عنوان: **مقدمه ای بر AutoMapper**
نویسنده: محمد صاحب
تاریخ: ۱۳۹۱/۰۴/۱۳ ۱۷:۲۲
آدرس: www.dotnettips.info
برچسب‌ها: ASP.Net MVC, AutoMapper, ASP.Net, MVC

AutoMapper کتابخانه ای ساده و سبک برای نگاشت اطلاعات یک شی به شی دیگر به صورت خودکار هست و...
اگه این پست رو مطالعه کرده باشید یه مشکل امنیتی بنام «Mass Assignment» مطرح شد.برای رفع این مشکل یک روش استفاده از ViewModel بود.
فرض کنید Model ما

```
public class User
    {
        public int Id { get; set; }

        public string FirstName { get; set; }

        public string LastName { get; set; }

        public string UserName { get; set; }

        public string Password { get; set; }

        public bool IsAdmin { get; set; }

        public virtual ICollection<BlogPost> BlogPosts { get; set; }
    }
```

و ViewModel ما

```
public class UserViewModel
    {
        public string FirstName { get; set; }

        public string LastName { get; set; }

        public string Password { get; set; }
    }
```

باشه(توجه کنید در واقع من برای View ی مورد نظرم فقط به نام , نام خانوادگی و پسورد نیاز دارم)
برای استفاده UserViewModel بعنوان Model در View ی مورد نظر باید شی UserViewModel رو با اطلاعات شی User مقدار دهی کنیم مثلا با کدی مثل این در کنترلر.

```
public ActionResult Index(int id = 1)
        {
            var user = _userService.GetById(id);
            var userViewModel = new UserViewModel
                {
                    FirstName = user.FirstName,
                    LastName = user.LastName,
                    Password = user.Password
                };

            return View(userViewModel);
        }
```

رهایی از نوشتن اینجور کدهای تکراری و خسته کننده باعث پیدایش AutoMapper شد...
برای استفاده از AutoMapper از نوگت استفاده میکنیم.

```
PM> Install-Package AutoMapper
```

در شروع برنامه نگاشت‌ها رو تعریف میکنم.یک روش ابداعی تعریف نگاشت‌ها در یک کلاس استاتیک و فراخوانی اون تو متد Application_Start هست.

```
public static class AutoMapperWebConfiguration
    {

        public static void Configure()
        {
            ConfigureUserMapping();
        }

        private static void ConfigureUserMapping()
        {
            Mapper.CreateMap<User, UserViewModel>();
        }
    }
```

اولین پارامتر نوع مبدا و دومین پارامتر نوع مقصد هست.
برای انجام نگاشت هم از متد Map استفاده میکنیم.

```
public ActionResult Index(int id=1)
        {
            var user = _userService.GetById(id);
            var userViewModel=new UserViewModel();
            AutoMapper.Mapper.Map(user, userViewModel);

            return View(userViewModel);
        }
```

همنطور که میبنید با نوشتن چندین خط کد عملیات نگاشت اطلاعات یک شی به شی دیگه انجام شد.
ادامه دارد...

نظرات خوانندگان

نویسنده: رضا
تاریخ: ۱۳۹۱/۰۴/۱۳ ۱۷:۳۲

دمت گرم مدتیه معطل همین مپینگ‌ها به صورت دستی هستم

نویسنده: سروش ترک زاده
تاریخ: ۱۳۹۱/۰۴/۱۳ ۱۷:۳۳

سلام
ممنون،
راه حل ساده واقعا لذت بخش هست...

نویسنده: daniel
تاریخ: ۱۳۹۱/۰۴/۱۳ ۲۱:۱۳

تو Compact Framwork از valueinjecter باید استفاده کرد.

نویسنده: امیرحسین جلوداری
تاریخ: ۱۳۹۱/۰۴/۱۳ ۲۳:۲

خیلی ممنونم ... میشه این سمپل رو ضمیمه کنید؟!

نویسنده: رضا.ب
تاریخ: ۱۳۹۱/۰۴/۱۳ ۲۳:۴۶

تشکر بابت معرفی.
کاربرد این Auto Mapper در اینجور نگاشت‌ها معنی میده فقط؟ یعنی نگاشت یه شیِ جزء به شیِ کل؟ یا مثلا کاربردهایی مثل ORM هم میتواند داشته باشد؟

نویسنده: محمد صاحب
تاریخ: ۱۳۹۱/۰۴/۱۴ ۹:۲۱

یکی از مهمترین کاربردهاش نگاشت از یک مدل بزرگ و پیچیده به یک مدل ساده هست که بهش Flattening میگن و کاربردهای دیگه مثلا Projection و... که همنطور که آخر پست گفتم تو پست‌های بعدی تکمیلش میکنم.
برای مطالعه بیشتر همه به این ویکی مراجعه کنید.

نویسنده: محمد صاحب
تاریخ: ۱۳۹۱/۰۴/۱۴ ۱۲:۱۶

سمپل خاصی نیست کدها رو از یک پروژه کشیدم بیرون...
تنها قسمتی که تو این پست نذاشتم(و البته نیاز هم نبود) View ش بود.

```
@model DotnetDevBlog.Web.Models.UserViewModel

@{
    ViewBag.Title = "Index";
}

<h2>Index</h2>
@Html.DisplayForModel()
```

نویسنده: سعید یزدانی
تاریخ: ۱۳۹۲/۰۴/۰۲ ۱۶:۱۹

با سلام
ممنون از مطلب مفیدتون
میخواستم بدونم اگه از چند model بخواهیم استفاده کنیم باید چکار کنیم؟

نویسنده: محمد صاحب
تاریخ: ۱۳۹۲/۰۴/۰۴ ۱۰:۵۸

از این کلاس کمکی استفاده کن

نویسنده: hossein101211
تاریخ: ۱۳۹۲/۰۴/۱۰ ۱۳:۵۸

با سلام میخواستم ببینم آیا امکان استفاده از automapper بدین صورت هم وجود داره؟

```
public void Edit_news_ajax(News news)
        {
            using (var Context = new ProCamContext())
            {
                var q=Context.News.Find(news.id);
                AutoMapper.Mapper.Map(news,q);
               //NewsRepository.EditNews(news.id, q);
                Context.SaveChanges();
            }//end news
        }
```

با تشکر

نویسنده: محمد صاحب
تاریخ: ۱۳۹۲/۰۴/۱۰ ۱۴:۳۰

اگه q از نوع news هست که نیاز ی به AutoMapper نداری !

```
AutoMapper.Mapper.Map(news,q);
```

این خط مشکلی نداره

نویسنده: hossein101211
تاریخ: ۱۳۹۲/۰۴/۱۰ ۱۵:۵

من میخوام محتویات news رو (که توسط model binding پر شده) رو در سطری از جدولم بریزم (با اون آیدی که با find پیدا کردم جایگزین بشه) اینکار با AutoMapper امکان پذیر هست؟
برای اینکار نیاز به create.map هم دارم؟
با تشکر

نویسنده: محمد صاحب
تاریخ: ۱۳۹۲/۰۴/۱۱ ۱۱:۲۷

اگه نوع هاشون متفاوته بله باید Mapper.CreateMap رو تعریف کنی.

