

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

بهبود اعتبارسنجی داده ها

مدرس : مهندس افشین رفوآ

دوره آموزش MVC

بهبود اعتبارسنجی داده ها

افزودن data annotations

همان طور که در مبحث قبلی مشاهده کردید، برخی از قوانین اعتبارسنجی به صورت خودکار به ورودی کاربر اعمال می شود. برای مثال، فقط یک عدد می توان در فیلد **Grade** درج کرد. جهت تعریف قوانین اعتبارسنجی اضافه بر آنچه قبلاً تعریف کرده ایم، می توان **data annotations** به کلاس **model** اضافه کرد. این **annotation** ها در کل برنامه ی تحت وب شما برای فیلد یا **property** مشخص شده اعمال می شوند. شما همچنین می توانید **formatting attribute** (خصیصه های فرمت بندی) که قادر اند نحوه ی نمایش فیلدها را مدیریت کنند (همچون اصلاح مقدار مورد استفاده ی **text label** ها)، اعمال نمایید.

در آموزش حاضر، با افزودن **data annotations**، طول مقادیری (تعداد کاراکترهای مجازی) که برای **property** های **FirstName**، **LastName** و **MiddleName** ارائه می شود را محدود خواهیم ساخت. در پایگاه داده ی مورد نظر، این مقادیر محدود به 50 کاراکتر هستند، اما در حال حاضر این محدودیت به برنامه ی تحت وب اعمال نشده است. چنانچه کاربری مقداری را بیش از 50 کاراکتر بدهد، در آن صورت صفحه در زمان ذخیره ی مقدار در پایگاه داده از کار خواهد افتاد. همچنین **Grade** را به مقادیری بین 0 تا 4 محدود خواهیم ساخت. فایل **Student.cs** را در پوشه ی **Models** باز کنید. کدهای یابایت شده ی زیر را به کلاس مورد نظر اضافه کنید.

```
namespace ContosoSite.Models
{
    using System;
    using System.Collections.Generic;
```

```

using System.ComponentModel.DataAnnotations;
public partial class Student
{
    public Student()
    {
        this.Enrollments = new HashSet<Enrollment>();
    }
    public int StudentID { get; set; }
    [StringLength(50)]
    public string LastName { get; set; }
    [StringLength(50)]
    public string FirstName { get; set; }
    public Nullable<System.DateTime> EnrollmentDate { get; set; }
    [StringLength(50)]
    public string MiddleName { get; set; }
    public virtual ICollection<Enrollment> Enrollments { get; set; }
}
}

```

کدهای بالا به شما کمک می‌کند تا در فایل **Enrollment.cs** جای گذاری کنید.

```

namespace ContosoSite.Models
{
    using System;
    using System.Collections.Generic;
    using System.ComponentModel.DataAnnotations;
    public partial class Enrollment
    {
        public int EnrollmentID { get; set; }
        [Range(0, 4)]
        public Nullable<decimal> Grade { get; set; }
        public int CourseID { get; set; }
        public int StudentID { get; set; }
        public virtual Course Course { get; set; }
        public virtual Student Student { get; set; }
    }
}

```

**Solution** را کامپایل کنید.

به صفحه‌ی مربوط به ویرایش مشخصات دانشجو (**edit**) و ایجاد دانشجوی جدید (**create**) پیام‌ها را تغییر دهید. چنانچه در یکی از فیلدها مقداری بیش از 50 کاراکتر وارد کنید، با پیام خطا مواجه خواهید شد.

# Edit

## Student

LastName  × The field LastName must be a string with a maximum length of 50.

FirstName

به صفحه ی **Edit Enrollment** مراجعه کرده و عددی بزرگتر از 4 را در فیلد **Grade** وارد نمایید.

# Edit

## Enrollment

Grade  The field Grade must be between 0 and 4.

