

چندی پیش در همین وب‌سایت مطلبی تحت عنوان «[اعمال کلاس‌های ویژه اعتبارسنجی Twitter bootstrap به فرم‌های ASP.NET MVC](#)» منتشر شد. این مقاله مرتبط با نسخه دوم فریم‌ورک محبوب Bootstrap بود. قصد داریم به بازنویسی کدهای مرتبط بپردازیم و کلاس‌های مرتبط با [نسخه سوم این فریم‌ورک](#) را هم با فرم‌های خودمان سازگار کنیم. مثل مقاله‌ی ذکر شده توضیحات را با یک مثال همراه می‌کنم.

مدل برنامه

```
using System.ComponentModel.DataAnnotations;
using System.ComponentModel.DataAnnotations.Schema;

namespace FormValidationWithBootstrap.Models
{
    [Table("Product")]
    public class ProductModel
    {
        [Key]
        public int Id { get; set; }
        [Required(ErrorMessage = "{0} باید آن را وارد کنید {0}")]
        [StringLength(50, ErrorMessage = "{1} کاراکتر باشد")]
        [Display(Name = "نام کالا")]
        public string Name { get; set; }
        [Required(ErrorMessage = "{0} باید آن را وارد کنید {0}")]
        [Display(Name = "قیمت")]
        [DataType(DataType.Currency)]
        public double Price { get; set; }
        [Required(ErrorMessage = "{0} باید آن را وارد کنید {0}")]
        [Display(Name = "موجودی")]
        public int Qty { get; set; }
    }
}
```

قرار هست که جدولی داشته باشیم با نام Product برای ثبت محصولات. مدل برنامه شامل خاصیت‌های مرتبط و همچنین اعتبارسنجی‌های مد نظر ما هست.

کنترلر برنامه

```
using System.Web.Mvc;
using FormValidationWithBootstrap.Models;

namespace FormValidationWithBootstrap.Controllers
{
    public class ProductController : Controller
    {
        // GET: Product
        public ActionResult Index()
        {
            return View();
        }

        public ActionResult New()
        {
            return View();
        }

        [HttpPost]
        public ActionResult New(ProductModel product)
        {
            if (!ModelState.IsValid)
                return View(product);

            if (product.Name != "پفک")
            {
                ModelState.AddModelError("", "!لطفا مشکلات را برطرف کنید");
                ModelState.AddModelError("Name", "نام پفک قابل ثبت است");
                return View(product);
            }
            // todo:save...
        }
    }
}
```

```

        return RedirectToAction("Index");
    }
}

```

در قسمت کنترلر نیز اتفاق خاصی نیفتاده و کارهای پایه فقط انجام شده؛ ضمن اینکه آمدم برای داشتن خطاهای سفارشی نام محصول را چک کردیم و گفتیم اگر نام محصول چیزی غیر از «پفک» بود، از سمت سرور خطایی را صادر کند و بگوید که فقط پفک قابل ثبت هست.

View برنامه

```

@model FormValidationWithBootstrap.Models.ProductModel
@{
    ViewBag.Title = "New";
}
<h2>کالای جدید</h2>
@using (Html.BeginForm())
{
    @Html.AntiForgeryToken()
    <div>
        <hr />
        @Html.ValidationSummary(true, "", new { @class = "alert alert-danger" })
        <div>
            @Html.LabelFor(model => model.Name, htmlAttributes: new { @class = "control-label col-md-2"
        })
            <div>
                @Html.EditorFor(model => model.Name, new { htmlAttributes = new { @class = "form-
control" } })
                @Html.ValidationMessageFor(model => model.Name, "", new { @class = "text-danger" })
            </div>
        </div>
        <div>
            @Html.LabelFor(model => model.Price, htmlAttributes: new { @class = "control-label col-md-
2" })
            <div>
                @Html.EditorFor(model => model.Price, new { htmlAttributes = new { @class = "form-
control" } })
                @Html.ValidationMessageFor(model => model.Price, "", new { @class = "text-danger" })
            </div>
        </div>
        <div>
            @Html.LabelFor(model => model.Qty, htmlAttributes: new { @class = "control-label col-md-2"
        })
            <div>
                @Html.EditorFor(model => model.Qty, new { htmlAttributes = new { @class = "form-
control" } })
                @Html.ValidationMessageFor(model => model.Qty, "", new { @class = "text-danger" })
            </div>
        </div>
        <div>
            <div>
                <input type="submit" value="ثبت" />
                <input type="reset" value="ریست" />
                @Html.ActionLink("بازگشت به لیست", "Index", "Product", null, new {@class="btn btn-
default"})
            </div>
        </div>
    </div>
}

