

Full Stack Veterinerian Application

Projemizde Spring, Thymeleaf, JPA, Maven ve MySQL teknolojilerini kullandık. Proje dosyamız oluşturulduktan sonra MySQL in kurulumu tamamlandı. Önce projemizin MySQL veritabanı ile configürasyonunu yaparak bağlantı oluşturduk. Bu işlemi application.properties dosyasında gerçekleştirdik.

```
spring.datasource.url=jdbc:mysql://localhost:2644/vet?useSSL=false
spring.datasource.username=root
spring.datasource.password=root
```

```
spring.datasource.driver-class-name=com.mysql.jdbc.Driver
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL5InnoDBDialect
```

```
spring.jpa.hibernate.ddl-auto=update
```

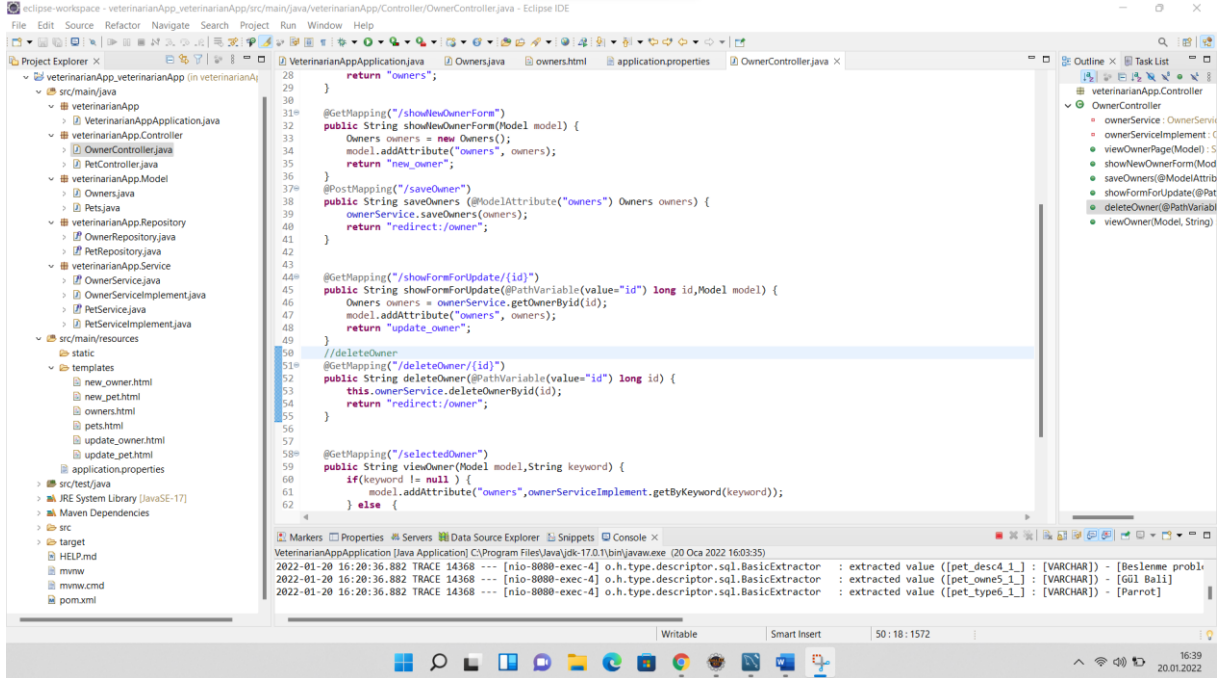
```
logging.level.org.hibernate.SQL=DEBUG
logging.level.org.hibernate.type=TRACE
```

Projemizde MVC yapısını kullanacağımız için Model ,View(templates), Controller ve Service paketlerimizi oluşturduk. Veri tabanında hayvanlar için Pets , sahipleri için Owner olmak üzere 2 adet table tanımlama işlemini model paketimizin içinde gerçekleştirdik. Table lar oluşturulup gerekli column lar tanımlandı.