عنوان: **بهبود سرعت نمایش صفحات در ASP.NET MVC با حذف View Engines اضافی**
نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۴/۱۳ ۲۳:۴۳
آدرس: www.dotnettips.info
برچسب‌ها: ASP.Net, MVC, ASP.Net MVC

در ASP.NET MVC امکان استفاده از چند View Engine به صورت همزمان وجود دارد و همچنین هربار که قرار است Viewایی رندر شود، از تمام این‌ها تا یافتن موتور مناسب نمایش View جاری کوئری می‌گیرد. بدیهی است هرچقدر تعداد موتورهای ثبت شده در اینجا بیشتر باشند، زمان بیشتری نیز برای یافتن موتور نمایشی مناسب صرف خواهد شد؛ خصوصا اگر موتور مناسب در آخر لیست ثبت شده باشد.

در ASP.NET MVC 3 دو موتور نمایشی به صورت پیش فرض نصب هستند (WebForms and Razor). بنابراین اگر صرفا از Razor استفاده می‌کنید، می‌توان موتور اول را کلا از سیستم پردازشی برنامه حذف کرد. برای اینکار تنها کافی است در فایل global.asax.cs برنامه بنویسیم:

```
protected void Application_Start() {
    ViewEngines.Engines.Clear();
    ViewEngines.Engines.Add(new RazorViewEngine());
    ...
}
```

این موارد را توسط Glimpse بهتر می‌توان بررسی کرد. Glimpse یک پروفایلر سمت سرور ASP.NET است و دارای نسخه مخصوص ASP.NET MVC نیز می‌باشد. برای نصب آن باید از طریق NuGet اقدام کرد و حتما دقت داشته باشید که نسخه MVC آن باید نصب شود تا برگه‌های Routing و View آن ظاهر شوند.

پس از نصب از طریق NuGet، به صورت خودکار اسمبلی‌های لازم به پروژه اضافه شده و همچنین فایل web.config برنامه نیز ویرایش می‌شود. در انتهای این فایل سطر ذیل مشخص می‌کند که Glimpse فعال باشد یا خیر.

```
<glimpse enabled="true" />
```

پس از نصب، برنامه را اجرا کرده و به آدرس http://localhost/ **glimpse.axd** مراجعه کنید تا صفحه تنظیمات آن ظاهر شود. تنها کاری که باید در اینجا صورت گیرد کلیک بر روی دکمه Turn Glimpse On است.

به این ترتیب یک کوکی به مرورگر اضافه شده و اکنون پس از بازگشت به صفحه اصلی برنامه و refresh کامل صفحه، در کنار سمت راست پایین صفحه، آیکن آن ظاهر خواهد شد.

بر روی این آیکن کلیک نمائید تا در برگه‌ی View آن، انواع Viewهایی که درگیر نمایش صفحه جاری بوده‌اند، مشخص شوند:

همانطور که ملاحظه می‌کنید در اینجا دو موتور پیش فرض فعال بوده و پس از سعی و خطای صورت گرفته، در انتهای کار Razor انتخاب شده است. اکنون اگر نکته حذف موتورهای نمایشی اضافی را اعمال کنیم به تصویر زیر خواهیم رسید:

/

Ajax | Binding | Config | Environment | Execution | MetaData | Remote | Request | Routes | Server

Ordinal	Requested View	Master Override	Partial	View Engine	Check Cache	Found	Details
1	Index		False	RazorViewEngine	True	False	Not Found In: RazorViewEngine cache
2	Index		False	RazorViewEngine	False	True	Key / Value: ViewData --, Model --, TempData --

هم تعداد سعی و خطاها کمتر شده و هم تعداد فایل‌هایی که بررسی شده است به حداقل رسیده (برای مثال در حالتیکه موتور WebForms فعال باشد، چهار فایل با پسوندهای مختلف در مکان‌های پیش فرض نیز حتما جستجو خواهند شد).

نظرات خوانندگان

نویسنده: شهروز جعفری
تاریخ: ۱۳۹۱/۰۴/۱۳ ۲۳:۵۸

یه سوال شاید نه چندان حرفه ای
شما گفتید که ا مکان استفاده از چند View Engine به صورت همزمان وجود دارد .آیا میشه تو یک ویو این کارو انجام داد؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۴/۱۴ ۰:۱

خیر. همزمانی در اینجا به معنای امکان استفاده همزمان از فایل‌های مستقل aspx و cshtml یا vbhtml است.

نویسنده: شهروز جعفری
تاریخ: ۱۳۹۱/۰۴/۱۴ ۰:۱۹

یه سوال نامرتبط Ninject چیه؟کاره همون structuremap میکنه؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۴/۱۴ ۰:۲۷

بله. از این نوع کتابخانه‌ها زیاد است. Ninject، StructureMap، AutoFac و غیره. خود مایکروسافت هم یک نمونه دیگر به نام Unity دارد.

عنوان: **محدود کردن کاربرها به آپلود فایل‌هایی خاص در ASP.NET MVC**
نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۴/۱۶ ۰:۱
آدرس: www.dotnettips.info
گروه‌ها: ASP.Net, MVC, Security, ASP.Net MVC

یک ادیتور آنلاین را تصور کنید که کاربران در قسمت ارسال تصویر آن قرار است فقط فایل‌های png، jpg و gif ارسال کنند و نه مثلا فایل test.aspx و موارد مشابه. در اینجا برای محدود کردن نوع فایل‌های آپلود شده می‌توان از فیلترهای سفارشی ASP.NET MVC کمک گرفت:

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Web.Mvc;

namespace SecurityModule
{
    public class AllowUploadSpecialFilesOnlyAttribute : ActionFilterAttribute
    {
        readonly List<string> _toFilter = new List<string>();
        readonly string _extensionsWhiteList;
        public AllowUploadSpecialFilesOnlyAttribute(string extensionsWhiteList)
        {
            if (string.IsNullOrWhiteSpace(extensionsWhiteList))
                throw new ArgumentNullException("extensionsWhiteList");