```

فایل View برنامه با Scaffold Template ساخته شده و چون از Visual Studio 2013 استفاده شده، به صورت پیش فرض با بوت استرپ سازگار هست. تغییری که ایجاد شده تعویض کلاس مربوط به ValidationSummary هست که به alert alert-danger تغییر پیدا کرده و همچنین دو دکمه «ریست» و «بازگشت به لیست» هم به کنار دکمه «ثبت» اضافه شده.

کالای جدید

| | |
|----------------------------------------------------|------------------------------------------------------------------------|
| <input type="text"/> | نام کالا |
| نام کالا یک فیلد اجباری است و باید آن را وارد کنید | |
| <input type="text"/> | قیمت |
| قیمت یک فیلد اجباری است و باید آن را وارد کنید | |
| <input type="text"/> | موجودی |
| موجودی یک فیلد اجباری است و باید آن را وارد کنید | |
| <input type="button" value="بازگشت به لیست"/> | <input type="button" value="ریست"/> <input type="button" value="ثبت"/> |

در فرم بالا شاهد هستیم که با کلیک بر روی دکمه ثبت تنها خطاهای مرتبط با هر ردیف ظاهر شده‌اند و هیچ تغییر رنگی که حاصل از کلاس‌های مرتبط با Bootstrap باشند حاصل نشده. برای رفع این مشکل کافی‌است اسکریپت زیر، به انتهای فایل View برنامه اضافه شود تا پیش‌فرض‌های jQuery Validator را تغییر دهیم و آن‌ها را با بوت‌استرپ سازگار کنیم. همچنین در حالت ارسال فرم به سرور و Postback و نمایش خطاهای سفارشی، قسمت بررسی field-validation-error صورت می‌گیرد و در صورتیکه موردی را پیدا کند، به سطر مرتبط با آن کلاس has-error اضافه خواهد شد.

```
@section Scripts {
    @Scripts.Render("~/bundles/jqueryval")
    <script>
        // override jquery validate plugin defaults
        $.validator.setDefaults({
            highlight: function (element) {
                $(element).closest('.form-group').addClass('has-error');
            },
            unhighlight: function (element) {
                $(element).closest('.form-group').removeClass('has-error').addClass('has-success');
            },
            errorElement: 'span',
            errorClass: 'help-block',
            errorPlacement: function (error, element) {
                if (element.parent('.input-group').length) {
                    error.insertAfter(element.parent());
                } else {
                    error.insertAfter(element);
                }
            }
        });
        $(function () {
            $('form').each(function () {
                $(this).find('div.form-group').each(function () {
                    if ($(this).find('span.field-validation-error').length > 0) {
                        $(this).addClass('has-error');
                    }
                });
            });
        });
    </script>
}
```

با افزودن اسکریپت فوق، در حالت اعتبارسنجی فرم‌ها به شکل زیر می‌رسیم:

کالای جدید

| | |
|----------------------------------------------------|------------------------------------------------------------------------|
| <input type="text"/> | نام کالا |
| نام کالا یک فیلد اجباری است و باید آن را وارد کنید | |
| <input type="text"/> | قیمت |
| قیمت یک فیلد اجباری است و باید آن را وارد کنید | |
| <input type="text"/> | موجودی |
| موجودی یک فیلد اجباری است و باید آن را وارد کنید | |
| <input type="button" value="بازگشت به لیست"/> | <input type="button" value="ریست"/> <input type="button" value="ثبت"/> |

همچنین هنگامیکه کاربر فیلد را به درستی وارد کرد، رنگ فیلد و همچنین آن ردیف به سبز تغییر خواهد کرد.

