



Spring Boot

Duration 2 day(s) (SPRING-BOOT)

Spring Boot Developer

Official Training



Zenika exclusivity



Description

Building on the benefits of the Spring framework, Spring Boot represents the next chapter in Java development. Spring Boot's opinionated approach removes much of the boilerplate and configuration that characterizes Spring, making development faster and deployment easier. This course offers experience with Spring Boot and its major features, including auto-configuration, data access, Actuator, and more. On completion, participants will have a foundation for creating enterprise-ready applications. Please note that this course is a subset of the material in our 4 day CoreSpring course - there is no need to take both courses. This course is recommended if you have a good working knowledge of Spring Basics (see Prerequisites) but are new to Spring Boot.

Goals

Upon completion of this course, participants will understand how Spring Boot enhances the following:

- Describe the benefits provided by Spring Boot
- Initialize a project using Spring Boot Starters
- Leverage Spring Boot's auto configuration features to enable JDBC setup and initialization
- Create simplified backing-store solutions using Spring Data and JPA
- Build a simple MVC application using Spring Boot, embedded Web Server and fat JARs or classic WARs
- Build a RESTful Web application
- Use Spring Security to secure Web and REST endpoints
- Enable and extend metrics and monitoring capabilities using Spring Boot actuator
- Leverage advanced configuration capabilities
- Utilize Spring Boot enhancements to testing

Public

Application developers who want to increase their understanding of Spring Boot and a focus on fundamentals.

Prerequisites

A good working knowledge web application development using Java and an IDE.

Basic understanding of Spring: Java Config, component-scanning, Spring driven testing and Spring data-management (JdbcTemplate, @Transactional). Experience using Java, Git, and a build tool (Maven or Gradle).

Structure

40% Theory, 60% Practice

REVIEW OF SPRING ESSENTIALS

- Why Spring
- Configuration using Spring
- Bean creation
- Data Management

SPRING BOOT INTRODUCTION

- Introduction to Spring Boot
- Value Proposition of Spring Boot
- High-level Spring Boot features
- Creating a simple Boot application using Spring Initializr web-site

SPRING BOOT DEPENDENCIES, AUTO-CONFIGURATION AND RUNTIME

- Dependency management using Spring Boot starters
- How auto-configuration works
- Configuration properties
- Overriding auto-configuration
- Using CommandLineRunner

JPA WITH SPRING AND SPRING DATA

- Quick introduction to ORM with JPA
- Benefits of using Spring with JPA
- JPA configuration in Spring
- Configuring Spring JPA using Spring Boot
- Spring Data JPA dynamic repositories

SPRING MVC ARCHITECTURE AND OVERVIEW

- Introduction to Spring MVC and request processing
- Controller method signatures
- Using @Controller, @RestController and @GetMapping annotations
- Configuring Spring MVC with Spring Boot
- Spring Boot packaging options, JAR or WAR

REST WITH SPRING MVC

- An introduction to the REST architectural style
- Controlling HTTP response codes with @ResponseStatus
- Implementing REST with Spring MVC, @RequestMapping, @RequestBody and @ResponseBody
- Spring MVC's HttpMessageConverters and automatic content negotiation

SPRING SECURITY

- What problems does Spring Security solve?
- Configuring authentication
- Implementing authorization by intercepting URLs
- Authorization using OAuth

- Authorization at the Java method level
- Understanding the Spring Security filter chain

ACTUATORS, METRICS AND HEALTH INDICATORS

- Enabling Spring Boot Actuator
- Custom Metrics
- Health Indicators
- Creating custom Health Indicators
- External monitoring systems

SPRING BOOT TESTING ENHANCEMENTS

- Spring Boot testing overview
- Integration testing and slices
- Slices to test different layers of the application