SpriralTrain BV
Standerdmolen 10, 2e verdieping.
3995 AA Houten

spiraltrain@gmail.com
www.spiraltrain.nl
Tel.: (0)30 737 0661
Fax: (0)30 737 1050

KvK nr. : 61618039
BTW nr. : NL854415798B01
IBAN : NL58RABO0156271540
BIC : RABONL2U

JAV800: Cursus Java Development met Spring

Code: JAV800 **Duur:** 4 dagen **Prijs:** € 1700

Doelgroep Cursus Java Development met Spring

Ervaren Java developers die het Spring Framework willen gebruiken voor Java Desktop, Web of Enterprise applicaties.

Voorkennis Cursus Java Development

Ervaring met programmeren in Java en object oriëntatie is vereist om deel te kunnen nemen aan deze cursus. Basis kennis van web applicaties en XML is bevorderlijk voor een goede begripsvorming.

Uitvoering Training Java Development met Spring

De concepten worden behandeld aan de hand van presentatie slides en demo's. De theorie wordt afgewisseld met oefeningen. De cursustijden zijn van 9.30 tot 16.30.

Java Development with Spring

Certificering Java Development met Spring

De deelnemers krijgen na het goed doorlopen van de cursus een officieel certificaat Java Development met Spring.

Inhoud Cursus Java Development met Spring

Deze cursus behandelt de concepten, componenten en architectuur van het Spring Framework. Ruime aandacht wordt besteed aan het concept van Dependency Injection of Inversion of Control dat een centrale rol speelt het framework. Verschillende soorten van een dergelijke dependency injection zoals setter injection en constructor injection worden besproken. De andere pijler van het framework, Aspect oriëntatie, komt ook aan de orde. De concepten van Aspect Orientation zoals Aspects, Joinpoints, Pointcuts, Advice en Weaving worden toegelicht. Vervolgens worden de verschillende opties om de gegevens van Spring Java applicaties in databases op te slaan behandeld. Er wordt aandacht besteed aan het gebruik van JDBC met een JdbcTemplate, alsmede op het gebruik van Object Relational Mapping frameworks zoals Hibernate met een HibernateTemplate of JPA via annotaties. In dit verband worden Spring Transacties besproken. Web Applications met het Spring MVC Framework zijn ook onderdeel van het cursus programma evenals de creatie en het gebruik van Spring Rest Web Services. Hierbij wordt de rol van controllers, views, page parameters en command objects besproken. Tot slot wordt aandacht besteed aan Spring en Security en wordt het vereenvoudigd opzetten van een Spring configuratie met Spring Boot besproken. De modules Spring met JMS en Spring met JMX zijn optioneel.

Module 1 : Spring Introduction	Module 2 : Dependency Injection	Module 3 : Application Configuration
What is Spring? Addressing Layers Characteristics Framework Overview Dependency Injection Inversion of Control Aspect Oriented Programming Portable Service Abstractions Spring Packages	Non-IoC or Dependency Injection Benefits of Dependency Injection Constructor Dependency Injection Setter Dependency Injection Bean Factory XmlBeanFactory Bean Configuration File Injection Parameter Types Bean Naming Autowiring Properties Application Context Multiple configuration files Working with interceptors Externalizing constant values Bean scopes	Bean definition inheritance Inner beans p and util namespaces Dependency injection of collections Spring Expression Language Autowiring and component scanning Stereotype annotations Java-based configuration Mixing configuration styles When to use XML, annotations, and Java configuration Testing Applications
Module 4 : Aspect Orientation	**Module 5 : Spring Persistence**	**Module 6 : Spring JDBC**
Aspect Oriented Programming The need for AOP Crosscutting Concerns Aspect Joinpoints Pointcuts Advise Weaving Target Introduction Spring AOP Static AOP Dynamic AOP Proxies ProxyFactory	Spring and Persistence Java Persistence Traditional Persistence Transparant Persistence Shared Persistence Concepts DAO Design Pattern Before and after DAO DAO Pattern JDBC Integration with IoC DAO Portability Spring DAO Concepts Transaction Management Spring Exceptions Exception Translation	Spring and JDBC JDBC Characteristics JDBC Architecture Executing Statements JDBC Drivers and URL's Spring JDBC Data Access DAO with JdbcTemplate Data Source Injection Querying using JdbcTemplate RowMapper Querying and Populating Objects Updating with JdbcTemplate ResultsetExtractor Callbacks SimpleJdbcTemplate NamedParameterJdbcTemplate

		JdbcDaoSupport
Module 7 : Spring ORM	**Module 8 : Transactions**	**Module 9 : Spring MVC**
Spring and Hibernate Hibernate Integration Mapping Classes HibernateTemplate Implementation HibernateTemplate HibernateTemplate execute Hibernate DAO Implementation Hibernate Annotations Spring and JPA LocalEntityManagerFactoryBean Using JPA API Persistence Unit Configuration LocalContainerEntityManagerFactoryBean Persistence Configuration PersistenceExceptionTranslationProcessor Container Managed Transactions Externalizing Database Properties Entity Manager from JNDI JpaTemplate and JpaDaoSupport JPA Java Configuration	Transaction Managers Declaring Transaction Managers Programmatic Transactions Transaction Callback API @Transactional annotation Declarative Transactions Isolation Levels Read-Only Hint Timeouts Declaring a Transaction Manager Configuring transaction propagation Transactions and integration testing	What Spring MVC? Request life-cycle DispatcherServlet URL Handler mapping Matching URLs Matching Methods Matching Content Types Path Variables Request Parameters Headers and Cookies Injectable Method Parameters Form Submissions Command Objects vs. Entities @RequestBody @ResponseBody Producing Responses ResponseEntity Spring MVC Validation
Module 10 : Spring REST	**Module 11 : Spring and Security**	**Module 12 : Spring BOOT**
REST Web Services @RestController HttpEntity and ResponseEntity Default Content Types Default Status Codes @ResponseStatus and HttpStatus Working with XML Working with JSON Multiple Representations Filtering with @JsonView REST Clients RestTemplate Sending HTTP Requests Translating Entities Reading Responses Error Handlers	Spring Security Model Process Behind Security Interceptors Authentication Manager Configuring authentication Intercepting URLs Security tag library for JSPs Security at the method level Customizing the Security filter chain Access Decision Manager Security Based on Roles Security Based on Identity Run-as Manager Custom Login Pages After Invocation Manager XSD Extensions Using Annotations	Convention over Configuration NO XML Spring Boot CLI Building and Deploying an Application Using Templates Gathering Metrics Using Java With start.spring.io Spring Boot Starters Building as a Runnable JAR Data Access with Spring Data Property Support Securing an Application Authentication and Authorization
Module 13 : Optional : Spring JMS	**Module 14 : Optional : Spring JMX**	
What is JMS? Messaging Characteristics JMS API Messaging Models JMS Architectural Components JMS Message Interfacs Configuring JMS resources with Spring Using the JmsTemplate Message listener containers	What is JMX? JMX API Managed Beans MBean flavors JMX Architecture Naming MBeans MBean Server Registering Mbeans Manipulating MBeans Export MBeans automatically	