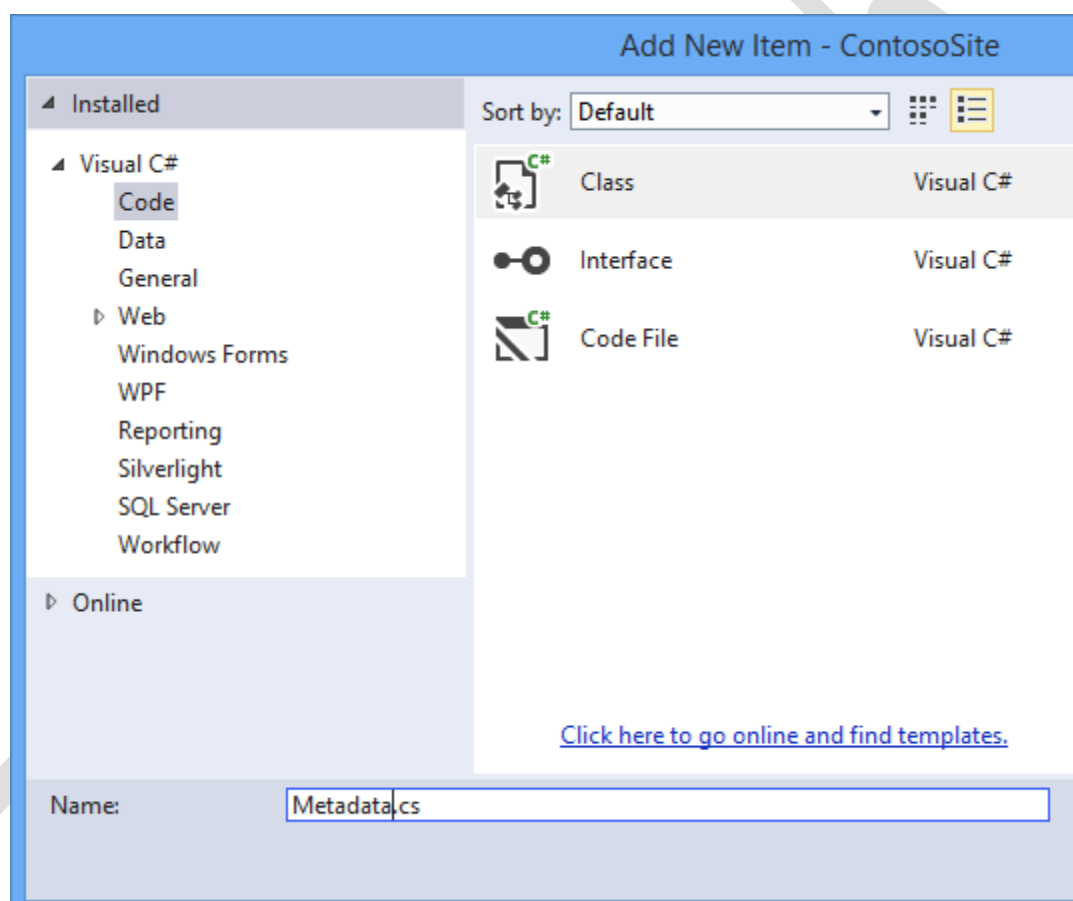
CourseID

StudentID

## افزودن کلاس های metadata

افزودن **validation attribute** ها مستقیماً به کلاس **model** زمانی کارساز خواهد بود که شما انتظار ندارید، پایگاه داده تغییر یابد. با این وجود اگر پایگاه داده ی شما تغییر پیدا کرد و شما مجبور به ایجاد مجدد کلاس **model** شدید، در آن صورت تمامی **attribute** هایی که به کلاس نام برده اعمال شده، از دست خواهد رفت. این روش بسیار ناکارآمد بوده و مستعد از دست دادن قوانین مهم اعتبار سنجی می باشد.

جهت رفع این مشکل، کافی است یک کلاس **metadata** که دارای خصیصه های مورد نظر می باشد، به کلاس **model** اضافه کنید. با متصل کردن کلاس **model** به کلاس **metadata**، تمامی آن **attribute** ها به **model** اعمال می شوند. در این روش، کلاس **model** می تواند بدون اینکه تمامی خصیصه هایی که به کلاس **metadata** اعمال شده اند را از دست بدهد، مجددا ایجاد شود. در پوشه ی **Models**، یک کلاس به نام **Metadata.cs** اضافه کنید.