MySQL Workbench interface showing the 'vet' schema. The 'owners' table is selected, and the SQL query 'SELECT * FROM owners o WHERE o.owner_fullname LIKE '%Seda%' is executed. The result grid shows 1 row returned for 'Seda Bali' with pet 'birman kedisi (pet id: 1)'. The output window shows the execution log with 10 actions, including table creation and data retrieval.

#	Time	Action	Message
5	16:15:56	DROP TABLE 'vet'.user	0 row(s) affected
6	16:16:00	DROP TABLE 'vet'.role	0 row(s) affected
7	16:16:02	SELECT * FROM owners o WHERE o.owner_fullname LIKE '%Seda%' LIMIT 0, 1000	1 row(s) returned
8	16:16:04	SELECT * FROM owners o WHERE o.owner_fullname LIKE '%Seda%' LIMIT 0, 1000	1 row(s) returned
9	16:16:05	SELECT * FROM owners o WHERE o.owner_fullname LIKE '%Seda%' LIMIT 0, 1000	1 row(s) returned
10	16:16:26	SELECT * FROM owners o WHERE o.owner_fullname LIKE '%Seda%' LIMIT 0, 1000	1 row(s) returned

Veri tabanı üzerinde kullanmak üzere gerekli fonksiyonlar tanımlandı. Bunlar create , read,update ,delete , get by id , get all ve get by keyword (filter) işlemleri için controller bölümünde front-end bölümü ile ilişkilendirildi.Örneğin , var olan bir kaydı silmek istediğimiz zaman “delete” butonuna bastığımız zaman delete butonu `@GetMapping("/deleteOwner/{id}")` üzerinden delete fonksiyonu aktif ederek seçtiğimiz işlemi gerçekleştirdi.



Front end kısmında , thymeleaf ve bootstrap kullanarak pets and owners safları oluşturuldu. Sayfalarımız aşağıdaki bağlantılara tanımlandı.

`@GetMapping("/owner")` `@GetMapping("/pets")`

Projemizi bir java app olarak run ettiğimiz zaman <http://localhost:8080/pets> veya `owner` üzerinden pets sayfasına erişim tanımlandı.

Pets ve owners sayfalarının ekran görüntüleri alttaki gibidir.

Owner List

Pet Owner Name

Search

Add New Pet Owner

Go to Pet List

id	Owner Full Name	Owner's Pets	Phone	Email	Address	Edit
1	Sedat Bali	birman kedisi (pet id:1)	5551102644	sedatbali44@yahoo.com	çamlıklı sk. Ataşehir/istanbul	Update Delete
2	Salih Akgölan	Bulldog (pet id:2)	5536832644	salih2003@outlook.com	Doğanşehir / Malatya	Update Delete
7	Serkan Arda	kanarya (pet id:47)	5387664588	serkan58@gmail.com	Odunpazarı/Eskişehir	Update Delete
9	Turan E. Şimşek	tavşan (pet id:33)	5397674827	turan27@gmail.com	Şahinbey/Gaziantep	Update Delete

Pet List

Pet type

Search

Add New Pet

Go to Owner List

Pet id	Owner	Type	Breed	Age	Description	Edit
1	Sedat Bali	cat	birman	3	cilt alerjisi (tedavi son seansda)	Update Delete
2	Musa Ak	dog	bulldog	5	aşları eksik(Leptospirosis yapılacak)	Update Delete
4	Salih Ayrar	horse	arabic	2	Bornaschen hastalığı	Update Delete
6	yigit demir	canaria	crest	4 aylık	sindirim rahatsızlığı	Update Delete
7	Hasan Kara	chicken	sussex	6 aylık	ayak yarası	Update Delete
9	Gül Bali	Parrot	Afrikan Grey	3	Beslenme problemi	Update Delete

Pet Update

Update Pet

Save Pet

[Back to Pet List](#)

Pet Register

Save new Pet

Save Pet

[Back to Pet List](#)