            _extensionsWhiteList = extensionsWhiteList;
            var extensions = extensionsWhiteList.Split(',');
            foreach (var ext in extensions.Where(ext => !string.IsNullOrWhiteSpace(ext)))
            {
                _toFilter.Add(ext.ToLowerInvariant().Trim());
            }
        }

        bool canUpload(string fileName)
        {
            if (string.IsNullOrWhiteSpace(fileName)) return false;

            var ext = Path.GetExtension(fileName.ToLowerInvariant());
            return _toFilter.Contains(ext);
        }

        public override void OnActionExecuting(ActionExecutingContext filterContext)
        {
            var files = filterContext.HttpContext.Request.Files;
            foreach (string file in files)
            {
                var postedFile = files[file];
                if (postedFile == null || postedFile.ContentLength == 0) continue;

                if (!canUpload(postedFile.FileName))
                    throw new InvalidOperationException(
                        string.Format("You are not allowed to upload {0} file. Please upload only these
files: {1}.",
                                      Path.GetFileName(postedFile.FileName),
                                      _extensionsWhiteList));
            }

            base.OnActionExecuting(filterContext);
        }
    }
}
```

توضیحات کدهای فوق:

برای تهیه فیلتر محدود سازی نوع فایل‌های قابل ارسال به سرور، با ارث بری از ActionFilterAttribute شروع خواهیم کرد.
سپس با تحریف متد OnActionExecuting آن، توسط filterContext.HttpContext.Request.Files می‌توان به کلیه فایل‌های درحال ارسال به سرور در طی درخواست جاری، دسترسی یافت.
به این ترتیب از طریق مقدار خاصیت postedFile.FileName می‌توان به پسوند فایل در حال ارسال رسید و بر این اساس امکان

ارسال فایل‌های غیرمجاز را در نیمه راه با صدور یک استثناء سلب کرد.

برای استفاده از این فیلتر سفارشی تهیه شده نیز می‌توان به نحو زیر عمل کرد:

```
[AllowUploadSpecialFilesOnly(".jpg,.gif,.png")]
public ActionResult ImageUpload(HttpPostedFileBase file)
```

در اینجا پسوند فایل‌های مجاز قابل ارسال، توسط یک کاما از هم جدا خواهند شد.

یک نکته تکمیلی:

اگر کاربر قرار است تنها تصویر ارسال کند، بررسی پسوند فایل لازم است اما کافی نیست. برای این منظور می‌توان از کلاس Image واقع شده در فضای نام System.Drawing نیز کمک گرفت:

```
public static bool IsImageFile(HttpPostedFileBase photoFile)
{
    using (var img = Image.FromStream(photoFile.InputStream))
    {
         return img.Width > 0;
    }
}
```

در اینجا اگر فایل ارسالی تصویر نباشد، به صورت خودکار یک استثناء صادر خواهد شد.

نظرات خوانندگان

نویسنده: حسین مرادی نیا
تاریخ: ۱۳۹۱/۰۴/۱۷ ۱۴:۵

سلام
همانطور که گفتید چک کردن پسوند فایل الزامی است ولی کافی نیست.برای همین از کلاس Image استفاده کردید که اگر فایل ارسال شده یک فایل تصویری نبود استثنایی صادر شود. حالا برای فایل‌ها با فرمت‌های دیگه چیکار کنیم؟ (مثلا ممکن است بخواهیم به کاربر اجازه ارسال فایل‌های zip و یا rar بدهیم و مواردی از این دست).

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۴/۱۷ ۱۴:۱۶

برای بررسی محتوایی خاص، نیاز به parser مخصوص این نوع فایل‌ها است تا بتواند بررسی کند محتوای دریافتی معتبر است یا نه و عموما کسی محتوای این نوع فایل‌ها را بررسی نمی‌کند. بنابراین بررسی پسوند در اکثر موارد کافی است.
مشکل این نیست که فایل rar واقعا rar است یا نه. مشکل این است که این فایل ارسالی، قابلیت اجرا را از طریق فراخوانی آدرس آن در مرورگر، نداشته باشد. بحث بررسی پسوند هم به همین دلیل است. فایل aspx را می‌شود از طریق فراخوانی در مرورگر بر روی سرور اجرا کرد ولی فایلی با پسوند rar این قابلیت را ندارد.

نویسنده: شاهین کیاست
تاریخ: ۱۳۹۱/۰۵/۰۹ ۲:۱

ممنون ، استثنایی که صادر می‌شود را چطور میشه ؟ در بدنه‌ی Action مربوطه مدیریت کرد ؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۵/۰۹ ۸:۲۳

لازم نیست مدیریت کنید. هدف این بوده که نظم جاری را تغییر دهد. این استثناء توسط ELmah دریافت و ثبت خواهد شد. همچنین کاربر به یکی از صفحات پیش فرض خطای برنامه هدایت می‌شود و متوجه خواهد شد که خطایی رخ داده است.
ولی در کل می‌شود IExceptionFilter را نیز پیاده سازی و مدیریت کرد:

```
public class CustomFilter : FilterAttribute, IExceptionFilter
{
    public void OnException(ExceptionContext filterContext)
    {
```

نویسنده: رضوی
تاریخ: ۱۳۹۱/۰۵/۲۸ ۸:۵۳

سلام

چند نکته تکمیلی برای امنیت بیشتر
1- بهتر است که فایل روی دیتابیس ذخیره شود، در غیر اینصورت نام فایل و پسوند توسط سیستم تعیین شود.
2- نام و پسوند ارسالی در نظر گرفته نشود.
3- با اندازه گیری سایز فایل GIF نمیتوان به معتبر بودن آن اطمینان داشت.

http://www.pyxsoft.com/en/hacking-hackers.html
/http://www.exploit-db.com/exploits/16181

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۱/۰۷ ۱۱:۱۶

یک نکته‌ی تکمیلی

نحوه‌ی فقط خواندنی کردنی دسترسی به فایل‌های یک پوشه (گرفتن دسترسی اجرا)

```
<location path="upload">
  <system.webServer>
    <handlers accessPolicy="Read" />
  </system.webServer>
</location>
```

عنوان: **ویژگی های پیشرفته ی AutoMapper - قسمت اول**
نویسنده: محمد صاحب
تاریخ: ۱۳۹۱/۰۴/۲۲ ۱۳:۴۰
آدرس: www.dotnettips.info
برچسب‌ها: ASP.Net MVC, AutoMapper

در پست قبلی مقدمه‌ای داشتیم بر AutoMapper؛ مثالی که در اون پست عنوان شد ساده‌ترین و پرکاربردترین روش استفاده از AutoMapper هست بنام Flattening که در واقع از یک شیء کل به یک شیء کوچکتر می‌رسیم.
همانطور که در قسمت اول گفتم AutoMapper کارش رو بر اساس قراردادها انجام میده یا همون Convention Base. یکی از قردادهای AutoMapper، نگاشت براساس نام اعضای اون شی هست؛ مثلا در مثال قبلی FirstName در مبداء، به خاصیتی با همین نام نگاشت شد و ...

Projection

روش استفاده بعدی که به اون Projection (معنی فارسی خوب برا Projection چیه ؟) میگن برای مواقعی هست که اعضای یک شیء در مبداء، همتایی در مقصد ندارد (از نظر نام) و در واقع می‌خواهیم یک شیء رو به یک شیء دیگه تغییر شکل بدیم.
مدل زیر رو در نظر بگیرید:

```
public class Person
    {
        public int Id { get; set; }

        public string FirstName { get; set; }

        public string LastName { get; set; }

    }
```

که قرار است به مدل PersonDTO نگاشت بشه.

