



## E-CommerceApp

E-CommerceApp, basit bir e-ticaret uygulamasının backendi için Spring Boot ile geliştirilen bir REST API dir.

### Proje Mimarisi:

#### 1-Entity ve ilişkiler

##### Product:

**id:** Long, otomatik generate

**productName:**String, space,null veya empty olmamalı, unique olmalı

**brand:**String, space,null veya empty olmamalı

**price:**Double, null olmamalı

##### Customer:

**id:** Long, otomatik generate

**name:** String, space,null veya empty olmamalı

**lastName:** String, space,null veya empty olmamalı

**email:** String, space,null veya empty olmamalı,unique ve email formatında olmalı

**phone:**String

**orders:**Set<OrderItem>

**One\_Customer <-> Many\_OrderItem**

##### OrderItem:

**id:** Long, otomatik generate

**quantity:**Integer,null olmamalı

**totalPrice:**Double,null olmamalı, persist ve updateden önce otomatik hesaplanır.

**product:**Product, null olmamalı

**customer:**Customer, null olmamalı

**Many\_OrderItem <->One\_Product , Many\_OrderItem <->One\_Customer**

### Endpointler:

## Customer:

1-customer oluşturma/kaydetme -> <http://localhost:8080/customers/save>

//email daha önce kullanılmışsa hata fırlatır.(ConflictException)

2-Tüm customerları getirme -> <http://localhost:8080/customers>

3-Id ile tek bir customer getirme -> <http://localhost:8080/customers/1>

//id tabloda yoksa hata fırlatır.(ResourceNotFoundException)

4-Id ile bir customerı silme -> <http://localhost:8080/customers/custom?id=1>

//Customer is deleted successfully mesajı döner.

//Customer silinirse, siparişleri de silinir

5-id ile customer ı update etme -> <http://localhost:8080/customers/update/1>

//Customer is updated successfully mesajı dönsün.

//emaili update ederken yeni değer tabloda var ve kendi maili değilse hata fırlatır. (ConflictException)

6-tüm customerları page page gösterme -> <http://localhost:8080/customers/page?page=1&size=2&sort=id&direction=ASC>

7-Name ile customer getirme -> <http://localhost:8080/customers/query?name=Jack>

8-fullname ile customer getirme-> <http://localhost:8080/customers/fullquery?name=Jack&lastName=Sparrow>

9-İsmi ... içeren customerlar -> <http://localhost:8080/customers/jpql?name=Ja>

10-Idsi verilen müşterinin tüm siparişlerini getirme -> <http://localhost:8080/customers/allorder/1>

11-ÖDEV:Requestle gelen "harf dizisi" name veya lastname inde geçen customerları döndür.

-> <http://localhost:8080/customers/search?word=pa>

ÖDEV:5 tane customer oluşturup ekleyiniz.

## Product:

1-product ekleme -><http://localhost:8080/products/add>

//isim daha önce kullanılmışsa hata fırlatır.(ConflictException)

2-Tüm productları getirme -><http://localhost:8080/products>

3-Id ile product getirme -><http://localhost:8080/products/5>

4-tüm productları page page gösterme -><http://localhost:8080/products/page?page=1&size=2&sort=id&direction=ASC>

ÖDEV:5 tane ürün oluşturup ekleyiniz.

### Order:

1-sipariş oluşturma -><http://localhost:8080/orders/save/filter?cid=1&prod=1&quant=3>

//farklı üründe yeni sipariş, aynı üründe sayı artırılır

2-tüm siparişleri getirme -><http://localhost:8080/orders>

3-Id ile sipariş getirme -><http://localhost:8080/orders/5>

4-Id ile sipariş miktarını update etme -><http://localhost:8080/orders/update/5/quantity/30>

//quantity=0 ise siparişi sil

5-Id ile sipariş delete etme -><http://localhost:8080/orders/delete?id=5>

ÖDEV:-tüm siparişleri page page gösterme-> [http://localhost:8080/orders/page?page=1  
&size=&sort=id&direction=ASC](http://localhost:8080/orders/page?page=1&size=&sort=id&direction=ASC)

ÖDEV:5 tane sipariş oluşturup ekleyiniz.