

Mastering Spring 4.3 and the Enterprise (TT3373-S4)

Modality: Virtual Classroom

Duration: 5 Days

SATV Value:

CLC:

NATU:

SUBSCRIPTION: No

About this course:

Mastering the Spring 4.x and the Enterprise is a five-day hands-on Spring training course geared for experienced Java developers who need to understand what the Spring Framework can bring to the table in terms of today's Enterprise systems and architectures, and how to use Spring in conjunction with other technologies and frameworks. This leading-edge course provides added coverage of Spring's Aspect-Oriented Programming and the use of Spring Boot, as well as Spring security. It includes complete coverage of Spring Cloud. Students will gain hands-on experience working with Spring, using Maven for project and dependency management, and, optionally, a test-driven approach (using JUnit) to the labs in the course.

The Spring framework is an application framework that provides a lightweight container that supports the creation of simple-to-complex components in a non-invasive fashion. Spring's flexibility and transparency is congruent and supportive of incremental development and testing. The framework's structure supports the layering of functionality such as persistence, transactions, view-oriented frameworks, and enterprise systems and capabilities. Spring's Aspect-Oriented Programming (AOP) framework enables developers to declaratively apply common features and capabilities across data types in a transparent fashion.

This course targets Spring 4.x, which includes full support for Java 8 and JEE 7 (earlier versions of Java and JEE continue to be supported). Spring supports the use of lambda expressions and method references in many of its APIs.

Spring makes JEE development easier. Spring simplifies common tasks and encourages good design based on programming to interfaces. Spring makes your application easier to configure and reduces the need for many JEE design patterns. Spring puts the OO design back into your JEE application, and it integrates nicely with many view technologies and the new features of HTML5.

Note that our Spring training covers the entire spectrum and is highly modularized. As such, we can customize courses to your specific needs. The following is a high-level listing of Spring topics to consider in building your customized Spring training:

- Core Spring Framework (including Inversion of Control, Dependency Injection, and Aspect-Oriented Programming)
- Advanced Framework Features and Projects (including Spring Boot, Handling Application Events, and Spring Security)

- Spring and Persistence (including Spring DAOs, Transactions, and Spring Data)
- Spring and the Web (including Spring MVC and Web Flow supporting WebSockets, HTML5, and asynchronous processing)
- Integrating Spring into the Enterprise (including Spring Integration working with JMS and other remoting options)

The average salary of a Java Spring Developer is **\$117,087** per year.

Course Objective:

This course provides a solid understanding of what Spring brings to the table and how to use Spring in the context of other technologies and frameworks.

Students who attend **Mastering Spring 4.x and the Enterprise** will leave the course armed with the required skills to design and implement Spring applications that effectively and transparently use various enterprise systems, tools, and technologies. This course provides coverage of the concepts and practices for interacting between Spring and relational databases, security components, distributed resources, RESTful services, messaging, the Cloud, and other components.

Working in a dynamic, lab-intensive hands-on coding environment, students will learn to:

- Examine how the Spring framework handles transactions, caching and many of the non-traditional data repositories
- Understand and work Aspect Oriented Programming (AOP) within the context of the Enterprise.
- Understand and work Spring Security to acquire and process authentication credentials as well as enforce authorization on enterprise resources
- Integrate JMS into the Spring framework to utilize messaging
- Implement RESTful services and clients using the Spring framework
- Understand how to leverage the power of Spring Boot
- Use Spring Boot to create and work with JPA repositories
- Use Spring Boot to work in conjunction with Spring Cloud
- Understand the Spring Cloud project and how it relates to general Cloud concepts and practices
- Work with Spring Cloud configuration and the Config Client

Audience:

This an intermediate- level Spring 4.x training course, designed for developers who need to understand how and when to use Spring in Java and JEE applications.

Prerequisite:

Attendees should have practical basic Java development experience.

Course Outline:

Module 1: Review of Spring 4

Lesson: The Spring Framework

- Spring Architecture
- Dependency Injection
- Spring DI Container
- Bean Creation Using Factories
- Configuration Options: XML, Annotations, or JavaConfig
- Use of Lambda Expressions and Method References in Spring
- Tutorial: Setup Eclipse Neon for Using Maven
- Exercise: Hello World Spring Application (Optional)
- Exercise: Configuring Dependencies

Lesson: Spring Beans and Advanced Configuration

- Spring's Pre-Built Factory Beans
- PropertyPlaceholderConfigurer
- Custom Property Editors
- Lazy Bean Resolution
- Ordered Autowiring
- Using Configuration Classes
- Organizing Configuration Classes
- Exercise: Advanced Configuration

