



SPRING CORE 5 EĞİTİMİ

4 GÜN



Digital Vizyon
Akademi

www.digitalvizyon.net



İçindekiler

Eğitim Hakkında.....	3
Neler Öğreneceksiniz?	3
Ön Koşullar	4
Kimler Katılmalı.....	4
Outline	4
Spring Overview	4
Java Configuration	4
More Java Configuration	4
Annotation and Component Scanning	5
Inside the Spring Container	5
Introducing Aspect-oriented programming.....	5
Testing a Spring-based Application.....	5
JDBC Simplification with JdbcTemplate	5
Transaction Management with Spring	5
Spring Boot Feature Introduction.....	5
Spring Boot – A closer look.....	6
Spring Boot – Spring Data JPA	6
Web Applications with Spring Boot.....	6
RESful Application with Spring Boot.....	6
Spring Boot Testing.....	6
Securing REST Application with Spring Security	6
Actuators, Metrics and Health Indicators.....	7

Eđitim Hakkında

Spring Core 5 Eđitimi, Java tabanlı bir açık kaynak kodlu uygulama çerçevesidir. Eđitimde, Spring Core'un nasıl kullanılacağına, projelerin nasıl yapılandırılacağına ve nasıl uygulanacağına dair bilgiler verilir.

Java uygulamalarının geliştirilmesinde Spring Core çerçevesi önemli bir rol oynar. Bu eğitim sürecinde, öğrenciler Spring Core'un nasıl yapılandırılacağını, nasıl kullanılacağını ve nasıl uygulanacağını öğrenirler. Bunun yanında, uygulama geliştirme sürecinde kullanabilecekleri diğer araçlar ve teknolojiler hakkında da bilgi sahibi olurlar.

Spring Core eğitimi, Java uygulama geliştiricileri için çok değerli bir kaynaktır. Eğitim, geliştiricilere kodlarını daha verimli ve güvenli hale getirme yöntemlerini sunar. Bu sayede, geliştiriciler, uygulamalarını daha etkili ve güvenilir bir şekilde geliştirebilirler.

Eđitim aynı zamanda, geliştiricilere Java platformunda uygulama geliştirme konusunda derin bir anlayış kazandırır. Bu derin anlayış, geliştiricilerin daha karmaşık ve ileri düzey Java uygulamalarını geliştirmelerine yardımcı olur. Bu nedenle, eğitim süreci boyunca edinilen bilgi ve beceriler, Java uygulama geliştiricileri için büyük bir avantaj sağlar.

Sonuç olarak, Spring Core eğitimi, Java tabanlı uygulamalar geliştirmek isteyenler için oldukça yararlıdır. Eğitim süreci, öğrencilere temel kavramları, çeşitli araçları ve teknolojileri öğretir. Ayrıca, geliştiricilerin kodlarını daha verimli ve güvenli hale getirebilmeleri için gerekli olan teknik bilgileri de sağlar.

Bu eğitim, geliştiricilere Java uygulamalarının nasıl daha etkili ve güvenli hale getirileceđi konusunda kapsamlı bilgiler sunar. Eğitimin sonunda, geliştiricilerin Java platformu üzerinde daha karmaşık ve ileri düzey uygulamalar geliştirebilecekleri konusunda kendilerine olan güvenleri artacaktır. Dolayısıyla, Spring Core 5 Eğitimi, Java uygulama geliştiricileri için değerli ve faydalı bir kaynaktır.

Neler Öğreneceksiniz?

Spring Core eğitimi sırasında, şunları öğrenebilirsiniz:

- Spring Core framework'ünün temel kavramlarını ve özelliklerini
- Bean yapılandırmasını ve yönetimini
- Dependency Injection (Bağımlılık Enjeksiyonu) ve Inversion of Control (Kontrol Tersine Alma) konseptlerini
- AOP (Aspect Oriented Programming – Yönünceli Programlama) ve çoklu modül uygulamalarını
- Spring Core'un veritabanı erişimi, güvenlik, yapılandırma ve diğer ilgili teknolojilerle nasıl entegre edilebileceđini öğreneceksiniz.

Bu konulara ek olarak, Spring Core 5'in özelleştirilmiş versiyonlarının nasıl oluşturulabileceđi ve uygulanabileceđi de görülebilir.



