

Spring Boot: JPA, JWT Security & JUnit Testing

Learn to build, secure and test Spring Boot REST APIs with JPA persistence, JWT authentication and JUnit.

10 Days

DURATION

81

TOPICS

32

HANDS-ON ACTIVITIES

COURSE OVERVIEW

This course begins with a recap of Java fundamentals and object-oriented programming before covering collections, exception handling, functional programming and multithreading. It then introduces Git, build tools and JUnit testing, followed by Spring Boot foundations including REST controllers and the service layer. Learners work with Spring Data JPA and database integration, apply validation and centralised exception handling, optimise REST APIs, and implement Spring Security, Actuator monitoring, session management and JWT authentication with role-based authorization and OWASP practices. The course concludes with a guided mini project extended across modules and a capstone project applying these concepts to a full system.

WHO THIS COURSE IS FOR

Java Developers

Backend Developers

API Developers

Software Engineers

QA Engineers

LEARNING OBJECTIVES

- Apply OOP concepts such as inheritance and polymorphism
- Implement collections, exception handling and multithreading in Java
- Write unit tests with JUnit and apply TDD principles
- Configure Maven and Gradle build tools and manage Git repositories
- Build REST APIs with Spring Boot controllers and service layers
- Implement data persistence using Spring Data JPA and Hibernate
- Apply request validation, pagination and caching to REST APIs
- Secure REST APIs using Spring Security, JWT and OWASP practices

81 Topics

DAY 1 Java Recap & Object-Oriented Programming

Refresh fundamentals and strengthen OOP for independent coding.

01

Java Basics Recap

TOPICS

- Variables (primitive vs reference types)
- Operators (arithmetic, relational, logical, bitwise)
- Control Flow (if-else, switch, loops)
- Methods (parameters, return types, overloading)

02

OOP Essentials

TOPICS

- Classes & Objects, Constructors
- Inheritance (extends keyword, method overriding)
- Polymorphism (compile-time vs runtime)
- Abstraction (abstract classes, methods)
- Encapsulation (access modifiers, getters/setters)

03

Advanced OOP

TOPICS

- Interfaces & multiple inheritance
- Inner Classes (static, member, anonymous)
- Composition vs Inheritance (HAS-A vs IS-A relationship)

3 Handon Labs

Build robust and concurrent programs.

01

Collections Framework

TOPICS

List (ArrayList, LinkedList) · Set (HashSet) · Map (HashMap) · Queue (PriorityQueue)

02

Exception Handling

TOPICS

- Checked vs Unchecked exceptions
- try-catch-finally block
- Throwing & creating Custom Exceptions
- Best practices (specific exceptions, avoiding empty catch blocks)

03

Functional Programming

TOPICS

Lambdas: (args) -> expression · Streams API: map, filter, reduce, collect

04

4 Handon Labs

Multithreading Basics

TOPICS

Creating threads: Thread vs Runnable · Thread lifecycle & Synchronization

Learn enterprise development tools for collaboration & testing.

01

Git Essentials

TOPICS

Git init, clone, add, commit, push, pull · Branching & merging · Handling merge conflicts

02

Build Tools

TOPICS

- **Maven:** POM.xml structure, dependencies, lifecycle (compile, test, package, install)
- **Gradle:** build.gradle scripts, dependency management
- Maven vs Gradle comparison

03**4 Handon Labs**

JUnit Testing

TOPICS

- **Lifecycle:** @BeforeEach, @AfterEach, @BeforeAll, @AfterAll
- **Assertions:** assertEquals, assertTrue, assertThrows
- Parameterized tests
- Basics of TDD (red -> green -> refactor cycle)

DAY 4

Spring Boot Foundations

Build REST APIs with Spring Boot.

01

Spring Core Concepts

TOPICS

- Inversion of Control (IoC), Dependency Injection (DI)
- Bean lifecycle, @Component, @Autowired

02

Spring Boot Basics

TOPICS

- Starters (spring-boot-starter-web, spring-boot-starter-data-jpa)
- Auto-configuration
- Project structure & application.properties

03

REST Controllers

TOPICS

@RestController, @RequestMapping · GET, POST, PUT, DELETE endpoints · RequestBody vs PathVariable vs RequestParam

04

Service Layer

TOPICS

Separation of concerns · Business logic in @Service classes

4 Handon Labs**DAY 5**

JPA, Spring Testing & Guided Mini Project

Persist data with JPA and integrate testing.

01

Spring Data JPA

TOPICS

Entities, @Id, @GeneratedValue · Repositories (CrudRepository, JpaRepository) · ORM basics & Hibernate overview · Custom Queries with @Query

02

Database Integration

TOPICS

H2 Database setup (in-memory) · Schema auto-generation

03

Spring Testing

TOPICS

@SpringBootTest for integration testing · Repository & service layer testing

5 Activities**DAY 6**

Validation, Exception Handling & Optimizing REST APIs

Apply validation, centralized exception handling and optimize REST APIs.

