# Course : JAVA - Spring Boot – Intermediate training Type : Hands on Training

**Hours : 24/6Days**

**Day1 – (4hrs)**

# Spring Basic

* Spring basic concept
* Environment setup (java, maven, spring etc)
* Dependency Injection as an Idea
* Spring Introduction  Concept of 'Bean'
* Starting a Spring based application
* **Spring Boot** Packaging Spring Boot Application as a 'fat jar'

**Day2 – (4hrs)**

# Spring As a DI Container

* Importance of application layering
* Problem of tight coupling
* Problem of easy unit-testability
* Configuration Metadata - XML
* ApplicationContext object and its use
* Spme patterns used in Spring o Factory Method o Singleton, Prototype, Strategy and Proxy

# Configuring Spring Beans

* Setter Vs Constructor based DI
* Bean Scopes and Lifecycle
* Autowiring dependencies
* Annotation Based Configuration - @Component, @Repository, etc.,
* Component Scanning
* Java Based Configuration - @Configuration, @Bean etc.,

**Day3 – (4hrs)**

# Spring Boot

* Why Boot?
* Spring Boot Maven Dependency Descriptors
* Spring Boot Starter Kits  Spring Boot Auto-Configuration
* Spring Boot Application Directory Structure
* @SpringBootApplication annotation details Configuring Spring Boot
* External properties & Property sources o Using Zookeeper for configuration Management
* Environment abstraction
* Using bean profiles
* application.properties and yml file(s) and their precedence
* Logging in Spring Boot Application

**Day4 – (4hrs)**

# Basics of Java Persistence API (JPA)

* How JPA works?
* Requirements of an Entity class
* Simple mapping using @Entity, @Id, @Table, @Column
* Spring Boot Starter for JPA
* Setting requisite properties in application.properties for JPA
* Injecting EntityManager
* Intro to Spring Transaction Management
* Declarative Transaction Management with @Transactional Spring Data JPA
* Spring Data Repositories
* Core concepts
* Query methods
* Defining repository interfaces
* Defining query methods
* Query creation
* Special parameter handling

**Day5 – (4hrs)**

# Spring Boot Web

* Request Processing in Spring Web MVC
* Dispatcher Servlet and its responsibilities
* Web Application starter in Boot
* Contrasting War Vs. Jar Deployment
* Excluding Web Container for WAR deployments
* Configuring Web Container Developing REST Services using Spring
* REST Basics - Correct use of HTTP's GET,POST,PUT and DELETE
* Using the most appropriate HTTP Status Codes for Responses
* REST Con-neg introduction
* Using @RequestBody, @ResponseBody and @RestController annotations
* Using ResponseEntity to fully control responses
* @RequestParam and Parameter Binding
* Writing Controllers, @Controller, @RequestMapping, @RequestParam, @PathVariable
* Consuming REST API using RestTemplate

**Day6 – (4hrs)**

# Spring Boot Web (continue )

* Using PostMan tool
* Introduction to HATEOAS Unit Testing and Integration Testing with JUnit and Mockito
* Setting up the project using Spring Initializr
* Writing Unit Test for a Simple Business Service
* Setting up a Business Service to call a Data Service
* Writing your first unit test with Stub
* Problem with Stubs.
* Writing Unit Tests with Mocking using Mockito
* Use Mockito  Refactoring - @Mock, @InjectMocks and @RunWith(MockitoJUnitRunner.class)
* Unit Testing Web Tier
* Working with MockMvc
* @MockBean annotation
* Using JsonPath for result assertion
* Integration Testing Spring Boot application