**📌 1. Spring Boot’a Genel Bakış (Overview)**

📝 Spring Boot, yapılandırma yerine varsayılan ayarları tercih eden bir Spring eklentisidir. Üretim düzeyinde bağımsız uygulamalar geliştirmenizi kolaylaştırır.)

**📌 2. Proje Kurulumu (Setup)**

📝 Spring Initializr kullanılarak proje başlatılır. spring-boot-starter-web, spring-boot-starter-data-jpa ve h2 bağımlılıkları eklenir.  
metin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

metin, ekran görüntüsü, menü içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**📌 3. Uygulama Yapılandırması (Application Configuration)**

📝 @SpringBootApplication ile uygulama başlatılır. application.properties dosyasında temel ayarlar (örn. server.port=8081) yapılır.  
metin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**📌 4. Basit Bir MVC Görünümü (Simple MVC View)**

📝 Thymeleaf ile frontend hazırlanır. Bir controller aracılığıyla kullanıcıya hoş geldin mesajı gösterilir.  
metin, ekran görüntüsü, yazılım, multimedya yazılımı içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

metin, ekran görüntüsü, yazılım, multimedya yazılımı içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**📌 5. Uygulama Güvenliği (Security)**

📝 spring-boot-starter-security ile temel güvenlik sağlanır. Özelleştirilmiş bir SecurityConfig sınıfı oluşturularak tüm erişimlere izin verilir.

metin, ekran görüntüsü, yazılım içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**📌 6. Basit Kalıcılık (Simple Persistence)**

📝 Book adında bir JPA entity tanımlanır. BookRepository ile veri erişimi sağlanır. H2 veritabanı otomatik yapılandırılır.  
metin, ekran görüntüsü, yazılım, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

metin, ekran görüntüsü, yazılım, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**📌 7. Web Katmanı ve Controller (Web and the Controller)**

📝 BookController sınıfı ile REST API üzerinden CRUD işlemleri gerçekleştirilir.  
metin, ekran görüntüsü, yazılım içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

metin, yazı tipi, ekran görüntüsü içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**📌 8. Hata Yönetimi (Error Handling)**

📝 @ControllerAdvice ile global hata yönetimi yapılır. Özel exception sınıfları tanımlanır (BookNotFoundException, BookIdMismatchException).

metin, ekran görüntüsü, yazılım, multimedya yazılımı içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

metin, ekran görüntüsü, yazı tipi, yazılım içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

metin, ekran görüntüsü, yazılım, multimedya yazılımı içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**📌 9. Test Senaryoları (Testing)**

📝 @SpringBootTest ile uygulama testi yapılır. REST Assured kullanılarak API uç noktaları test edilir.

metin, ekran görüntüsü içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**📌 10. Sonuç (Conclusion)**

📝 Bu eğitim ile Spring Boot'un temel işleyişi, yapılandırması, web ve veri katmanları tanıtılmıştır.