Module 2: Spring AOP

Lesson: Introduction to Aspect Oriented Programming

- Aspect Oriented Programming
- Cross Cutting Concerns

Lesson: Spring AOP

- Spring's AOP in a Nutshell
- The Three Technologies of "Weaving"
- Spring Advice Types
- Exercise: Spring AOP - Combined Advice
- Exercise: Spring AOP - Around Advice

Module 3: Persistence in Spring

Lesson: Overview: Persistence in Spring

- DAO Implementation
- Transaction Support
- Spring Support for JCache

- Spring Data: JPA to NoSQL

Lesson: Spring, ORMs, and Hibernate

- Benefits of Using Spring with Hibernate
- Configuring Hibernate in Spring
- Transaction Management
- Open Session in View
- Exercise: Using Spring and Hibernate

Module 4: Spring Security Framework

Lesson: Enterprise Spring Security

- Spring Security Framework
- Security Interceptors
- Authentication Managers
- Wiring in Encoders and Salts
- Access Decision Managers

Lesson: Spring Web Security

- Spring Security Transparent to Client
- Standard Set of Filters
- Spring Security Config File
- Securing Java Code
- Securing Java Spring Beans
- Exercise: Using Spring Security (Pt 1)
- Exercise: Using Spring Security (Pt 2)
- Exercise: Using Spring Security (Pt 3)
- Exercise: Using Spring Security (Pt 4)

Module 5: Spring JMS

Lesson: JMS Overview (Optional)

- Java Message Service (JMS)
- The JMS Factory Model
- JMS Queue Architecture
- Topic Architecture
- Messages

Lesson: Spring and JMS

- JmsTemplate
- Callback Methods
- Spring Messaging Module
- Message Converters

- MessagePostProcessor
- Destinations
- Working with @JmsListener
- Exercise: Using JMS with Spring

Module 6: Implementing REST with Spring

Lesson: RESTful Services in Spring

- Understand how Spring supports the implementation of RESTful services
- Use Spring to map URIs and extract values from the URI
- Work with @RequestMapping to support routing decisions based on what type should be processed by associated method
- Handle response codes
- Work with view resolvers, HTTP message converters, and content negotiation
- Exercise: Working with Spring REST

Lesson: RESTful Clients in Spring

- Understand how Spring supports browser-based RESTful clients
- Understand how Spring supports Spring-based RESTful clients
- Exercise: Injection in Spring REST
- Exercise: Exception Mapping in Spring REST
- Exercise: Content Negotiation in Spring REST

Module 7: Spring Boot

Lesson: Spring IO Platform

- Understand the Spring IO Platform
- Understand the IO Bill of Materials
- Understand the IO Foundation
- Learn how the IO Execution will be leveraged
- Learn how Spring Cloud is used for Platform Coordination

Lesson: Spring Boot Overview

- What is Spring Boot
- Explore Spring Boot starters
- Examine Spring Boot's AutoConfiguration as well as its command-line interface (CLI)
- Understand the Spring Boot Actuator

Lesson: Spring Boot Introduction

- Spring Boot JPA Starter
- Examine Spring Boot's AutoConfiguration
- Understand the Spring Conditionals
- Understand Spring Boot DevTools

- Exercise: Create a "REST JPA Repository"

Lesson: Advanced Spring Boot

- Explore additional Spring Boot starters
- Bootstrapping Spring Boot
- Understand Spring Boot Actuators
- Create and run a Spring Thymeleaf MVC application
- Exercise: Create a "Thymeleaf MVC With JPA Repository"

Module 8: Spring Cloud

Lesson: Spring Boot Additional Configuration

- Common Property Overrides
- ConditionEvaluationReport
- Additional Configuration Options
- Working with AutoConfiguration with Spring Cloud
- Exercise: Preparing Spring Boot for Spring Cloud

Lesson: Spring Cloud Project

- Spring Cloud Project Overview
- 12-Factor Applications
- Spring Cloud Tools Supporting 12-Factors
- Registration and Discovery
- Circuit Breakers

Lesson: Spring Cloud Configuration

- Spring Cloud Config Project
- Config Server and Client
- Securing Server URI's
- Storing Config Properties
- Accessing Config Properties
- Exercise: Working with Config Server
- Exercise: Config Branching

Lesson: Spring Cloud Configuration Client

- Spring Cloud Config Client
- Starter and Bootstrap
- Overriding Remote Properties
- Exercise: Working with Config Client