```
public class PersonDTO
    {
        public int Id { get; set; }

        public string Name { get; set; }

        public string Family { get; set; }

public string FullName { get; set; }

        public string Email { get; set; }
    }
```

همنطور که می‌بینید در مقصد خاصیت‌هایی داریم که در مبداء همتایی ندارند. برای نگاشت چنین اشیایی، از متد ForMember استفاده و نگاشت‌های شی‌های موردنظر رو Custom می‌کنیم.

```
Mapper.CreateMap<Person, PersonDTO>().ForMember(des => des.Name, op => op.MapFrom(src =>
src.FirstName)).
                    ForMember(des => des.FullName, op => op.MapFrom(src => src.FirstName + " " +
src.LastName)).ForMember(
                        des => des.Email, op => op.Ignore());
```

متد ForMember از یک Action Delegate برای کانفیگ کردن هر عضو استفاده میکنه. در مثال بالا ما از MapFrom برای اعمال نگاشت Custom استفاده کردیم.

```
AutoMapper.IMappingExpression<TSource,TDestination>.ForMember(System.Linq.Expressions.Expression<System
.Func<TDestination,object>>, System.Action<AutoMapper.IMemberConfigurationExpression<TSource>>)
```

نکته :برای تست کردن اینکه آیا نگاشت ما Exception داره یا نه میتوان از متد زیر استفاده کرد.

```
Mapper.AssertConfigurationIsValid();
```

نکته : همیشه موقع نگاشت، اعضای شیء مقصد برای AutoMapper مهمن؛ مثلا در مثال بالا خاصیت LastName در مبداء به هیچ عضوی در مقصد نگاشت نشده (به صورت مستقیم) و این تولید Exception نمیکنه ولی برعکس اون باعث تولید Exception میشه؛ مثلا اگه خاصیت Email رو که در مبداء همتایی نداره رو کانفیگ نکنیم، باعث تولید Exception میشه.

نحوه استفاده

```
var person = new Person
                {
                    Id = 1,
                    FirstName = "Mohammad",
                    LastName = "Saheb",

                };
var personDTO = Mapper.Map<Person, PersonDTO>(person);
```

خروجی به شکل زیر میشه

Collection

بعد از نوشتن کانفیگ نگاشت‌ها، بدون نیاز به تنظیمات خاصی میتونیم مجموعه ای از شی‌های مبداء رو به مقصد نگاشت کنیم.
مجموعه‌های پشتیبانی شده به شرح زیرن.

```
IEnumerable, IEnumerable<T>, ICollection, ICollection<T>, IList, IList<T>, List<T>.
```

و البته آرایه ها.
مثال

```
var persons = new[]
                {
                    new Person { Id = 1, FirstName = "Mohammad", LastName = "Saheb" },
                    new Person { Id = 2, FirstName = "Babak", LastName = "Saheb" }
                };

var personDTOs = Mapper.Map<Person[], List<PersonDTO>>(persons);
```

خروجی به شکل زیر میشه

Nested Mappings

برای نگاشت کلاس‌های تو در تو از این روش استفاده می‌کنیم و ...

فرض کنید در کلاس Person خاصیتی از نوع Address داریم و در کلاس PersonDTO خاصیتی از نوع AddressDTO.

```
public class Address
    {
        public string Ad1 { get; set; }
        public string Ad2 { get; set; }
    }


public class AddressDTO
    {
        public string Ad1 { get; set; }
        public string Ad2 { get; set; }
    }
```

برای نگاشت‌هایی از این دست باید تنظیمات نگاشت مربوط به نوع‌های تودرتو را صریحا معین کنیم.

```
Mapper.CreateMap<Address, AddressDTO>();
```

نکته: هرچند منطقی‌تر بنظر میرسه که تعریف نگاشت‌های داخلی ابتدا بیاد، ولی فرقی نمیکنه که تعریف این نگاشت قبل یا بعد از نگاشت اصلی باشه.

ادامه دارد...

نظرات خوانندگان

نویسنده: ابراهیم
تاریخ: ۱۳۹۱/۰۴/۲۲ ۲۱:۴۱

ممنون از مطالب خوبتان. Projection را به صورت پرتو در درس پایگاه دادهٔ استاد حق‌جو شنیده‌ام و به نظرم ترجمهٔ خوبی است.

نویسنده: فرهاد یزدان پناه
تاریخ: ۱۳۹۲/۰۳/۱۰ ۱:۳

وقت بخیر جناب صاحب
در حالاتی که از AutoMapper در یک عبارت LINQ استفاده بشه در صورت وجود مواردی همچون FullName در مقصد خطای "Object reference not set to an instance of an object" رو throw می‌کنه.
آیا شما هم با چنین مشکلی برخورد کردید؟

نویسنده: محمد صاحب
تاریخ: ۱۳۹۲/۰۳/۱۱ ۱۰:۴۶

نمیدونم شاید.برای برسی بیشتر کدهاش رو بذار

نویسنده: لیلا
تاریخ: ۱۳۹۲/۰۹/۰۴ ۲۳:۱۶

با تشکر
به چه روشی می‌توان یک vieamodel که ترکیبی از 3 model هست به مدلهای مربوطه به وسیله AutoMapper ، نگاشت کرد

عنوان: **مدیریت خطا ها در ASP.NET MVC**
نویسنده: حسین رضیئی
تاریخ: ۱۳۹۱/۰۴/۲۲ ۲۰:۴۰
آدرس: www.dotnettips.info
برچسب‌ها: ASP.Net MVC, Handle Errors

احتمال بوجود اومدن خطا در اجرای یک برنامه همیشه هست. خطاهایی که برنامه نویس ممکنه هیچ وقت فکر نکنه چنین خطایی در اجرای برنامه ای که نوشته بوجود بیاد و هیچ وقت هم از اون خطا اطلاع پیدا نکنه. ثبت و بررسی خطاهایی که در برنامه بوجود میاد میتونه مفید باشه اما آیا باید همه‌ی خطاها رو ثبت کرد؟
خیر. ممکنه برخی از این خطاها سلامت سیستم رو به خطر نندازه و اصلا ارتباطی هم به برنامه نداشته باشه . به طور مثال زمانی که کاربر یک صفحه ای و یا فایلی رو درخواست میده که وجود نداره.
توسط رویداد OnException میتونیم به خطاهای بوجود اومده در سطح یک Controller دسترسی داشته باشیم و اون رو مدیریت کنیم.

