

JAVA API GELİŞTİRMESİ

Abdulkerim Atik

Senior Java Developer

Eğitimin İçeriği

- Java Web Service protokolleri
- Eclipse ve Mvn Kurulumu
- Spring Nedir?
- Spring Boot nedir ve avantajları nelerdir ?
- Hibarnate nedir ?
- Spring Security ?
- Envanter Uygulama Geliştirmesine başlama

Java Web Servis protokolleri

Öncelikle Web Servis Nedir ?

Verilerinizi tüm cihazlara göndermek istediğinizde devreye Web Service kavramı girer. Web Service ile platform bağımsız tüm cihazlara veri aktarımı gerçekleştirilir.

En Çok Kullanılan Web Servis Protokolleri

- REST
- SOAP

Rest ve Restful Kavramları

Rest, client-server arasındaki iletişimin HTTP protokolü sayesinde kolay ve hafif bir şekilde yapılmasını sağlayan bir mimaridir.

- **GET:** Veri listeleme - veri görüntülemek için kullanılır.
- **POST:** Veri eklemek için kullanılır.
- **PUT:** Veriyi Güncelleme isteği olarak kullanılır.
- **PATCH:** Verinin sadece bir parçasını güncellemek için kullanılır. Örneğin bir issue'nun durumunun aktiften çözüldü haline getirilmesi.
- **DELETE:** Veriyi silmek için kullanılır.
- **OPTIONS:** Bir api urline Options isteği yapıldığında o url in hangi istekleri kabul ettiği bilgisi verilir.

Rest Servislerin Cevap Kodları

- **200 OK:** Genelde veri listeleme sonuçları 200 ile dönüş yapılır.
- 201 CREATED:** Veri eklendiği zaman verinin kendisi ile 201 dönülüyor.
- 204 NO CONTENT:** Veri silindiği zaman 204 dönülüyor.
- 400 BAD REQUEST:** Genel olarak kayıt ekleme ya da güncelleme isteklerinde gönderilen veri validasyondan geçemediyse neden geçemediği hakkında bilgiyle beraber 400 http statusuyla dönülür.
- 403 Forbidden:** Yetkiye dayalı bir işlem yapılıyorsa bu api uç noktasında işlem yapmaya çalışan kişinin bu işlemi yapmaya yetkisi yoksa 403 status kodu döndürülür.
- 401 Unauthorized:** Api ucunuzda bu işlemi yapmak için login olmak zorunlu ise ve apiye istek yapan kullanıcı login değil ise bu http status ile cevap verilir. Örneğin kullanıcının kendi bilgilerinin güncellemesi denilebilir.
- 404 Not Found:** Bu http status kullanıcının istek yaptığı url yok ise ya da url deki veri geçersiz ise bu hatayı alırız.

Eclipse Ve Maven

Eclipse İndirmek için :<https://www.eclipse.org/downloads/>

En Çok Kullanılan Yapı Otomasyonları(Build Project)

- Maven
- Gradle
- Ant

Maven

Apache Maven çıkış noktası *build* işlemlerini sadeleştirmek olan bir **build otomasyon** aracıdır.

Maven ne işe yarar?

- Build işlemini kolaylaştırmak
- Sabit bir build sistemi sağlamak
- Kaliteli proje bilgisi sağlamak
- En iyi uygulamayı geliştirmek için kurallar sağlamak
- Yeni özelliklere geçişi kolaylaştırmak

POM nedir?

Project Object Model ya da POM, Maven'ın en önemli parçalarından biridir. Proje hakkında build işlemi için gerekli bilgileri ve konfigrasyon detaylarını içeren bir XML dosyasıdır. Maven bir görevi yerine getirmeden önce, ilk olarak çalışma dizininde POM'u bulur ve gerekli bilgileri okur.

Maven nasıl çalışır?

Maven'ın üç *build* yaşam döngüsü vardır

clean: Önceki build işlemi sonucu oluşan dosyaları siler

site: dokümantasyon oluşturur.

build: build yaşam döngüsünü ifade eder.

