One To One Mapping:

Example 1: One-to-one association that maps a foreign key column

```
public class Person
{
            @Column(name=”P_ID”)
            private int personId;
            private String name;
            private int mobileno;
            @OneToOne(fetch = FetchType.LAZY, mappedBy = "person", cascade = CascadeType.ALL, optional=false,)
            private PanCard pancard;

            //getter & setter
}
Public class PanCard
{
             private String panno;
             private fatherName;

            @OneToOne(fetch = FetchType.LAZY, optional=false)
            @JoinColumn(name="P_ID", unique=true, nullable=false, updatable=false)
            private Person person;

}
```

One To Many Mapping:

Example 1: One-to-Many association using generics

```
public class Person
{
            @Column(name=”P_ID”)
            private int personId;
            private String name;
            private int mobileno;
            @OneToMany(fetch = FetchType.LAZY, mappedBy = "person", cascade = CascadeType.ALL, optional=false,)
            private List<Order> order;

            //getter & setter
}
Public class Order
{
             private String panno;
             private fatherName;

            @ManyToOne(fetch = FetchType.LAZY, optional=false)
            @JoinColumn(name="P_ID", unique=true, nullable=false, updatable=false)
            private Person person;

}
```