# Spring: Core Training

## Objectives

By the end of the course, you should be able to meet the following objectives:

- Spring configuration using Java Configuration and Annotations
- Aspect-oriented programming with Spring
- Testing Spring applications using JUnit 5
- Spring Data Access—JDBC, JPA and Spring Data
- Spring Transaction Management
- Simplifying application development with Spring Boot
- Spring Boot autoconfiguration, starters and properties
- Build a simple REST application using Spring Boot, embedded Web Server and fat JARs or classic WARs
- Implementing REST client applications using RestTemplate and WebClient
- Spring Security
- Enable and extend metrics and monitoring capabilities using Spring Boot actuator
- Utilize Spring Boot enhancements to testing

## Summary

This 40-hour course offers hands-on experience with the major features of Spring and Spring Boot, which includes configuration, data access, REST, AOP, autoconfiguration, actuator, security and Spring testing framework to build enterprise and microservices applications. On completion, participants will have a foundation for creating enterprise and cloud-ready applications.

This course prepares students for the Spring Professional certification exam.

## Target Audience

Application developers who want to increase their understanding of Spring and Spring Boot with hands-on experience and a focus on fundamentals

## Prerequisites

Knowledge on basics of Java, an IDE (Eclipse, STS or IntelliJ), and build tools such as Maven or Gradle

## Course Modules

1. Introduction to Spring
   – Java configuration and the Spring application context
   – @Configuration and @Bean annotations
   – @Import: working with multiple configuration files
   – Defining Bean scopes
   – Launching a Spring Application and obtaining Beans
2. Spring Java Configuration: A Deeper Look
   – External properties and property sources
   – Environment abstraction
   – Using Bean profiles
   – Spring Expression Language (SpEL)
3. Annotation-Based Dependency Injection
   – Component scanning
   Autowiring using @Autowired
   – Java configuration versus annotations, mixing
   – Lifecycle annotations: @PostConstruct and @PreDestroy
   – Stereotypes and meta-annotations

## 4. Factory Pattern in Spring

- Using Spring FactoryBeans

## 5. Advanced Spring: How Does Spring Work Internally?

- The Spring Bean Lifecycle
- The BeanFactoryPostProcessor interception point
- The BeanPostProcessor interception point
- Spring Bean Proxies
- @Bean method return types

## 6. Aspect-Oriented Programming

- What problems does AOP solve?
- Defining pointcut expressions
- Implementing various types of advice

## 7. Testing a Spring-Based Application

- Spring and test-driven development
- Spring 5 integration testing with JUnit 5
- Application context caching and the @DirtiesContext annotation
- Profile selection with @ActiveProfiles
- Easy test data setup with @Sql

## 8. Data Access and JDBC with Spring

- How Spring integrates with existing data access technologies
- DataAccessException hierarchy
- Spring's JdbcTemplate

## 9. Database Transactions with Spring

- Transactions overview
- Transaction management with Spring
- Transaction propagation and rollback rules
- Transactions and integration testing

## 10. Spring Boot Introduction

- Introduction to Spring Boot Features
- Value Proposition of Spring Boot
- Creating a simple Boot application using Spring Initializer website

## 11. Spring Boot Dependencies, Autoconfiguration and Runtime

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

## 12. 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

## 13. 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

## 14. 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

## 15. Spring Security

- What problems does Spring Security solve?
- Configuring authentication
- Implementing authorization by intercepting URLs
- Authorization at the Java method level
- Understanding the Spring Security filter chain
- Spring security testing

## 16. Actuators, Metrics and Health Indicators

- Exposing Spring Boot Actuator endpoints
- Custom metrics
- Health indicators
- Creating custom health indicators
- External monitoring systems

## 17. Spring Boot Testing Enhancements

- Spring Boot testing overview
- Integration testing using @SpringBootTest
- Web slice testing with MockMvc framework
- Slices to test different layers of the application

## 18. Spring Security OAuth (optional topic)

- OAuth 2 Overview
- Implementing OAuth 2 using Spring Security OAuth

## 19. Reactive Applications with Spring (optional topic)

- Overview of Reactive Programming concepts
- Reactive Programming support in Spring
- Using Spring's reactive WebClient

For additional information, please contact itacademy@vmware.com.

Item No: VMW-ITA-DS-SPRNG-CRS-USLET-101 12/21