Spring Nedir

- ▶ **Spring**, Java ve .NET için geliştirilmiş açık kaynak kodlu bir framework'tür. JavaEE uygulamalarını geliştirmeyi kolaylaştırır. Spring'i "*frameworks of framework*" olarak düşünebiliriz çünkü Struts, Hibernate, Tapestry, EJB, JSF gibi frameworklerin kullanımını destekleyen bir frameworktur.
- ▶ <https://spring.io/guides>

Dependency Injection (Bağımlılık Enjeksiyonu)

- ▶ *Dependency Injection*, S.O.L.I.D prensiplerinin 5. ayağını oluşturan "*Dependency Inversion*" prensibinin uygulanmasını içeren bir patterndir.
- ▶ Dependency Injection temel olarak bağımlılıkların kontrolü ve yönetimi için kullanılmaktadır. Peki nedir bu bağımlılıklar?

S.O.L.I.D

- ▶ **SOLID** yazılım prensipleri; geliştirilen yazılımın esnek, yeniden kullanılabilir, sürdürülebilir ve anlaşılır olmasını sağlayan, kod tekrarını önleyen ve Robert C. Martin tarafından öne sürülen prensipler bütünüdür.
- ▶ **S – Single-responsibility principle**
- ▶ **ÖZET:** Bir sınıf (nesne) yalnızca bir amaç uğruna değiştirilebilir, o da o sınıfa yüklenen sorumluluktur, yani bir sınıfın(fonksiyona da indirgenebilir) yapması gereken yalnızca bir işi olması gerekir.
- ▶ **O – Open-closed principle**
- ▶ **ÖZET:** Bir sınıf ya da fonksiyon halihazırda var olan özellikleri korumalı ve değişikliğe izin vermemelidir. Yani davranışını değiştirmiyor olmalı ve yeni özellikler kazanabiliyor olmalıdır.
- ▶ **L – Liskov substitution principle**
- ▶ **ÖZET:** Kodlarımızda herhangi bir değişiklik yapmaya gerek duymadan alt sınıfları, türedikleri(üst) sınıfların yerine kullanabilmeliyiz.
- ▶ **I – Interface segregation principle**
- ▶ **ÖZET:** Sorumlulukların hepsini tek bir arayüze toplamak yerine daha özelleştirilmiş birden fazla arayüz oluşturmaliyiz.
- ▶ **D – Dependency Inversion Principle**
- ▶ **ÖZET:** Sınıflar arası bağımlılıklar olabildiğince az olmalıdır özellikle üst seviye sınıflar alt seviye sınıflara bağımlı olmamalıdır.

SpringBoot Nedir

- ▶ **Spring Boot** bağımsız(stand-alone) Spring tabanlı uygulamalar geliştirmenizi kolaylaştırmaya yarar. **Spring Boot**'u bu kadar popüler yapan şey ise otomatik konfigürasyon özelliği. Çok az Spring konfigürasyonu ile çoğu **Spring Boot** projesini hayata geçirebilirsiniz.

Spring Boot'un bazı temel özellikleri

- ▶ Bağımsız(stand-alone) Spring uygulamaları oluşturabilmek.
- ▶ Gömülü bir web sunucusu(Tomcat, Jetty, Undertow) ile gelmesi.
- ▶ Build konfigürasyonunu kolaylaştırmak için sağladığı starter'lar.
- ▶ Otomatik konfigürasyon.
- ▶ Kod üretimi(code generation) ve XML konfigürasyona ihtiyaç duymaması.

Hibernate Nedir?

- Hibernate Java geliştiriciler için geliştirilmiş bir ORM kütüphanesidir. Nesne yönelimli modellere göre veritabanı ile olan ilişkiyi sağlayarak, veritabanı üzerinde yapılan işlemleri kolaylaştırmakla birlikte kurulan yapıyı da sağlamlaştırmaktadır.

Spring Security nedir?

- Spring framework kullanılarak geliştirilen doğrulama(Authentication), yetkilendirme(Authorization), şifreleme>Password Encoder) ve CSRF gibi güvenlik önlemleri sağlayan, Spring platformunda yer alan bir projedir.

Neden Spring Security

Kullanıcılar uygulamalardan beklenen işlevin yanı sıra kullanıcıların sahip olduğu yetkilere(yönetim, editör veya üye) göre işlem yapması istenebilir.

Uygulama bu ihtiyaç ile birlikte doğrulama, yetkilendirme ve saldırganın yetkileri elde ederek uygulama işlevini değiştirmesini önleme gibi güvenlik tedbirlerine gereksinim duyulur.