Ön Koşullar

Spring Core 5 eğitimine katılmak için genel olarak aşağıdaki ön koşullar bulunabilir:

- Java programlama dilleri hakkında temel bilgi
- Object Oriented Programming (OOP) kavramlarının anlaşılması
- Web uygulamaları ve MVC mimarisi hakkında bilgi

Bu ön koşullar, eğitim verilen şirket veya kurumun belirlemiş olduğu standartlar ve ihtiyaçlarına göre farklılık gösterebilir. Eğitim verilen yerde daha detaylı bilgi alabilirsiniz.

Kimler Katılmalı

Spring Core 5 eğitimi, aşağıdaki profesyonel gruplar için uygun olabilir:

- Java uygulamalarının geliştirilmesi ve bakımı ile ilgilenen yazılım geliştiricileri
- Enterprise Java uygulamalarının tasarımı ve yapılandırması ile ilgilenen yazılım mühendisleri
- Java tabanlı web uygulamalarının tasarımı ve geliştirilmesi ile ilgilenen web geliştiricileri
- Spring Framework ile ilgili projeler üzerinde çalışan yazılım takımları

Eğitim, Java ve web uygulamaları ile ilgisi olan herkes katılabilir. Ancak, temel Java ve OOP kavramlarının anlaşılması, bu eğitimi en iyi şekilde faydalanmanıza yardımcı olacaktır.

Outline

Spring Overview

- What is the Spring Framework?
- The DI Container
- The Spring Framework History and EcoSystem

Java Configuration

- Java configuration and the Spring application context
- @Configuration and @Bean annotations
- @Import: working with multiple configuration files
- Defining bean scopes
- Launching a Spring Application and obtaining Beans

More Java Configuration

- External properties & Property sources
- Environment abstraction
- Using bean profiles



- Spring Expression Language (SpEL)

Annotation and Component Scanning

- Component scanning
- Autowiring using `@Autowired`
- Java configuration versus annotations, mixing.
- Lifecycle annotations: `@PostConstruct` and `@PreDestroy`
- Stereotypes and meta-annotations

Inside the Spring Container

- The Spring Bean Lifecycle
- The `BeanFactoryPostProcessor` interception point
- The `BeanPostProcessor` interception point
- Spring Bean Proxies
- `@Bean` method return types

Introducing Aspect-oriented programming

- What problems does AOP solve?
- Defining pointcut expressions
- Implementing various types of advice

Testing a Spring-based Application

- Spring and Test-Driven Development
- Spring 5 integration testing with JUnit 5
- Application context caching and the `@DirtiesContext` annotation
- Profile selection with `@ActiveProfiles`
- Easy test data setup with `@Sql`

JDBC Simplification with `JdbcTemplate`

- How Spring integrates with existing data access technologies
- Spring's `JdbcTemplate`
- `DataAccessException` hierarchy

Transaction Management with Spring

- Transaction overview
- Transaction management with Spring
- Transaction propagation and rollback rules
- Transactions and integration testing

Spring Boot Feature Introduction

- Introduction to Spring Boot Features



- Value Proposition of Spring Boot
- Creating a simple Boot application using Spring Initializer website

Spring Boot – A closer look

- Dependency management using Spring Boot starters
- How auto-configuration works
- Configuration properties
- Overriding auto-configuration
- Using CommandLineRunner

Spring Boot – Spring Data JPA

- Quick introduction to ORM with JPA
- Benefits of using Spring with JPA
- JPA configuration in Spring
- Configuring Spring JPA using Spring Boot
- Spring Data JPA dynamic repositories

Web Applications with Spring Boot

- Introduction to Spring MVC and request processing
- Controller method signatures
- Using @Controller, @RestController and @GetMapping annotations
- Configuring Spring MVC with Spring Boot
- Spring Boot packaging options, JAR or WAR

RESful Application with Spring Boot

- An introduction to the REST architectural style
- Controlling HTTP response codes with @ResponseStatus
- Implementing REST with Spring MVC, @RequestMapping, @RequestBody and @ResponseBody
- Spring MVC's HttpMessageConverters and automatic content negotiation

Spring Boot Testing

- Spring Boot testing overview
- Integration testing using @SpringBootTest
- Web slice testing with MockMvc framework
- Slices to test different layers of the application

Securing REST Application with Spring Security

- What problems does Spring Security solve?
- Configuring authentication
- Implementing authorization by intercepting URLs



- Authorization at the Java method level
- Understanding the Spring Security filter chain
- Spring security testing

Actuators, Metrics and Health Indicators

- Exposing Spring Boot Actuator endpoints
- Custom Metrics
- Health Indicators
- Creating custom Health Indicators
- External monitoring systems