

Kategori Yönetimi - 1 (Çok-Çok İlişkiler İçin Ara Tablonun Oluşturulması)

4 Mart 2020 Çarşamba 23:03

Öncelikli Sistem



Kaynaklar

- <https://www.microsoft.com/tr-tr/azure/monitoring/relationships?tab=relationships&source=7&type=relationships&view=relationships>
- <https://www.jetbrains.com/idea/doc/faq/faq-relationships.html>
- <https://www.jetbrains.com/idea/doc/faq/faq-relationships.html>
- <https://www.jetbrains.com/idea/doc/faq/faq-relationships.html>

Entity Framework Core'da Çok-Çok İlişkiler Maalesef Biraz Sıkıntılı :(

Eski EF'de daha kolaydı. Lütfen Karşılaştığınız tablolara bakın: <https://aka.ms/efcore-docs-relationships>

Çok-Çok İlişki Nedir?

```

class Program
{
    static void Main()
    {
        // ...
    }
}

class User
{
    public int Id { get; set; }
    public string Name { get; set; }
    public string Surname { get; set; }
}

class Post
{
    public int Id { get; set; }
    public string Title { get; set; }
    public string Content { get; set; }
    public int UserId { get; set; }
}

class Comment
{
    public int Id { get; set; }
    public string Content { get; set; }
    public int PostId { get; set; }
    public int UserId { get; set; }
}
    
```

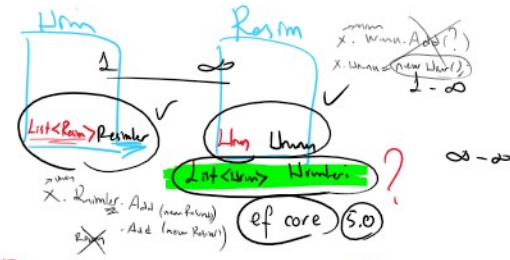
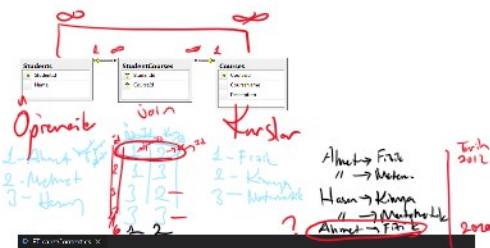
```

class Program
{
    static void Main()
    {
        // ...
    }
}

class User
{
    public int Id { get; set; }
    public string Name { get; set; }
    public string Surname { get; set; }
}

class Post
{
    public int Id { get; set; }
    public string Title { get; set; }
    public string Content { get; set; }
    public int UserId { get; set; }
}

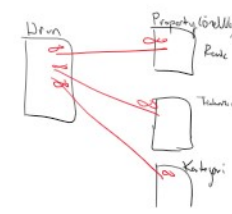
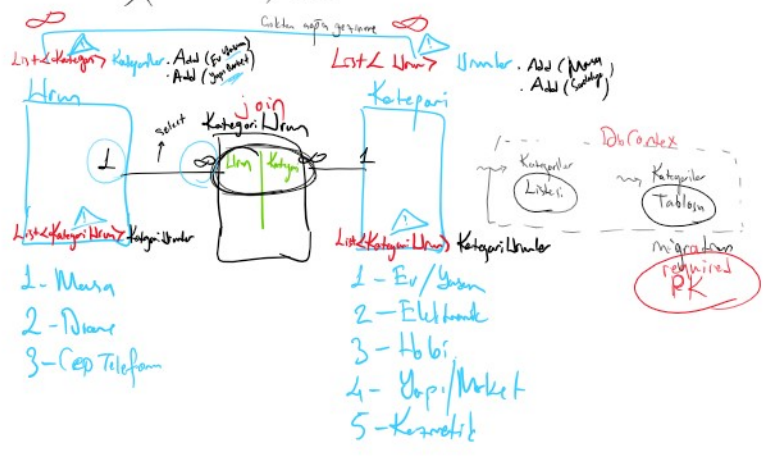
class Comment
{
    public int Id { get; set; }
    public string Content { get; set; }
    public int PostId { get; set; }
    public int UserId { get; set; }
}
    
```



```

public class ApplicationDbContext : DbContext
{
    public DbSet<User> Users { get; set; }
    public DbSet<Post> Posts { get; set; }
    public DbSet<Comment> Comments { get; set; }

    protected override void OnModelCreating(ModelBuilder modelBuilder)
    {
        modelBuilder.Entity<User>().Property(p => p.Id).HasKey();
        modelBuilder.Entity<Post>().Property(p => p.Id).HasKey();
        modelBuilder.Entity<Comment>().Property(c => c.Id).HasKey();
        modelBuilder.Entity<Comment>().Property(c => c.PostId).HasForeignKey().IsRequired();
        modelBuilder.Entity<Comment>().Property(c => c.UserId).HasForeignKey().IsRequired();
    }
}
    
```



```

Models > Urunca > {} Elicaret.Models
1 using System;
2 using System.ComponentModel.DataAnnotations;
3 using System.ComponentModel.DataAnnotations.Schema;
4 using Microsoft.AspNetCore.Mvc;
5 using System.Collections.Generic;
6
7 namespace Elicaret.Models
8 {
9     14 references
10     public class Urun //tekil
11     {
12         13 references
13         public Urun()
14         {
15             Resimler = new List<Resim>();
16         }
17         [key]
18         //TCKimlikNo
19         public int Id { get; set; }
20         [Display(Name="Adi")]
21         [Required(ErrorMessage="0) alani bos gicilsevidir")]
22         public string Ad { get; set; }
23         [Display(Name="Aciklama")]
24         [Required(ErrorMessage="0) alani bos gicilsevidir")]
25         public string Aciklama { get; set; }
26         [Display(Name="Fiyat")]
27         [Required(ErrorMessage="0) alani bos gicilsevidir")]
28         public decimal Fiyat { get; set; }
29         [NotMapped] //veritabanina yansimamasini saglar
30         //-----Istisnalar-----
31         4 references
32         public IReadOnlyList<Resim> Resimler { get; set; }
33         24 references
34         public List<KategoriUrun> KategoriUrunlari { get; set; }
35     }
36 }

```

```

Models > Kategorica > ...
1 using System.Collections.Generic;
2
3 namespace Elicaret.Models
4 {
5     29 references
6     public class Kategori
7     {
8         13 references
9         public int Id { get; set; }
10        21 references
11        public string Adi { get; set; }
12        12 references
13        public string Aciklama { get; set; }
14        //Istisnalar
15        18
16        public List<KategoriUrun> KategoriUrunlari { get; set; }
17    }
18 }

```

```

Models > Kategorilunca > {} Elicaret.Models
1 namespace Elicaret.Models
2 {
3     1 reference
4     public class KategoriUrun
5     {
6         1 reference
7         public int UrunId { get; set; }
8         1 reference
9         public Urun Urun { get; set; }
10        1 reference
11        public int KategoriId { get; set; }
12        4 references
13        public Kategori Kategori { get; set; }
14    }
15 }

```

- Mevcut migration'ları sil
- Mevcut veritabanını sil
- dotnet tool install -g dotnet-ef
- dotnet ef migrations add Yapilandi
- dotnet ef database update

