**17.03.2023**

İlgili günde Spring Boot ile Hibernate ORM kullanarak MySQL veritabanına belli bir değerin kaydettirilmesi üzerine araştırma ve çalışmalar yapılmıştır. Bu çalışmalar sonucunda bir örnek uygulamanın geliştirilmesine başlanmıştır. Gün içerisinde uygulamanın Entity ve Repository katmanları geliştirilmiştir.

Hibernate ile veritabanı üzerinde kayıt işlemi yapmak için birden fazla metot bulunmaktadır. Bunlar save, persist, update, merge, saveOrUpdate metotlarıdır. Bu metotların her biri arka planda kendine özgü bir SQL sorgusunu çalıştırmaktadır.

**save():** Tabloya kayıt eklemek için kullanılır. SQL dilinde INSERT INTO komutunu çalıştırır.

**persist():** Bellekte geçici olarak tutulan bir değerin kalıcı statüsüne geçmesini ve kaydedilmesini sağlar. Ancak bu kayıt işleminin hemen gerçekleşeceği taahhüt edilmez.

**update():** Var olan bir kaydı güncellemek için kullanılır.

**saveOrUpdate():** İlgili nesne veritabanında mevcutsa üzerinde update gerçekleştirir, yoksa kaydeder.

**merge() :** Değeri güncellenmiş bir nesnenin, kayıtlı olan eski haliyle birleştirilmesini sağlar.

Örnek olarak yapılan çalışmada kullanıcı uygulamadan “/kaydet” alt adresi ile bir istekte bulunduğunda parametre olarak gelen isim ve soyisim bilgilerinin veritabanına kaydedilmesi hedeflenmiştir. Spring Boot – Hibernate entegrasyonu için bir önceki gün application.properties dosyası üzerinde yapılan değişikliklerin aynısı uygulanmıştır. Bu dosyada yalnızca veritabanının ismi “userdb” olarak değiştirilmiştir.

Veritabanında tutulacak olan kişileri nesne olarak temsil edecek olan Entity dosyası “User” isminde bir sınıf olarak oluşturulmuştur.

@Entity

@Table(name = "users")

public class User {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private int id;

@Column(name = "Name")

private String name;

@Column(name = "Surname")

private String surname;

public int getId() { return id; }

public void setId(int id) { this.id = id; }

public String getName() { return name; }

public void setName(String name) { this.name = name; }

public String getSurname() { return surname; }

public void setSurname(String surname) { this.surname = surname; }

}

Entity katmanı ile Service katmanı arasında köprü görevi olacak olan UserRepository ismindeki Repository katmanı interface olarak oluşturulmuştur. Bu interface CRUD işlemlerinin gerçekleştirilebilmesi için CrudRepository sınıfından kalıtım almaktadır. Gerekli tanımlama aşağıdaki gibi yapılmıştır.

@Repository

public interface UserRepository extends CrudRepository<User, Integer> {}