

Spring Fundamentals

Duration: 2 days

Prerequisites: Knowledge of Java Programming

Day 1 – Spring Core & IOC

Module 1: Introduction to Spring Framework

- What is Spring Framework?
 - Why do we use Spring?
 - Key features of Spring
 - Spring Ecosystem overview (Core, MVC, Data, Boot)
 - Difference between Spring and Spring Boot
-

Module 2: Inversion of Control (IoC) & Dependency Injection

- Understanding IoC
 - What is Dependency Injection (DI)?
 - Types of DI (Constructor, Setter)
 - BeanFactory vs ApplicationContext
 - Bean lifecycle (simple overview)
 - Bean scopes (Singleton, Prototype)
-

Module 3: Spring Core Configuration

- XML configuration basics
 - Annotation-based configuration
 - @Component
 - @Autowired
 - Java-based configuration basics
 - @Configuration, @Bean
 - What is Component Scanning?
-

Module 4: Introduction to Spring AOP

- What is AOP?
 - Why we use AOP (logging, auditing, etc.)
 - Basic AOP terms: Aspect, Pointcut, Advice
 - Simple aspect using @Aspect (Beginner-level example)
-

Day 2 – Spring MVC, Data Access & Spring Boot

Module 5: Introduction to Spring MVC

- What is MVC architecture?
- Role of DispatcherServlet

- Creating a simple Controller
 - Understanding @RequestMapping
 - Simple GET and POST examples
 - Basic introduction to Views (JSP/Thymeleaf overview)
-

Module 6: Spring Data Access Basics

- What is JdbcTemplate?
 - Connecting to a database in Spring
 - Performing simple CRUD operations
 - Introduction to ORM (simple explanation of Hibernate/JPA)
 - Basic idea of @Transactional
-

Module 7: Spring Boot Fundamentals

- What is Spring Boot?
 - Why use Spring Boot?
 - Starter dependencies
 - Auto-Configuration
 - Project structure understanding
 - Creating a simple REST API
 - application.properties introduction
-

Module 8: Exception Handling & Validation Basics

- Introduction to Bean Validation (@Valid)
- Simple input validation example
- Basic error handling concepts
- @ExceptionHandler simple usage