

Spring Training Outline

Topic covered in spring (4 days)

- **spring core (.5 day)**
- **spring MVC (.5 day)**
- **spring DAO and spring ORM (.5 day)**
- **spring AOP (.5 day)**
- **spring boot and spring data (1 day)**
- **spring security (.5 day)**
- **spring rest and spring transaction (.5 day)**

Detail outline SPRING

SPRING CORE AND MVC

- a. Shortcomings of Java EE and the Need for Loose Coupling
- b. Managing Beans, The Spring Container, Inversion of Control
- c. The Factory Pattern
- d. Configuration Metadata - XML, @Component, Auto-Detecting Beans
- e. Dependencies and Dependency Injection (DI) with the BeanFactory
- f. Setter/Constructor Injection
- g. Using the Application Context
- h. Configuring Collections
- i. Bean Definition Inheritance and Collection Merging
- j. The Spring Managed Bean Lifecycle
- k. Key interfaces, Annotations
- l. Autowiring Dependencies
- m. Annotation Configuration @Autowired, @Required, @Resource
- n. MVC2 pattern
- o. @Component, Component Scans. Component Filters
- p. @Value and @Qualifier
- q. Java Configuration, @Configuration, XML free configuration (Optional)
- r. The AnnotationConfigApplicationContext SPRING

SPRING ORM AND AOP

1. introduction of
Hibernate and JPA
2. Spring and ORM
3. HibernateTemplate
4. spring aop
5. PointCuts, JoinPoints, Aspects, Advices
6. Before, After, AfterReturning, AfterThrowing, Around
7. Annotation Configuration
8. XML Configuration SPRING

Spring transaction and boot spring data

- i. TRANSACTIONS
 - ii. PlatformTransactionManager and its subtypes
 - iii. Transaction attributes
 - iv. Working with HibernateTransactionManager
 - v. @Transactional, @TransactionalAttribute
 - vi. why spring boot ?
 - vii. zero configuration
 - viii. spring core with boot
 - ix. spring mvc with boot
-
- i. Spring Boot Configuration
 - 1. Profiling
 - 2. Exception Handling
 - ii. spring data

Spring security and rest

Spring security

authentication via BASIC and FORM

spring unit testing

session 3&4 (2:00pm-6:00pm) spring REST

- i. RESTFUL WEB SERVICES
 - 1. Core REST concepts
 - 2. REST support in Spring 4.x
- ii. Use Spring MVC to create RESTful Web services
- iii. REST specific Annotations in Spring
- iv. URITemplates, @PathVariable, @RequestParam, JSON and XML data exchange
- v. @RequestMapping
- vi. Contract Negotiation (CONNEG)
- vii. Client access with HttpClient or RestTemplate