

Java Development with Spring

Audience Java Development with Spring Course

Experienced Java developers who want to use the Spring Framework for Java Desktop, Web and Enterprise applications.

Prerequisites Course Java Development with Spring

Experience with Java programming and object orientation is required to participate in this course. Basic knowledge of Web Applications and XML is beneficial for a proper understanding.

Realization Training Java Development with Spring

The concepts are treated by means of presentation slides and demos. The theory is interspersed with exercises. The course material is in English. The course times are from 9.30 up and to 16.30.

Certification Java Development Spring

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

Duration: 5 days

Price: € 2999

[Open Schedule](#)



Java Development
with Spring



Content Java Development with Spring

The course Java Development with Spring covers the concepts, components and architecture of the Spring Framework. The course is done with the latest Spring version with Spring Boot. In the course, annotations are central to the configuration of the Spring Applications.

Spring Intro

The course Java Development with Spring starts with an overview of the different parts of Spring and the types of applications in which Spring can be used.

Spring Boot

Next the essentials of Spring Boot, which greatly simplifies the configuration of Spring applications, are discussed. The main Spring Boot annotations, the opinionated defaults and the built-in web servers are treated.

Dependency Injection

Attention is paid to the concept of Dependency Injection or Inversion of Control which plays a central role in the framework. Different variants of dependency injection such as setter injection and constructor injection are discussed.

Aspect Oriented Programming

The other pillar of the framework, Aspect Orientation, is also covered in detail. The concepts of Aspect Orientation such as Aspects, Joinpoints, Pointcuts, Advice and Weaving are explained.

Data Access

Next the different options for storing the data of Spring Java applications in databases are discussed. Attention is paid to the use of JDBC with a JdbcTemplate, as well as the use of Object Relational Mapping frameworks such as Hibernate with a HibernateTemplate or JPA via annotations. Spring Transactions are also treated in this context.

Spring MVC

Web Applications with the Spring MVC Framework are part of the program of the course Java Development with Spring as well as the creation and use of Spring Rest Web Services. The role of controllers, views, page parameters and command objects is explained.

Spring Security

Finally attention is paid to Spring and Security and the simplified setup of a Spring configuration with Spring Boot is treated. The modules Spring with JMS and Spring with JMX are optional.

Modules Java Development with Spring

Module 1 : Spring Introduction	Module 2 : Spring Boot	Module 3 : Dependency Injection
Spring Mission Statement Spring Addressing Layers Spring Characteristics Spring Jars Core Container Packages Spring AOP Packages Data Access Packages Spring Web Packages Spring Versions Spring Projects Spring Configuration	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	Benefits of DI BeanFactory Bean Configuration File Application Context Setter Injection Constructor Injection Beans Life Cycle Injection Simple Values Autowiring Aware Interfaces ResourceLoader
Module 4 : Application Configuration	Module 5 : Aspect Orientation	Module 6 : Spring Persistence
XML Configuration with Namespaces Best Practices XML Configuration Bean Definition Inheritance Spring XML Namespaces Component Scanning Configuration with Annotations Context Namespace Spring Expression Language Autowiring Spring Java Configuration Spring Expression Language PropertyPlaceholder Configuration	The need for AOP Crosscutting Concerns AOP Key Terms ProxyFactoryBean Spring AOP Configuration Aspects and Advices Join Points and Pointcut AspectJ Configuration Advice Arguments Programmatic Proxies Regexp Pointcut NameMatchMethod	Java Persistence Traditional Persistence Transparent Persistence Shared Persistence Concepts DAO Design Pattern DAO Pattern JDBC Integration with IoC DAO Portability Spring DAO Concepts Transaction Management Spring Exceptions Exception Translation
Module 7 : Spring JDBC	Module 8 : Spring ORM	Module 9 : Transactions
JDBC Architecture Executing Statements 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 Callbacks NamedParameterJdbcTemplate	Spring and Hibernate Integration Mapping Classes HibernateTemplate Implementation HibernateTemplate HibernateTemplate execute Hibernate Annotations Spring and JPA Using JPA API Persistence Unit Configuration Container Managed Transactions Externalizing DB Properties Entity Manager from JNDI JPA Java Configuration	Using Spring Transactions Transaction Strategy TransactionDefinition TransactionStatus Injecting Implementations Declarative Transactions Transactional Methods Transaction Configuration Rollback Rules Using @Transactional Transaction Propagation Programmatic Transactions Using TransactionTemplate
Module 10 : Spring REST	Module 11 : Spring and Security	Module 12 : Spring JMS
@RestController HttpEntity and ResponseEntity Default Content Types Default Status Codes @ResponseStatus and HttpStatus XML and JSON Multiple Representations Filtering with @JsonView RestTemplate Sending HTTP Requests Translating Entities Reading Responses Error Handlers	Spring Security Security Concerns Spring Authentication Spring Security Modules Spring Security Elements Defining Allowed Users Form Authentication Authentication Managers Security Interceptor Security Filters Access Decision Manager Web Authorization Method Authorization	Messaging Characteristics Messaging Models JMS Architectural Components Spring JMS JMS with Spring Spring JMS Options Spring JmsTemplate Send and Convert Execute Method ProducerCallback Receive and Convert Message Listener Containers MessageListenerAdapter