

SPRING FRAMEWORK DEVELOPMENT

Course Description

The Spring Framework is one of the leading lightweight architectures for creating enterprise scale applications in Java. This course describes how to use Spring to create Web applications, Web services, middle-tier components, data-access components using JDBC, and messaging components using JMS.

You will also learn how to use Inversion of Control (IoC) and Dependency Injection to minimize dependencies and to achieve Test-Driven Development, and how to use Aspect-Oriented Programming (AOP) to implement cross-cutting functionality, and get an introduction to Spring Boot.

What you'll learn

- Creating and using beans
- Implementing dependency injection
- Aspect-oriented programming using Spring
- Using Spring data access and transaction APIs
- Creating Spring MVC Web applications
- Defining and using Web services using Spring
- Introduction to Using Spring Boot

Duration

4 days

Target Audience

Java Programmers, Web Developers

Course Prerequisites

At least 6 months Java programming experience

Course Content

Spring Framework – Essentials

- Overview of Spring
- Dependency injection and Inversion of Control (IoC)
- Aspect-Oriented Programming (AOP) with Spring
- Test-Driven Development principles
- Defining a first application

Using Inversion of Control (IoC)

- Implementing IoC in Spring
- Implementing dependency injection via beans and bean factories
- Spring bean definition profiles and environments

More about Spring Bean Configuration

- Property editors
- Type converters
- Autowiring and component scanning
- Spring Expression Language
- Spring unified property management
- Bean definition profiles
- Caching

Java-Based Bean Configuration

- Using the @Configuration annotation
- Dependency injection in Java-based configuration
- Using Spring support for XML namespaces in Java-based configuration
- Accessing properties

Understanding the Application Context Lifecycle

- Bean factory post processing
- Bean post processing
- Implementing @PostConstruct and @PreDestroy methods
- Understanding dynamic proxies

Aspect-Oriented Programming (AOP)

- Spring AOP architecture
- Defining pointcuts
- Defining joinpoints
- Implementing advice methods
- Understanding pointcut designators
- Implementing introductions

Spring Data Access

- Spring data access concepts
- JdbcTemplate
- Spring repositories and application architecture
- JPA integration
- Hibernate integration
- Transaction management

Creating Web Applications with Spring Web MVC

- Overview of MVC
- Spring MVC implementation
- Configuring a dispatcher servlet
- Defining a controller
- Mapping request parameters
- Mapping path variables
- Accessing HTTP cookies, headers, and session state

RESTful Web Services

- Overview of RESTful services
- Creating a Spring RESTful service
- Additional mapping techniques
- Using HTTP POST to insert
- Using HTTP PUT to update
- Using HTTP DELETE to delete
- Implementing RESTful clients

Spring Data

- Introduction to Spring Data
- Using the API
- Worked examples

Introduction to Spring Boot

- Overview of Spring Boot
- Implementing application functionality
- Spring Boot Actuator