

Specialized - Mastering Spring 4.2

Code:	TT3330-S4
Length:	5 days
URL:	View Online

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. This course targets Spring 4.2, 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.

Skills Gained

- Explain the issues associated with complex frameworks such as JEE and how Spring addresses those issues
- Understand the relationships between Spring and JEE, AOP, IOC, JDBC, Hibernate, JSF, Struts, JMS, and EJBs.
- Write applications that take advantage of the Spring container and the declarative nature of assembling simple components into applications.
- Understand how to configure the framework with XML, annotations and JavaConfig as well as explore the advantages of each option.
- Understand and work on integrating persistence into a Spring application.
- Explain Spring's support for transactions and caching
- Understand and work with various options for integrating view-oriented frameworks for web applications into spring.
- Work with Spring Boot and JavaConfig to more effectively and efficiently develop Spring applications.
- Work with various Spring Application Events, implementing and registering listeners, and monitoring for transactional events

Who Can Benefit

- Experienced Java Developers

Prerequisites

- TT2100 Mastering Java for OO Developers

Course Details

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

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

Lesson: Spring AOP

- Aspect Oriented Programming - AOP
- Crosscutting Concerns
- Spring's AOP in a Nutshell
- The Three Technologies of "Weaving"
- Four Kinds of Advice
- Joinpoint and Pointcuts

Lesson: Using Annotations with AOP

- Using Annotations to Create Aspects
- Other Advice Types
- Types of Pointcuts
- Spring Pointcut Pattern Expressions
- Introductions

Lesson: Overview: Persistence in Spring

- DAO Implementations
- Transaction Support
- Spring Support for JCache
- Spring Data: JPA to NoSQL

Lesson: Spring JDBC

- JdbcDaoSupport - JDBC DAO Implementation
- The jdbcTemplate
- Exception Handling
- Operation Classes

Lesson: Spring and Hibernate

- Benefits of Using Spring with Hibernate

- Configuring Hibernate in Spring
- Transaction Management
- Open Session in View

Lesson: Overview: Spring Views

- SpringMVC and Web Flow
- View Technologies
- WebSockets, SockJS, and STOMP
- Support for AppCache and HTML5
- SpringMVC Test Framework

Lesson: SpringMVC Overview

- The DispatcherServlet
- Workflow of Request Handling
- Using Handler Mappings
- ModelAndView and View
- Controllers
- Spring Form Tags

Lesson: HandlerMapping

- Mapping URLs to Controllers
- HandlerMapping Interface
- RequestMapping Annotation
- Mapping Requests
- Custom Handler Mapping

Lesson: Controllers

- Spring MVC's Controller Hierarchy
- How to Select a Controller
- Controller Interface
- Controller annotation
- Form Controllers
- Handling Exceptions
- Testing Controllers

Lesson: Spring's form Tag Library

- The Spring Form tags
- Using a PropertyEditor
- Survey of form tags
- Support for HTML5 Attributes

Lesson: Application Events in Spring

- Overview of Application Events
- Built-in Events
- @EventListener
- Returning Object from Listener
- Transaction-Bound Events

Lesson: Spring Boot

- Understanding Spring Boot
- Convention Over Configuration
- Working with Embedded Servers
- Providing metrics, health checks, and more

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

Lesson: JMS Overview (Optional)

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

Lesson: Spring and JMS

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

- Working with @JmsListener
-

ExitCertified® Corporation and iMVP® are registered trademarks of ExitCertified ULC and ExitCertified Corporation and Tech Data Corporation, respectively
Copyright ©2019 Tech Data Corporation and ExitCertified ULC & ExitCertified Corporation.
All Rights Reserved.

Generated 1