```
protected override void OnException(ExceptionContext filterContext)
{
    // Bail if we can't do anything
    if (filterContext == null)
        return;

    // log
    var ex = filterContext.Exception ??
            new Exception("No further information exists.");
    // save log ex.Message

    filterContext.ExceptionHandled = true;

    filterContext.Result = View("ViewName");
    base.OnException(filterContext);
}
```

ما تمام خطا هایی که در سطح یک کنترل اتفاق می‌افته رو با دوباره نویسی(override) متد OnException مدیریت میکنیم. اما اگه بخواهیم محدوده‌ی این مدیریت رو بیشتر کنیم و کاری کنیم که روی تمام نرم افزار اعمال بشه باید چه کار کنیم؟
رویداد Application_Error در Global.asax محل مناسبی برای انجام این کار هست اما باید چند مسئله رو در نظر بگیریم:

Server.Transfer : این متد یک فایل (میتونه یک صفحه باشه)رو به خروجی میفرسته بدون اینکه آدرس تغییر کنه(کاربر به صفحه دیگه ای منتقل بشه) نکته ای که وجود داره اینه که برای انتقال نیاز به وجود یک فایل فیزیکی بر روی سیستم داره تا اون فایل رو به خروجی منتقل کنه ، در پروژه‌های MVC فایل فیزیکی (به شکلی که در ASP.NET وجود داره) وجود نداره و بوسیله routeها ، controllerها و view ها یک صفحه(خروجی) تولید میشه بنابراین ما نمیتونیم در ASP.NET MVC از این متد استفاده کنیم و باید به دنبال راه حلی برای فرستادن پاسخ مناسب باشیم.
Response.Redirect : این متد کاربر رو به یک صفحه‌ی دیگه منتقل میکنه و StatusCode رو برابر با 301 مقدار دهی میکنه که معنای انتقال صفحه هست و برای مشخص کردن "خطای داخلی سرور" (Internal Server Error) با کد 500 و پیدا نشدن فایل با کد 404 مناسب نیست .این کد(StatusCode) برای موتورهای جستجو اهمیت زیادی داره لیست کامل این کدها و توضیحاتشون رو میتونید در این آدرس مطالعه کنید.
Response.Clear : این متد باید فراخوانی بشه تا خروجی تولید شده تا این لحظه رو پاک کنیم.
Server.ClearError : این متد باید فراخوانی بشه تا از ایجاد صفحه زرد رنگ خطا جلوگیری کنه.
با توجه به نکات بالا، با طی کردن مراحل زیر میتونیم خطاهای ایجاد شده رو مدیریت کنیم.

بدست آوردن آخرین خطای ایجاد شده.
بدست آوردن کد خطای ایجاد شده.
ثبت خطا برای بررسی (ممکن هست شما برخی خطاها مثل پیدا نشدن فایل رو ثبت نکنید).
پاک کردن خروجی ایجاد شده تا این لحظه.
پاک کردن خطای ایجاد شده در سرور.
فرستادن یک خروجی مناسب بدون تغییر مسیر.

```
protected void Application_Error(object sender, EventArgs e)
{
    var error = Server.GetLastError();
    var code = (error is HttpException) ? (error as HttpException).GetHttpCode() : 500;

    if (code != 404)
    {
        // save log error.Message
    }

    Response.Clear();
    Server.ClearError();

    string path = Request.Path;
    Context.RewritePath(string.Format("~/Errors/Http{0}", code), false);
    IHttpHandler httpHandler = new MvcHttpHandler();
    httpHandler.ProcessRequest(Context);
    Context.RewritePath(path, false);
}
```

خروجی مناسب به کاربر از طریق از یکی از Action هایی انجام میشه که در ErrorsController هست. باید توجه داشته باشید که اگه در خود این ErrorsController خطایی رخ بده ، یک حلقه‌ی بی پایان بین این کنترلر و رویداد Application_Error اتفاق خواهد افتاد.

```
public class ErrorsController : Controller
{
    [HttpGet]
    public ActionResult Http404()
    {
        Response.StatusCode = 404;
        return View();
    }

    [HttpGet]
    public ActionResult Http500()
    {
        Response.StatusCode = 500;
        return View();
    }
}
```

نظرات خوانندگان

نویسنده: eli
تاریخ: ۱۳۹۱/۰۴/۲۵ ۱۱:۳۷

عالی بود مرسی ممنون

عنوان: **ارسال فایل در ASP.NET MVC و اعتبار سنجی سمت کاربر**
نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۴/۲۳ ۱۸:۱۶
آدرس: www.dotnettips.info
برچسب‌ها: ASP.Net, MVC, ASP.Net MVC

پیشنیازها:
- نحوه ارسال فایل به سرور توسط ASP.NET MVC
- نحوه اعتبار سنجی سمت سرور ارسال فایل‌ها

در ASP.NET MVC برای آپلود فایل‌ها عموما عنوان می‌شود که از تگ input به نحو زیر استفاده شود:

```
<input type="file" name="file" />
```

مشکلی که با این روش وجود دارد، عدم فعال شدن اعتبار سنجی سمت کاربر در حد مثلا «لطفا یک فایل را انتخاب کنید» است.
برای فعال سازی آن می‌توان از همان روش unobtrusive معرفی شده در ASP.NET MVC 3 به نحو زیر به صورت دستی استفاده کرد:

```
<input type="file" data-val="true" data-val-required="لطفا فایلی را انتخاب کنید" name="file" />
@Html.ValidationMessage("file")
```

در اینجا ویژگی‌های data-val و data-val-required به صورت خودکار اعتبار سنجی سمت کاربر را فعال خواهند کرد (شبیه به required field validator عمل می‌کنند).

از این نکته خصوصا در طراحی html helperهای سفارشی نیز می‌توان استفاده کرد.
برای مثال فرض کنید مشغول طراحی یک کنترل captcha هستید. قسمتی از آن مربوط به دریافت ورودی از کاربر است. به علاوه، پیغام خطایی هم که باید نمایش داده شود نیز باید توسط کاربر قابلیت سفارشی شدن را داشته باشد (و نمی‌توان آن‌را توسط یک attribute به سادگی به یک مدل خاص انتساب داد).
در این حالت تنظیم data-val و data-val-required به کمک anonymously typed objects پارامتر htmlAttributes میسر نیست (چون بین حروف آن dash وجود دارد) و باید از overload و نوع dictionary دار آن به نحو زیر استفاده کرد:

