

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

آموزش پیکر بندی رابطه ی یک به چند

مدرس: مهندس افشین رفوآ

آموزش پیکر بندی رابطه ی یک به چند

خوب ما قصد داریم یک رابطه یک به چند بین کلاس های **Student** و **Standard** را پیاده سازی کنیم همان طور که بسیاری از دانشجویان در یک استاندارد تحصیل می کنند.

پیاده سازی رابطه یک به چند با استفاده از **DataAnnotation**

کلاس **Student** یک مشخصه ارجاعی از کلاس **Standard** با مشخصه کلید خارجی **StandardId** دارد و کلاس **Standard** یک مشخصه **collection** برای دانشجویان دارد. بنابراین این **DataAnnotation** با نتیجه رابطه یک به چند حاصل می شود.

```
public class Student
{
    public Student() { }
    public int StudentId { get; set; }
    [Required]
    public string StudentName { get; set; }
    public int StdandardId { get; set; }
    public virtual Standard Standard { get; set; }
}
public class Standard
{
    public Standard()
    {
        StudentsList = new List<Student>();
    }
    public int StandardId { get; set; }
    public string StandardName { get; set; }
    public string Description { get; set; }
    public virtual ICollection Students { get; set; }
}
```

پیکر بندی رابطه یک به چند با استفاده از **Fluent API**

فرض می کنیم که کلاس **Student** و کلاس **Standard** از قوانین **Code-First** پیروی نمی کنند و اسم های متفاوتی برای مشخصه ها دارند.

```

public class Student
{
    public Student() { }
    public int StudentId { get; set; }
    [Required]
    public string StudentName { get; set; }
    //StdId is not following code first conventions
name
    public int StdId { get; set; }
    public virtual Standard Standard { get; set; }
}
public class Standard
{
    public Standard()
    {
        StudentsList = new List<Student>();
    }
    public int StandardId { get; set; }
    public string StandardName { get; set; }
    public string Description { get; set; }
    public virtual ICollection StudentsList { get;
set; }
}
}

```

بنابراین شما با استفاده از **Fluent API** می‌توانید یک رابطه یک به چند را بین کلاس‌های **Student** و **Standard** پیاده‌سازی کنید.

```

protected override void OnModelCreating(DbModelBuilder
modelBuilder)
{
    //one-to-many
    modelBuilder.Entity().HasRequired(s =>
s.Standard)
        .WithMany(s =>
s.StudentsList).HasForeignKey(s => s.StdId);
}

```

و روش دیگری هم به شکل زیر وجود دارد.

```

protected override void OnModelCreating(DbModelBuilder
modelBuilder)
{
    //one-to-many
    modelBuilder.Entity().HasMany(s =>

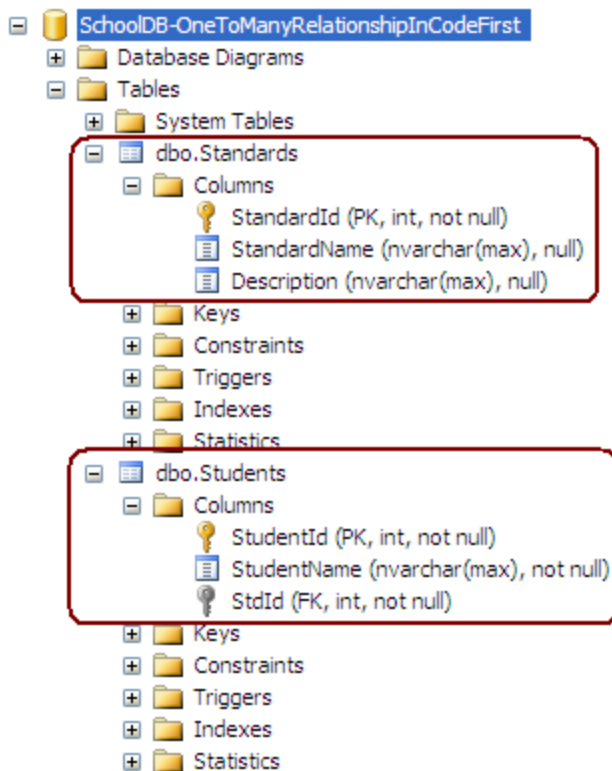
```

```

s.StudentsList)
        .WithRequired(s =>
s.Standard).HasForeignKey(s => s.StdId);
    }

```

کد بالا نشان داده شده بالا پایگاه داده زیر را ایجاد می کند.



مدل داده در **designer** به شکل دیاگرام پایین نشان داده می شود.

