Core Spring Foundations

Course Summary

Description

This course is designed for Java programmers that want to master the core capabilities of the spring framework. The course assumes no previous experience with spring and shows students the most important and critical features of spring including: dependency injection, aspect oriented programming, data access, JPA integration, MVC, and testing. The course includes an integrated case study that walks students through the process of designing and building a complete spring application.

Topics

- Dependency Injection / Inversion of Control
- Spring Architecture
- Spring IOC container
- Aspect Oriented Programming
- Spring Aspect Oriented Programming
- Spring Data Access Framework
- Integrating Spring and JPA
- Spring MVC
- Testing spring based applications
- Spring Integrated Case Study
- Integrating Spring into legacy applications (Optional)
- Integrating Spring with Struts (optional)
- Integrating Spring JSF (optional)

Audience

This course is designed for programmers & Architects who need to quickly get up to speed on Spring.

Prerequisites

Students should have Java programming experience.

Duration

Five days

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically

ProTech
protechtraining.com

"Charting the Course ...

... to Your Success!"

Core Spring Foundations

Course Outline

I. Dependency Injection
- A. What is Dependency Injection
- B. Benefits of Dependency injection
- C. Dependency Injection Frameworks
- D. Cost of Dependency injection

II. Spring Architecture
- A. Overview of Spring
- B. Architecture of the Spring Project
- C. Spring Distribution
- D. Configuring Spring Projects
- E. Selecting which parts of Spring to use

III. Spring IOC Container
- A. Spring IOC Container Overview
- B. XML based Spring IOC
- C. Factory Beans
- D. Bean Life cycle
- E. Annotation based Spring IOC

IV. Aspect Oriented Programming
- A. What problem does AOP Solve?
- B. Code tangling, Code Scattering
- C. Cross Cutting Concerns
- D. Aspects, Weaving, Join Points, Point Cuts, and Introductions.
- E. AOP programming environments
- F. Load Time Weaving, Compile Time Weaving

V. Spring AOP
- A. Java Dynamic Proxies
- B. Java Byte code Enhancement
- C. Proxy based Spring AOP
- D. Configuring aspects using XML
- E. Configuring aspects using AspectJ annotations
- F. Integrating Spring and AspectJ Load Time Weaving

VI. Spring Data Access Framework
- A. Spring Data Access Architecture
- B. Common Exception Hierarchy
- C. Spring DaoSupport classes
- D. Transaction Management with Spring
- E. Spring JDBCTemplate class

VII. Integrating Spring and JPA
- A. JPA Overview and Review
- B. Spring JPA support
- C. Spring Transaction Management with JPA
- D. Implementing Data Access Objects with Spring and JPA
- E. Configuring Spring + Hibernate + JPA

VIII. Spring MVC
- A. Spring MVC Architecture
- B. Developing Controllers
- C. Developing Views
- D. XML Based Spring MVC configuration
- E. Annotation based Spring MVC configuration
- F. Data Validation

IX. Integrating Spring with Web Frameworks (Optional Module)
- A. Spring Struts Integration
- B. Spring JSF integration

X. Testing Spring Based Applications
- A. Spring Unit testing support
- B. Spring Mock testing support
- C. Testing Data Access Objects
- D. Testing Spring MVC code

XI. Spring Integrated Case Study
- A. Students will go through the process of designing and building a small phone book application using spring. This integrated case study will ensure that all students in the class are able to apply all the fundamental features of spring to building applications.

XII. Integrating Spring into legacy applications
- A. This module shows students various scenarios in which spring can be introduced into existing java based applications. We discuss technical obstacles to using spring with legacy applications as well as the solutions.

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically