duration • 40 Total Lectures (5 Lectures/Week) • Beginner to Advanced

About the Course

- **Comprehensive Full-Stack Mastery:** Master backend engineering with Java & Spring Boot and modern frontend with React
- **Production Architecture:** Build scalable microservices, secure RESTful APIs, and relational databases with JPA
- **Application-First Pedagogy:** Direct hands-on coding from Day 1 building real-world enterprise web systems
- **Industry Best Practices:** Clean Code, OOP, SOLID principles, JWT authentication, and Docker containerization

Who Can Attend?

- Aspiring Software Engineers & Developers wanting a complete Full-Stack Java career
- Frontend developers seeking strong backend, database & microservices skills
- Backend Java developers looking to master modern React, Hooks & REST integration
- College graduates and IT professionals preparing for enterprise Full-Stack developer interviews

**No previous Java experience required. Starting points are tailored to learner background.*

Core Technology Stack

Backend: Java (OOP, Streams), Spring Boot, Spring MVC, Spring Data JPA, Hibernate, Spring Security (JWT), REST APIs • **Frontend:** React.js, JavaScript (ES6+), HTML5, CSS3, React Router, Axios, Context API • **Database & Tools:** MySQL, PostgreSQL, JDBC, Git, Docker, Postman, Maven

8-Week Execution Roadmap (Part 1: Backend Engineering & REST APIs)

Week 1 – 2: Core Java Fundamentals & OOP Essentials (Lectures 1 – 10)

- **Java Architecture & Syntax:** JVM, JRE, JDK, primitives, variables, operators, conditional branching, and loops
- **Object-Oriented Programming (OOP):** Classes, Objects, Inheritance, Polymorphism, Abstraction, and Encapsulation
- **Interfaces & Abstract Classes:** Contract-based programming, loose coupling, and SOLID design fundamentals
- **Collections Framework:** List (ArrayList, LinkedList), Set (HashSet), Map (HashMap), Queue, Iterator & Generics
- **Java 8+ Features:** Lambda expressions, Functional Interfaces, Streams API for filtering, mapping & reduce
- **Exception Handling & Multithreading:** Try-catch-finally, custom exceptions, Thread lifecycle, and synchronization

Week 3: Database Connectivity with JDBC & Hibernate ORM (Lectures 11 – 15)

- **Relational Database Fundamentals:** MySQL schema design, primary/foreign keys, indexing, and complex SQL joins
- **JDBC Architecture:** DriverManager, Connection, Statement, PreparedStatement, and navigating ResultSet
- **Transaction Management:** Commit, rollback, ACID properties, transaction isolation, and connection pooling
- **Introduction to ORM & JPA:** Object-Relational Mapping concepts, EntityManager, and entity lifecycle states
- **Hibernate Entity Mapping:** @Entity, @Table, @Id, @GeneratedValue, and mapping relationships (@OneToMany, @ManyToOne)

Week 4: Spring Boot Architecture & RESTful API Engineering (Lectures 16 – 20)

- **Spring Framework Core:** Inversion of Control (IoC), Dependency Injection (@Autowired), and Spring ApplicationContext
- **Spring Boot Starters:** Auto-configuration, project bootstrap via Spring Initializr, and application.properties/yml
- **Spring MVC Controller Architecture:** @RestController, @RequestMapping, @GetMapping, @PostMapping, @PutMapping, @DeleteMapping
- **Request Handling & Payloads:** @PathVariable, @RequestParam, @RequestBody, and ResponseEntity with proper HTTP status codes
- **API Validation & Error Handling:** Bean Validation (@Valid, @NotNull), custom exceptions, and @ControllerAdvice handlers

SUPPORT ME — BY THE DEVELOPERS' GROUP

8-Week Execution Roadmap (Part 2: Security, Microservices & Full-Stack React)

Week 5: Spring Data JPA, JWT Security & Authentication (Lectures 21 – 25)

- **Spring Data JPA Repositories:** JpaRepository interface, derived query methods, and custom JPQL / native @Query
- **Database Pagination & Sorting:** Pageable, Sort parameters, and efficient database query optimization
- **Spring Security Architecture:** SecurityFilterChain, AuthenticationManager, UserDetailsService, and password hashing (BCrypt)
- **Stateless JWT Authentication:** Generating, signing, and validating JSON Web Tokens (JWT) for secure REST endpoints
- **Role-Based Access Control (RBAC):** Protecting endpoints using @PreAuthorize, roles, and granular user permissions

Week 6: Microservices Architecture Basics & Cloud Services (Lectures 26 – 30)

- **Microservices vs. Monoliths:** Service decomposition principles, bounded contexts, and independent deployability
- **Service Discovery & Registry:** Configuring Netflix Eureka Server and registering microservices instances
- **Spring Cloud API Gateway:** Routing requests, path predicates, filter chains, and cross-cutting security
- **Inter-Service Communication:** Synchronous communication using OpenFeign Client and resilient error handling

Week 7: Modern Frontend Development with React (Lectures 31 – 35)

- **React Core Architecture:** JSX syntax, component hierarchy, props, state, and unidirectional data flow
- **Essential React Hooks:** useState for local state, useEffect for side-effects and API lifecycle, and custom hooks
- **Single Page Applications:** Client-side routing with React Router v6, dynamic routes, and URL parameters
- **State Management & Forms:** Controlled components, form validation, Context API, and handling complex UI state

Week 8: Full-Stack Integration, Containerization & Deployment (Lectures 36 – 40)

- **Frontend-Backend Integration:** Axios HTTP client, CORS configuration, handling JWT in headers, and protected routes
- **Full-Stack Capstone Defense:** Connecting React UI to Spring Boot APIs with live database persistence
- **Docker Containerization:** Writing Dockerfiles for Spring Boot & React, building images, and multi-container Docker Compose
- **Career Acceleration & Placement:** Technical resume optimization, GitHub portfolio defense, and mock Java interview rounds

Projects You'll Build

- **Production E-Commerce Platform:** Full-stack store with product catalog, cart management, checkout, and JWT role-based security
- **Enterprise Microservices Order System:** Decoupled services communicating via Feign Client with Spring Cloud API Gateway
- **Interactive Task & Project Dashboard:** Responsive React frontend consuming secured Spring Boot REST APIs with live CRUD operations

Learning Outcomes & Career Impact

- **Architect & Build:** Build enterprise-grade backend APIs with Spring Boot and dynamic frontend applications with React
- **End-to-End Proficiency:** Seamlessly integrate database entities, business logic, secure auth, and interactive UI components
- **Interview Readiness:** Confidently clear Full-Stack Java engineering interviews covering OOP, multithreading, Spring, and React
- **Deploy with Confidence:** Containerize full-stack applications with Docker and deploy to production cloud infrastructure

General Trainer Profile & Mentorship

Led by Senior Java Enterprise Architects with 10+ years of software engineering leadership. Benefit from direct mentor-led code reviews, weekly architectural teardowns, production debugging sessions, and personalized interview coaching designed to accelerate your career transition.

SUPPORT ME
— BY THE DEVELOPERS' GROUP