```
htmlHelper.TextBox("CaptchaInputText", string.Empty,
                           new Dictionary<string, object>
                           {
                             { "data-val", "true"},
                             { "data-val-required", validationErrorMessage},
                           })
```

نظرات خوانندگان

نویسنده: مولانا
تاریخ: ۱۳۹۲/۰۲/۱۷ ۱۷:۳۰

با سلام.
آیا امکان آپلود فایل بوسیله ajax نیز وجود دارد. و یا این روش معایبی دارد؟ با تشکر.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۲/۱۷ ۱۸:۵۴

بله. مرورگرهایی که XMLHttpRequest Level 2 را پشتیبانی کنند (از IE 10 به بعد مثلا)، امکان Ajax upload توکار به همراه گزارش درصد آپلود را بدون نیاز به فلش یا سیلورلایت، دارند. یک نمونه پیاده سازی آن

عنوان: **فعال سازی قسمت ارسال فایل و تصویر ویرایشگر آنلاین RedActor در ASP.NET MVC**
نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۵/۰۳ ۲۳:۲۰
آدرس: www.dotnettips.info
گروه‌ها: MVC, ASP.Net MVC, jQuery

در سایت جاری از ویرایشگر آنلاین RedActor استفاده شده و کار کردن با آن هم بسیار ساده است:
یک TextArea ساده را به صفحه اضافه کرده و این افزونه جی‌کوئری را بر روی آن اجرا می‌کنید. به این ترتیب TextArea به صورت خودکار تبدیل به یک ویرایشگر مطلوب خواهد شد. برای مثال:

```
@Html.TextAreaFor(model => model.ArticleBody, htmlAttributes: new { style = "width:98%; height:500px"
})

<script type="text/javascript">
$('#ArticleBody').redactor({
                autoformat: false,
                convertDivs: false
            });
</script>
```

اما فعال سازی قسمت ارسال فایل و تصویر همراه با آن چطور؟
یک سری مثال نوشته شده با PHP به همراه این ویرایشگر آنلاین هستند که برای ایده گرفتن بد نیستند (البته به این معنا نیست که این ویرایشگر نیازی به PHP دارد. تنها قسمت سمت سرور مثال‌های آن با PHP است). برای مثال اگر در PHP از دستور echo برای ارائه یک نتیجه نهایی به ویرایشگر RedActor استفاده شده، معادل آن در ASP.NET MVC مساوی return Content است.

```
<script type="text/javascript">
$('#ArticleBody').redactor({
                imageUpload: "@Url.Action(result: MVC.RedactorUpload.ImageUpload())",
                fileUpload: "@Url.Action(result: MVC.RedactorUpload.FileUpload())",
                linkFileUpload: "@Url.Action(result: MVC.RedactorUpload.FileLinkUpload())"
                , autoformat: false
                , convertDivs: false
            });
</script>
```

برای فعال سازی قسمت آپلود این ادیتور نیاز است پارامترهای fileUpload، imageUpload و linkFileUpload مقدار دهی شوند.
همانطور که ملاحظه می‌کنید از T4MVC برای م شخص سازی مسیرها استفاده شده. برای مثال
MVC.RedactorUpload.ImageUpload به این معنا است که در کنترلری به نام RedactorUpload، اکشن متدی به نام ImageUpload پذیرای ارسال فایل ادیتور خواهد بود و به همین ترتیب در مورد سایر پارامترها.
RedactorUploadController هم ساختار بسیار ساده‌ای دارد. برای مثال هر کدام از متدهای آپلود یاد شده یک چنین امضایی دارند:

```
[HttpPost]
public virtual ActionResult ImageUpload(HttpPostedFileBase file)
{
}
```

البته در مورد مسایل امنیتی آپلود هم پیشتر در سایت بحث شده است . برای مثال در اینجا استفاده از فیلتر زیر را فراموش نکنید:

```
[AllowUploadSpecialFilesOnly(".jpg,.gif,.png")]
```

در هر کدام از متدهای آپلود (به سه متد برای سه پارامتر یاد شده نیاز است)، ابتدا HttpPostedFileBase را در پوشه‌ایی که مدنظر دارید ذخیره کنید. سپس باید محتوایی را به RedActor بازگشت دهید و اصل کار یکپارچگی با ASP.NET MVC نیز در همینجا است:
در حالت imageUpload، محتوایی به شکل زیر باید بازگشت داده شود:

```
return Content("<img src='" + path + "' />");
```

در حالت fileUpload، پس از ذخیره سازی فایل در سرور، مسیر آن باید به نحو زیر بازگشت داده شود:

```
return Content("<a href=" + path + ">" + someName + "</a>");
```

و در حالت linkFileUpload فقط باید مسیر نهایی فایل ذخیره شده بر روی سرور را بازگشت دهید:

```
return Content(path);
```

همچنین باید دقت داشت که کار ارسال فایل به سرور توسط خود افزونه RedActor انجام می‌شود و نیازی به کدنویسی ندارد. فقط باید سمت سرور آن‌را به نحوی که عنوان شد مدیریت کنید. ابتدا فایل را در سرور ذخیره کنید. سپس باید یک محتوای رشته‌ای را به نحو یاد شده، ساخت و توسط return Content بازگشت داد.

پ.ن.
قسمتی از مطالب متن فوق در نگارش جدید این ویرایشگر به نحو زیر تغییر کرده است.

نظرات خوانندگان

نویسنده: امیرحسین
تاریخ: ۱۳۹۱/۰۵/۰۴ ۳:۹

آقای نصیری واقعا شرمنده کردی استاد!

واقعا انتظار نداشتم که به این سرعت پاسخ بدید.
یک دنیا ممنونم

نویسنده: امیرحسین
تاریخ: ۱۳۹۱/۰۵/۰۴ ۳:۳۹

توی این لینک و این هم در مورد تابع‌های سمت سرورش هست که با جمع شدن با این مطلب نیاز شما برطرف می‌شه.
البته این موضوع برای سی کی ادیتور نوشته شده که تفاوتی با این ادیتور نمی‌کنه.

نویسنده: Zahra
تاریخ: ۱۳۹۱/۰۵/۰۴ ۹:۵

سلام

آقای نصیری میشه طریقه استفاده از این ادیتور رو برای asp.net webform هم شرح بدهید؟

باتشکر

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۵/۰۴ ۱۰:۲۱

در وب فرم‌ها:
- یک TextBox را به صفحه اضافه کنید. TextMode آن باید MultiLine باشد تا تبدیل به TextArea شود. همچنین ClientID آن‌را هم مقدار دهی کنید تا بشود در jQuery ازش استفاده کرد.
- تمام توضیحات یکی است با این تفاوت که:
الف) return Content در اینجا می‌شود Response.Write
ب) بجای کنترلر شما از یک http handler می‌تونید استفاده کنید (فایل‌های ashx معروف)

```
public class Upload : IHttpHandler {
    public void ProcessRequest (HttpContext context) {
        HttpPostedFile uploads = context.Request.Files["upload"];
        //... save the file
        // return context.Response.Write(...)
        // and then  context.Response.End();
```

در اینجا context.Request.Files امکان دسترسی به فایل‌های آپلود شده را می‌دهد.
آن‌ها را ذخیره کنید. در آخر کار هم با context.Response.Write مواردی را که در مقاله فوق توضیح داده شد، بازگشت دهید.

نویسنده: ali ghomi
تاریخ: ۱۳۹۱/۰۵/۰۴ ۱۹:۱۳

سلام من در Action مقدار زیرو برمی‌گردونم ولی در RedActor نشون نمیده

ولی آپلود درست انجام میشه.

