ONE TO MANY

Hüseyin Demirel
github.com/HsynDmrl

costumers

PK	id
	name
	surname

orders

PK	id
FK	costumer_id
	pick_up_date
	drop_date

```java
...

@Entity
@Table(name="costumers")
public class Costumer {

    @Id
    @Column(name="id")
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;

    @Column(name="name")
    private String name;

    @Column(name="surname")
    private String surname;

    @OneToMany(mappedBy="costumer")
    private List<Order> orders;
}
```

```java
...

@Entity
@Table(name="orders")
public class Order {

    @Id
    @Column(name="id")
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;

    @ManyToOne
    @JoinColumn(name="costumer_id")
    private Costumer costumer;

    @Column(name="pick_up_date", columnDefinition = "date")
    private LocalDate pickUpDate;

    @Column(name="drop_date", columnDefinition = "date")
    private LocalDate dropDate;
}
```

```java
@OneToMany(mappedBy="costumer")
private List<Order> orders;
```

```java
@ManyToOne
@JoinColumn(name="costumer_id")
private Costumer costumer;
```

costumers

PK	<u>id</u>
	name
	surname
FK	address_detail_id

address_details

PK	<u>id</u>
	address
	city
	postal_code

```java
...

@Entity
@Table(name = "customers")
public class Customer {

    @Id
    @Column(name="id")
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;

    @Column(name="name")
    private String name;

    @Column(name="surname")
    private String surname;

    @OneToOne(mappedBy = "customer", cascade = CascadeType.ALL)
    private AddressDetail addressDetail;
}
```

```java
...

@Entity
@Table(name = "address_details")
public class AddressDetail {

    @Id
    @Column(name="id")
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;

    @Column(name="address")
    private String address;

    @Column(name="city")
    private String city;

    @Column(name="postal_code")
    private String postalCode;

    @OneToOne
    @JoinColumn(name = "address_detail_id", referencedColumnName = "id")
    private AddressDetail addressDetail;
}
```

```java
@OneToOne(mappedBy = "customer", cascade = CascadeType.ALL)
private Address address;
```

```java
@OneToOne
@JoinColumn(name = "address_detail_id", referencedColumnName = "id")
private AddressDetail addressDetail;
```

"cascade = CascadeType.ALL" Costumer nesnesi üzerinde yapılan herhangi bir değişiklik (ekleme, güncelleme, silme gibi) aynı değişikliğin AddressDetail nesnesine de otomatik olarak yansıtılmasını sağlar. Bu sayede, ilişkili nesneler arasındaki tutarlılığı korumak ve yönetmek daha <u>kolay</u> hale gelir.

Burada `referencedColumnName = "id"` ifadesi, `addressDetail` alanının `AddressDetail` sınıfındaki `id` sütunuyla eşleşeceğini belirtir. Yani, `address_detail_id` sütunu `id` sütunuyla ilişkilendirilmiş olur. Bu, iki sınıf arasındaki ilişkinin bu iki sütun arasında kurulduğunu belirtir. Eğer `id` sütun adları aynı olsaydı, `referencedColumnName` belirtilmemize gerek <u>olmazdı</u>.