کالای جدید

| | |
|--------------------------------------------------|------------------------------------------------------------------------|
| <input type="text" value="تست"/> | نام کالا |
| قیمت | |
| قیمت یک فیلد اجباری است و باید آن را وارد کنید | |
| <input type="text"/> | موجودی |
| موجودی یک فیلد اجباری است و باید آن را وارد کنید | |
| <input type="button" value="بازگشت به لیست"/> | <input type="button" value="ریست"/> <input type="button" value="ثبت"/> |

و همچنین در حالت رخداد یک خطای سفارشی پس از postback از سمت سرور به حالت زیر خواهیم رسید.

کالای جدید

• لطفا مشکلات را برطرف کنید!

| | |
|-----------------------------------------------|------------------------------------------------------------------------|
| <input type="text" value="آدامس"/> | نام کالا |
| فقط محصولی با نام پفک قابل ثبت است :-) | |
| <input type="text" value="۱۲۰۰۰"/> | قیمت |
| <input type="text" value="۱۰"/> | موجودی |
| <input type="button" value="بازگشت به لیست"/> | <input type="button" value="ریست"/> <input type="button" value="ثبت"/> |

کدهای کامل این مثال را از اینجا می‌توانید دریافت کنید

[FormValidationWithBootstrap.rar](#)

نظرات خوانندگان

نویسنده: مهدی پایروند
تاریخ: ۱۳۹۳/۰۵/۱۰ ۲۲:۳۴

سلام لطفا مثال پست را هم برای دانلود بذارید. ممنون

نویسنده: علی اکبر کورش فر
تاریخ: ۱۳۹۳/۰۵/۱۱ ۱۳:۳۳

سلام. متأسفانه حجم پروژه به دلیل فایل‌های دی‌ال‌ال موجود 18 مگ بود و من نتونستم کمترش کنم

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۵/۱۱ ۱۳:۳۸

برای ارسال پروژه‌ها در وب، همیشه پوشه‌های **packages**، پوشه‌های **bin** و **obj** را حذف کنید. نیازی به آن‌ها نیست، چون [قابل بازسازی](#) هستند.

نویسنده: علی اکبر کورش فر
تاریخ: ۱۳۹۳/۰۵/۱۱ ۲۳:۰۰

خیلی ممنونم. خیلی دنبال این مسئله بودم. مچکرم از راهنماییتون

نویسنده: علی اکبر کورش فر
تاریخ: ۱۳۹۳/۰۵/۱۱ ۲۳:۰۱

با عرض سلام. پروژه پیوست شد

نویسنده: هادی ام
تاریخ: ۱۳۹۳/۰۵/۱۳ ۱۹:۲۱

این بوتسترپ شما، فارسی شده هست؟

نویسنده: علی اکبر کورش فر
تاریخ: ۱۳۹۳/۰۵/۱۴ ۸:۲۸

فارسی‌شده در حد نیاز... قسمت‌هایی که ازشون استفاده می‌کنم رو فارسی کردم.
فکر می‌کنم نسخه فارسی سازی شده بوت‌استرپ 3 هم هستش

نویسنده: محمد دلیری
تاریخ: ۱۳۹۳/۰۵/۲۰ ۰:۵۱

اون قسمت‌های که میگی کجاها واسه فارسی کردن تغییر دادین که راست به چپ شده ؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۵/۲۰ ۱:۳۲

- [نسخه 3 راست به چپ](#)
- [نسخه 3.1.1 راست به چپ](#)
- [نسخه 3.2 راست به چپ](#)