```
string fileName = Path.GetFileName(file.FileName);
string filePath = "~/App_Data/Uploads/Pic/" + file.FileName;
if (file.ContentLength > 0 && fileName != null)
{
string path = Path.Combine(Server.MapPath("~/App_Data/Uploads/Pic"), fileName);
file.SaveAs(path);
}

return Content("<img src='" + filePath + "' />");
```

```
return Content("<img src='" + path + "' />");
```

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۵/۰۴ ۱۹:۵۵

درسته. علت این است که پوشه App_Data محافظت شده است و از طریق فراخوانی لینک آن به صورت مستقیم قابل دسترسی نیست.
اگر فایل‌ها رو در این پوشه ارسال می‌کنید نیاز هست مثلا یک کنترلر دیگر به نام file درست کنید و کار این کنترلر فایل، دریافت نام تصویر و ارسال آن به صورت return File به مرورگر کاربر باشد.
یا اینکه کلا از یک پوشه معمولی استفاده کنید.
+
ضمن اینکه کاراکتر ~ برای تگ معمولی img مفهومی ندارد و نباید از آن در پایان کار استفاده شود.

نویسنده: محمد آزاد
تاریخ: ۱۳۹۱/۰۵/۱۱ ۲:۳۷

جناب نصیری من بعد از اجرای خط

```
return Content("<img src='" + filename + "' />");

//filename =C:\Users\mta\Documents\Visual Studio 2010\Projects\IAUG\GIAU\Images\Actors.png
```

یک ارور جاوا اسکریپتی دارم از این خط:

```
e.match(/\{.*\}/)[0]
```

که مقدارش نال هست.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۵/۱۱ ۸:۱۷

filename یک مسیر نسبی است (نسبت به ریشه سایت) مثلا:
GIAU/Images/Actors.png
یا مثلا (یک مسیر مطلق):
http://mysite.com/GIAU/Images/Actors.png

نویسنده: محمد آزاد
تاریخ: ۱۳۹۱/۰۵/۱۱ ۲۳:۲۹

جناب نصیری من کدم رو اصلاح کردم .البته بازم همین مشکل رو دارم

```
virtualpath = "Images/Main.jpg";
  return Content("<img src='" + virtualpath + "' />");
```

ولی بازهم تو کتابخونه جاوا اسکریپت RedActor از همون خطی که گفتم ارور میگیره

```
var c=e.match(/\{.*\}/)[0];

//e=<IMG src="Images/Main.jpg">

//e.match=null !
```

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۵/۱۱ ۲۳:۵۰

من این سطر **e.match** رو در سورس redactor (فایل **js**. redactor) پیدا نمی‌کنم. مطمئن هستید تداخلی در برنامه ندارید؟
ضمنا این ویرایشگر با نگارش‌های نهایی مرورگرها سازگاری خوبی دارد.

نویسنده: محمد آزاد
تاریخ: ۱۳۹۱/۰۵/۱۲ ۰:۱۵

بله درست گفتید اسم فایلش اینه

```
eval code[dynamic]
```

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۵/۱۲ ۰:۱۸

من از نسخه معمولی redactor استفاده می‌کنم.
اگر نیازی به فشرده سازی آن بود از این روش استفاده کنید: (^)

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۵/۱۲ ۹:۴۲

یک نکته تکمیلی رو اینجا اضافه کنم.
نحوه عملکرد RedActor در نگارش جدید آن با نگارش قبلی که من استفاده کردم، کاملا تفاوت کرده و قسمت‌هایی از مطلب فوق دیگر معتبر نیست.
RedActor اینبار به خروجی json نیاز دارد. برای مثال به فایل مثال image_upload.php آن دقت کنید. در نگارش جدید به این نوع خروجی نیاز دارد:

```
// displaying file
$array = array(
'filelink' => '/images/'.$filename
);

echo stripslashes(json_encode($array));
```

معادل آن در ASP.NET MVC به نحو زیر است:

```
var array = new { filelink = "Images/k1.jpg" };
return Json(array, System.Net.Mime.MediaTypeNames.Text.Plain, JsonRequestBehavior.AllowGet);
```

یک anonymously typed object دلخواه را مطابق array مثال PHP آن درست کنید و به Json result به نحو فوق ارسال کنید.
برای مابقی هم به همین ترتیب است. مثلا file_upload.php مثال آن اینبار به شکل زیر تعریف شده:

```
$array = array(
'filelink' => '/files/'.$_FILES['file']['name'],
'filename' => $_FILES['file']['name']
);

echo stripslashes(json_encode($array));
```

معادل ASP.NET MVC آن به نحو زیر است:

```
var array = new { filelink = "files/myfile.zip", filename="myfile.zip" };
return Json(array, System.Net.Mime.MediaTypeNames.Text.Plain, JsonRequestBehavior.AllowGet);
```

نویسنده: امیرحسین
تاریخ: ۱۳۹۱/۰۵/۱۹ ۱۱:۵۹

سلام
لطفا این اصلاحات رو توی اصل متن اصلاح کنید
چون همه کامنت‌ها را تا انتها نمی‌خونند! و حدود دو روز مثل من علاف بشوند!

نویسنده: donya
تاریخ: ۱۳۹۱/۰۶/۱۵ ۱۵:۳۳

سلام
آقای نصیری من از ادیتور redactor در استفاده کردم، محتوای ادیتور رو در دیتابیس ذخیره می‌کنم اما برای نمایش دادنش مشکل دارم کل تگهای Html رو تو خروجی نشون میده.
در asp.net از literal برای نمایش محتویات ادیتور استفاده می‌کردیم ولی در MVC لیترال وجود نداره و نمیدونم چطوری محتوای ادیتور رو نمایش بدم ممنون میشم اگه جواب بدین.
با تشکر

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۶/۱۵ ۱۵:۵۳

نکته‌اش استفاده از Html.Raw است که در قسمت پنجم سری MVC توضیح داده شده.

نویسنده: roya
تاریخ: ۱۳۹۱/۰۶/۱۶ ۱۵:۱

آقای نصیری من برای ذخیره یه متن با redactor به مشکل خوردم. Error ای به صورت زیر میدهد:

