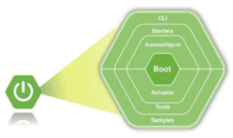
www.spiraltrain.nl

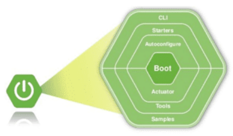
Spring Boot Development

Duration: 4 days **Price: € 2650**

Open Schedule



Spring Boot Development



Audience Spring Boot Development

The course Spring Boot Development is intended for experienced Java Developers who want to use Spring Boot for application development.

Prerequisites Course Spring Boot Development

Experience with programming in Java and object orientation is required to participate in this course. Basic knowledge of the Spring Framework is beneficial to good understanding.

Realization Training Spring Boot Development

The concepts are treated on the basis of presentations and demos. The theory is interspersed with exercises. The course times are from 9.30 to 16.30.

Certification Spring Boot Development

Participants receive an official certificate Spring Boot Development after successful completion of the course.

Content Course Spring Boot Development

In the course Spring Boot Development you will learn to develop applications and microservices with Spring Boot in a fast and efficient way. The training covers essential features of Spring Boot such as opinionated Spring Boot Starters, embedded servers, automatic configuration, metrics and externalized configuration.

Spring Intro

The course starts with an overview of the most important Spring principles such as loading beans in the bean container and dependency injection.

Spring Boot

Subsequently it is discussed how predefined configurations in Spring Boot act as a starting point for a Spring Boot application. Other main components of Spring Boot are also discussed, such as the Autoconfigurator, the Actuator and the Command Line Interface (CLI).

Dependency Injection

Dependency injection with its associated annotations such as @Component, @Qualifier, @Repository and @Service is treated in detail. The internal workings of dependency injection based on Java Reflection is also explained.

Application Configuration

Also part of the course program are the auto configuration options in Spring Boot. Components can be linked with annotations such as @EnableAutoConfiguration if registered in classes annotated with @Configuration.

Aspect Orientation

And Aspect Orientation in Spring Boot is discussed as well. Crosscutting concerns in an application such as security or profiling can thus be included in so called aspects without disrupting the main program flow.

Spring JDBC and Spring Data

Database access from Spring Boot applications is covered in the modules Spring JDBC and Spring Data. Various Spring templates that help prevent boilerplate code such as jdbcTemplate and MongoTemplate as well as JPA repositories are discussed.

Spring REST

Spring Boot is ideally suited to access REST APIs with compact code. The various annotations that are important here, such as @RestController, @ResponseStatus and @JsonView, are treated.

Spring Extensions

Finally various Spring extension projects are covered such as Spring Security and Spring Cloud.

SpiralTrain BV
Standerdmolen 10, 2e verdieping
3995 AA Houten

info@spiraltrain.nl
www.spiraltrain.nl
Tel.: +31 (0) 30 – 737 0661

Locations
Houten, Amsterdam, Rotterdam, Eindhoven, Zwolle, Online

SpiralTrain
developer courses

Modules Course Spring Boot Development

Module 1 : Spring Core	Module 2 : Spring Boot	Module 3 : Dependency Injection
Spring Framework Overview Spring Configuration Spring Dependency Injection Non IoC versus IoC Application Context Beans Life Cycle XML Configuration Configuration with Annotations Component Scanning Spring Java Configuration Aware Interfaces	What is Spring Boot? Advantages Spring Boot Spring Boot Flavors Key Spring Boot Components Spring Boot Starter Starter Dependencies Spring Boot Autoconfigurator @SpringBootApplication Spring Boot CLI Spring Boot Internals Spring Boot Actuator	Non-IoC or Dependency Injection Benefits of Dependency Injection Constructor Dependency Injection Setter Dependency Injection Autowiring with @Autowired @Qualifier Annotation @Component Annotation @Repository and @Service Bean scopes Event Handling Internationalization
Module 4 : Application Configuration	**Module 5 : Aspect Orientation**	**Module 6 : Spring JDBC**
Configuration Classes @Configuration Annotation @Bean Annotation @Enable Annotations @EnableAutoConfiguration Autowiring and Component Scanning @EnableScheduling Wire External Values Spring Expression Language @Value Annotation @PropertySource Annotation	What is AOP? The need for AOP Crosscutting Concerns Traditional Approach Spring AOP AOP Concepts AOP Key Terms Aspects and Weaving Pointcuts and Joinpoints ProxyFactoryBean Spring AOP Configuration	Spring and JDBC JDBC Architecture JDBC Drivers and URL's Spring JDBC Data Access Spring DAO with JdbcTemplate Data Source Injection Querying using JdbcTemplate RowMapper Querying and Populating Objects Updating with JdbcTemplate ResultsetExtractor
Module 7 : Spring Data	**Module 8 : Spring REST**	**Module 9 : Spring Security**
What is Spring Data? Spring Data Configuration CRUD Out of the Box JPA Repositories Persisting and Modifying Entities Spring Data Queries @Query Annotation Named and Async Queries Paging Results Transaction Handling @Transactional Annotation MongoDB Template Mapping and Inserting Documents	REST Web Services @RestController HttpEntity and ResponseEntity Default Content Types Default Status Codes @ResponseStatus and HttpStatus Working with XML and JSON Multiple Representations Filtering with @JsonView REST Clients RestTemplate Sending HTTP Requests Reading Responses	Spring Security Model Process Behind Security Interceptors Authentication Manager Configuring authentication Intercepting URLs Security at the method level Access Decision Manager Security Based on Roles Security Based on Identity Access Denied Handling Securing REST Services JSON Web Tokens OAuth2 Authentication
Module 10 : Spring Cloud		
What is Spring Cloud? Spring Cloud Config Eureka Service Spring Cloud Bus Spring Cloud Cluster Spring Cloud Security Spring Cloud Data Flow Spring Cloud Connectors Spring Cloud Task App Starters Spring Cloud Zookeeper Spring Cloud CLI Spring Cloud Gateway Spring Cloud Pipelines		