TOPICS

- Global exception handling using @RestControllerAdvice
- Custom exception handling
- Request validation using @Valid & Hibernate Validator (e.g., @NotNull, @Email, @Size)
- Pagination & filtering for large datasets (Pageable)
- Caching with Spring Cache

4 Activities

Implement spring security and monitoring with actuator.

TOPICS

4 Activities

- Securing REST APIs with Spring Security
- Enabling Spring Boot Actuator endpoints
- Health checks & metrics (health, info, metrics)
- Securing actuator endpoints

Secure Spring Boot apps using sessions, cookies, JWT, and OWASP.

TOPICS

4 Activities

- Session management & cookies in Spring Boot
- JWT Authentication flow (login -> token -> verify -> access)
- Role-based authorization for APIs
- OWASP Top 10 Security Practices

Apply all concepts to a full-fledged system.

Project Ideas

TOPICS

- Online Course Management System - Manage courses, students, enrolments
- Inventory Management System - Track products, stock, suppliers
- Online Task Manager - CRUD + multithreading for scheduling tasks
- Employee Payroll System - Calculate salaries, maintain payroll history

Requirements

TOPICS

- Spring Boot backend with REST APIs
- JPA persistence with DB
- GitHub repo for collaboration
- JUnit tests for at least 3 critical services
- Request validation & exception handling
- JWT authentication + role-based access
- Apply OWASP Top 10 best practices

Deliverables

TOPICS

Working Spring Boot application · Database persistence with JPA · Unit test coverage · Code hosted on GitHub · Final demo & architecture explanation

HANDS-ON ACTIVITIES

Practise What You Learn

#	MODULE	ACTIVITY
03	Advanced OOP	HANDON LABS
		■ Bank Account System: Implement deposits, withdrawals, and balance check using classes & methods
		■ Employee Class Hierarchy: Create base class Employee and subclasses (Manager, Developer) demonstrating inheritance & polymorphism
04	Multithreading Basics	■ Debugging in IDE: Set breakpoints, inspect variables, step into/step over methods
		HANDON LABS
		■ Library Management System: Store books in a List, search/filter using Streams API
03	JUnit Testing	■ Custom Exception: Create InvalidBookException for invalid data input
		■ Multi-threaded Ticket Booking System: Multiple threads booking seats
		■ Employee Records with Stream API: Filter employees by salary, department, etc.
04	Service Layer	HANDON LABS
		■ Git Repo Setup: Initialize repo, create branches, push to GitHub
		■ Maven Project Creation: Create & run simple HelloWorld app
03	Spring Testing	■ Convert Maven -> Gradle Project: Add Gradle build scripts
		■ JUnit Tests for Services: Write unit tests for Calculator or Employee Service class
		HANDON LABS
04	Service Layer	■ Create Spring Boot Project: Using Spring Initializr
		■ Employee Service: CRUD operations with hardcoded data
		■ Build REST APIs: Expose endpoints for employee CRUD
03	Spring Testing	■ Test with Postman: Call APIs and check responses
		LAB (GUIDED MINI PROJECT)
		■ Employee Management System - Phase 1
04	Service Layer	■ Create Entity: Employee (id, name, department, salary)
		■ CRUD APIs with JPA repository
		■ Unit tests for Employee repository & service
03	Spring Testing	■ Push project to GitHub

#	MODULE	ACTIVITY
	Validation, Exception Handling & Optimizing REST APIs	LAB (ENHANCE MINI PROJECT) ■ Add validation to Employee entity (name not empty, salary > 0) ■ Implement GlobalExceptionHandler with @ControllerAdvice ■ Add pagination API for employee list ■ Add caching for “Get All Employees” API
	Spring Security & Monitoring with Actuator	LAB (ENHANCE MINI PROJECT) ■ Add Spring Security to Employee Management System ■ Secure endpoints with Basic Auth ■ Enable actuator endpoints & check health, metrics ■ Restrict actuator endpoints
	Sessions, JWT & OWASP Essentials	LAB (SECURE MINI PROJECT) ■ Implement JWT Authentication (login -> token generation) ■ Apply role-based access to Employee APIs ■ Add secure headers ■ Apply OWASP best practices

WHAT IS INCLUDED

Everything You Need to Succeed

- Koenig course material for every module
- Delivery by certified instructors and subject matter experts
- Hands-on exercises throughout the course
- Course completion certificate from Koenig

Who We Are

Established in 1993, Koenig Solutions is a globally recognised training provider with 35+ technology partnerships, 5,000+ training programmes and a network of certified instructors and subject matter experts, supporting individuals and enterprise teams worldwide.

FLEXIBLE DELIVERY

Learn the Way That Suits You

Classroom Training

Live Online Classes (ILO)

Fly-Me-A-Trainer (FMAT)

Flexi (Self-Paced)

1-on-1 Training

Ready to Get Started?

Talk to a learning advisor about schedules, group training and customised delivery.

Visit koenig-solutions.com

✉ info@koenig-solutions.com