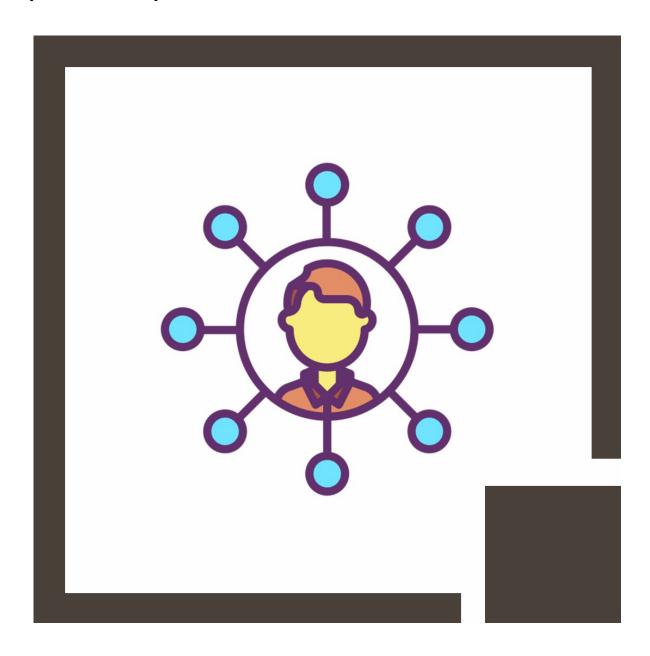**Document Generated: 12/18/2025**

Learning Style: Virtual Classroom

Technology:

Difficulty: Intermediate

Course Duration: 5 Days

Mastering Spring 4.3 and the Enterprise (TT3373-S4)

About this course:

Learning the Spring 4.x Framework is a 5 days practical Spring training program for experienced Java developers who want to learn what the Spring Framework is for today's architectures and systems, and how Spring can be used in combination with other frameworks and technologies. This online training provides additional coverage of Aspect-Oriented programming from Spring and the use of Spring Security, and also Spring Boot. It provides a full overview of all new functions in Spring 4.x. Learners can obtain practical experience collaborating with Spring, utilizing Maven for dependency and project management and, additionally, a test-driven method (utilizing JUnit) to the workshops in the program.

The Spring system is a framework that offers a lightweight container that supports the development in the non-invasive fashion of simple to complex components. Flexibility and openness from Spring align with and help gradual growth and testing. The structure of the platform facilitates the layering of features like view-oriented interfaces, transactions, persistence, and capabilities and enterprise systems.

This program aims Spring 4.x, which provides complete support for Java 8 and Java EE 7 (it continues to support previous versions of Java and Java EE). In most of its application program interfaces, Spring encourages the use of lambda expressions and process references.

Spring encourages the production of Java EEs. Spring streamlines specific tasks and favors strong interface-based design. Spring makes it simpler to customize your program and reduces the need for many Java EE design patterns. Spring brings the OO style back into your Java EE app and suits well with other view innovations and HTML5's latest features.

Remember that our Spring training spans the whole spectrum and is extremely flexible. As such, we will customize the programs to your individual needs. A high-level overview of Spring topics to consider when designing your personalized Spring training is as follows:

- Advanced Project and System Features (such as Spring Security, Managing Application Events, and Spring Boot)
- Spring and Web (such as WebSockets Supporting Spring MVC and Cloud Flow, asynchronous processing, and HTML5)
- Core Spring System (such as Aspect-Oriented Programming, Injection of Dependencies, and Control Inversion)
- Integrating Spring into the organization (such as JMS operating with Spring Integration and other remote options)
- Spring and Persistence (such as Transactions, Spring Data, and Spring DAOs)

The Java Spring Developer can earn an average salary of $117,087 per annum.

Course Objective:

- Understanding and operating Spring Protection to obtain and manage authentication keys and to impose authorization on client resources
- Comprehend how Spring Boot can leverage its power
- Evaluate how the Spring Framework manages caching, transactions and many of the non-conventional repositories of data
- Comprehend the project at Spring Cloud and how it applies to general technology principles and activities
- Using Spring system to implement RESTful services and clients
- Knowing and working AOP within the Enterprise context
- Using Spring Boot to work alongside Spring Cloud
- Integrate JMS in the Spring System for Messaging
- Working with Spring Cloud and the Setup Client
- To build and work with JPA repositories using Spring Boot

Audience:

This is a Spring 4.x intermediate- course for developers who want to learn when and how to use Spring in Java and Java EE apps.

Prerequisite:

The participants should have practical experience in fundamental Java development.

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