کد زیر را جایگزین کد موجود در فایل **Metadata.cs** نمایید.

```
using System;
using System.ComponentModel.DataAnnotations;
namespace ContosoSite.Models
{
```

```

public class StudentMetadata
{
    [StringLength(50)]
    [Display(Name="Last Name")]
    public string LastName;
    [StringLength(50)]
    [Display(Name="First Name")]
    public string FirstName;
    [StringLength(50)]
    [Display(Name="Middle Name")]
    public string MiddleName;
    [Display(Name="Enrollment Date")]
    public Nullable<System.DateTime> EnrollmentDate;
}
public class EnrollmentMetadata
{
    [Range(0, 4)]
    public Nullable<decimal> Grade;
}
}

```

این کلاس **metadata** در بردارنده ی تمامی **validation attribute** هایی هستند که قبلا به کلاس های **model** اعمال کرده بودید. به وسیله ی خصیصه ی **Display** مقدار مورد استفاده ی **text label** ها را اصلاح می کنیم. حال می بایست کلاس های **model** را به کلاس های **metadata** ارتباط داده و متصل (**associate**) کنید. کلاس به نام **PartialClasses.cs** در پوشه ی **Models** ایجاد کنید. محتویات فایل را با کد زیر جایگزین نمایید:

```

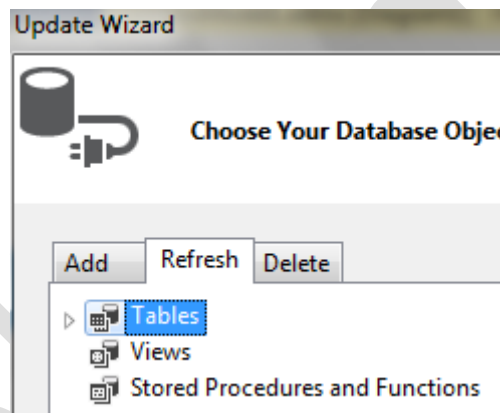
using System;
using System.ComponentModel.DataAnnotations;
namespace ContosoSite.Models
{
    [MetadataType(typeof(StudentMetadata))]
    public partial class Student
    {
    }
    [MetadataType(typeof(EnrollmentMetadata))]
    public partial class Enrollment
    {
    }
}

```

همان طور که مشاهده می کنید، هر کلاس به صورت **partial** علامت گذاری و تعریف شده و هر کدام نیز با نام و فضای نام (**namespace**) کلاسی که به صورت خودکار ایجاد شده، منطبق (**match**) می باشد. با اعمال خصیصه ی **metadata** به کلاس **partial**، مطمئن می شوید **data validation attribute** (خصیصه هایی که

داده را اعتبارسنجی می کنند) به کلاسی که به صورت اتوماتیک ایجاد شده نیز اعمال می شوند. این خصیصه ها، هنگامی که کلاس های **model** مجدداً ایجاد می شوند، از دست نمی رود، زیرا خصیصه ی **metadata** در کلاس های **partial** که دیگر ایجاد نمی شوند، اعمال شده اند.

به منظور ایجاد مجدد کلاس هایی که به صورت خودکار ساخته شده اند، می بایست فایل **ContosoModel.edmx** را باز کنید. بار دیگر، در محیط **design** راست کلیک کرده و گزینه ی **Update Model from Database** را انتخاب کنید. اگرچه پایگاه داده را تغییر نداده اید، با این وجود فرایند مورد نظر کلاس ها را مجدداً ایجاد می کند. در تب **Refresh**، **Tables** و **Finish** را انتخاب نمایید.



با ذخیره ی فایل **ContosoModel.edmx**، تغییرات را ذخیره کنید.

یکی از دو فایل های **Student.cs** و **Enrollment.cs** را باز کنید. خواهی دید **validation attribute** هایی که قبلاً اعمال کرده بودید، دیگر در فایل مورد نظر وجود ندارند. برنامه را اجرا کنید، می بینید صرف نظر از مسئله ی ذکر شده، هنگامی که داده هایی را وارد می کنید، آن قوانین اعتبارسنجی پیاده شده و در نتیجه هنوز پابرجا هستند.