

@Hibernate ManyToMany Annotasyonu Kullanm

```
1 public class Belge {
2     @Id
3     @GeneratedValue(strategy = GenerationType.IDENTITY)
4     @Column(name = "BelgeKayitId")
5     private int id;
6
7     @ManyToMany(cascade = CascadeType.ALL)
8     @JoinTable(
9         name="Belge_FirmaOzellik",
10        joinColumns = @JoinColumn( name="BelgeID"),
11        inverseJoinColumns = @JoinColumn( name="FirmaID")
12    )
13    private List<FirmaOzellikBilgileri> firmaOzellikBilgileri;
14 }
15 public class FirmaOzellikBilgileri {
16     @Id
17     @GeneratedValue(strategy = GenerationType.IDENTITY)
18     private int firmaId;
19 }
```

@JoinTable annotasyonunda **joinColumns** ksm **FirmaOzellikBilgileri** entity'nin **DEL**, bulundu entity'in yani **Belge** entity'nin **PRIMARY KEY** deerini, yani **id** deerini temsil eder. id deerleri veritabannda **Belge** isimli tablonun **BelgeKayitId** isimli sütunda tutular. Fakat @JoinColumn(name="BelgeID") diyerek, bu id deerinin, **Belge_FirmaOzellik** tablosunun **BelgeID** isimli sütunda tutulmasn salarz. @JoinColumn(name="BelgeID") yerine istediimiz sütün adn verebiliriz: @JoinColumn(name="adfsfmsdfmsaf"). Bu ekilde kullanld zaman **Belge** tablosunun primary key deerleri yani id deerleri **Belge_FirmaOzellik** tablosundaki adfsfmsdfmsaf isimli sütunda tutulmu olur.

Ayn mantkla `inverseJoinColumns = @JoinColumn(name="FirmaID")` **Belge_FirmaOzellik** tablosunda **FirmaOzellikBilgileri** entity'nin **primary key** deerlerinin yani id deerlerin **FirmaID** isimli sütunda tutulacan ifade eder.