```
System.Web.HttpRequestValidationException: A potentially dangerous Request.Form value was detected from
the client (Body="<p>aaaaaaa...").
```

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۶/۱۶ ۱۷:۱۴

مراجعه کنید به قسمت 17 سری MVC سایت؛ قسمت «بررسی اعتبار درخواست (Request Validation) در ASP.NET MVC».

نویسنده: احمد احمدی

تاریخ: ۱۳۹۱/۰۶/۲۱ ۱۵:۳۸

در صورت امکان در مورد وضعیت لایسنس این ادیتور کمی توضیح بدید .
بنده تصمیم دارم به دو صورت از این ادیتور استفاده کنم ، یکی در یک پروژه‌ی شخصی ، بار دیگر هم در یک پروژه‌ی رایگان و متن باز .
وضعیت لایسنس در هریک از شرایط بالا به چه صورت است ؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۶/۲۱ ۱۶:۱۰

مطابق توضیحات این صفحه ، اگر نیاز به پشتیبانی خاصی از تیم مربوطه داشته باشید، نیاز است هزینه کرده و مثلا در ایمیل خودتون Paypal transaction number را نیز ارسال کنید تا به شما پاسخ دهند.

نویسنده: احمد احمدی
تاریخ: ۱۳۹۱/۰۶/۲۷ ۱۳:۵۸

لطفا در مورد دکمه‌ی ارسال کد کمی توضیح بدید .
آیا از پلاگین خاصی باید استفاده کنیم یا باید بصورت دستی این امکان رو اضافه کنیم؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۶/۲۷ ۱۴:۶

نیاز هست افزونه نویسی کنید. استاندارد نیست.

نویسنده: کیارش سلیمان زاده
تاریخ: ۱۳۹۱/۱۰/۲۰ ۱۲:۴۷

به نظر من دیگه نسخه رایگان این ادیتور وجود نداره! اگه امکانش هست میشه نسخه ای که شما دارید و رایگانه برای دانلود بذارید؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۰/۲۰ ۱۳:۱۱

آخرین نسخه‌ای که من دارم: redactor803.zip

نویسنده: کیارش سلیمان زاده
تاریخ: ۱۳۹۱/۱۰/۲۰ ۱۳:۱۵

ممنون وحید جان،
اینم هستش.

نویسنده: Mohsen
تاریخ: ۱۳۹۱/۱۱/۱۱ ۲۳:۲۹

با سلام
زمانی که از این ادیتور استفاده می‌کنیم آیا کاربر می‌تواند فونت را تغییر دهد؟ ممنونم

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۱/۱۱ ۲۳:۴۲

خیر. احتمالا به این علت که آنچنان مرسوم نیست از چندین قلم در وب استفاده شود، منهای CSS اصلی سایت و تعریف قلم‌های اصلی برای هدرها و امثال آن. به علاوه قلم متن نمایش داده شده در صفحات باید تابع CSS سایت باشد نه ادیتور آن.

ولی در کل می‌تونید براش افزونه بنویسید تا اینکار را انجام دهد.

نویسنده: Mohsen
تاریخ: ۱۳۹۱/۱۱/۱۱ ۲۳:۵۲

من هر کاری کردم که فونت پیشفرض اون رو تغییر دهم نتونستم این کار رو انجام بدم حتی تمام propertyهای مربوط به فونت رو از فایل Css حذف کردم ولی باز فونت خودش رو می‌گیره فکر کردم که شاید این کار امکان پذیر نباشد ولی شما این کار رو کردید . امکانش هست بفرمایید چگونه فونت پیشفرض این ادیتور رو تغییر دادید. ممنونم.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۱/۱۲ ۰:۶

من چیزی رو از فایل css آن حذف نکردم. فقط یک tahoma به اول تعاریف فونت‌های آن اضافه کردم.
به علاوه باید دقت داشت که تمام این ادیتورها در یک iframe بارگذاری می‌شوند. یعنی در حین نمایش، body آن‌ها تابع css خودشون است و نه css سایت. به همین جهت نیاز به کمی تغییر دارند:

```
body{
font-family:Tahoma,sans-serif;
font-size:9pt;
margin:0;padding:0;
overflow-x:hidden;
background:#fff;
direction:rtl
}
```

نویسنده: ناصر
تاریخ: ۱۳۹۱/۱۲/۲۰ ۱۹:۰

ببخشید آقای نصیری این نسخه ای که شما گذاشتید (این فایل zip) آیا خروجی json میخاد یا خروجی باید مثل مثال بالا باشه؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۲/۲۰ ۱۹:۹

روش تشخیص رو اینجا توضیح دادم. نیاز به json داره.

نویسنده: ناصر
تاریخ: ۱۳۹۱/۱۲/۲۰ ۱۹:۵۳

با عرض پوزش من در یک برنامه asp.net web form دارم از این ادیتور استفاده میکنم و نتونستم معادل دستورات زیر در asp.net رو بفهمم .

```
var array = new { filelink = "files/myfile.zip", filename="myfile.zip" };
return Json(array, System.Net.Mime.MediaTypeNames.Text.Plain, JsonRequestBehavior.AllowGet);
```

چیزی که خودم حدس زدم یک خروجی شبیه زیر بود :) اما مثل اینکه درست نبود

```
context.Response.Write("{'filelink'='" + filePath + "'}");
```

لطفا برای هر دو مورد فایل و تصویر راهنمایی بفرمایید چه خروجی json مدنظر هست.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۲/۲۰ ۲۱:۴۶

مراجعه کنید به مطلب آشنایی با کلاس JavaScriptSerializer

نویسنده: ناصر
تاریخ: ۱۳۹۱/۱۲/۲۱ ۲۲:۴۶

با تشکر فراوان از لطفتون. همچنین یک راهنمای جالب برای این سایت اینجاست. دوستانی که میخوان کمی تغییرات در این ادیتور بدن مثلا زبان فارسی اضافه کنن یا یک سری امکانات این چنینی حتما آدرس زیر رو ببین. چیزهای جالبی داخلش هست.
/http://imperavi.com/redactor/docs

همچنین داخل این راهنما خروجی هم دقیقا آئرده شده که من بعد از سوال پیدا کردم. خروجی برای image باید به شکل زیر باشه

```
{ "filelink": "/images/img.jpg" }
```

و برای فایل هم به شکل زیر باشه

```
{ "filelink": "/files/file.doc", "filename": "Filename" }
```

چیزی که خیلی وقت من رو گرفت اون اسلش قبل نام فایل بود که نمیزاشتم و خروجی مد نظر رو بم نمیداد. دوستان این قضیه رو حتما مد نظر داشته باشن تا مثل من وقتشون زیاد گرفته نشه :)

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۲/۲۱ ۲۲:۵۹

یک نکته: سعی نکنید این فرمت‌ها رو دستی درست کنید. علت این است که کلاس JavaScriptSerializer ذکر شده، زمانیکه var array ساخته شده رو دریافت می‌کنه، یک سری از کاراکترها رو هم escape می‌کنه .

نویسنده: بهروز
تاریخ: ۱۳۹۱/۱۲/۲۲ ۱۳:۶

آقای نصیری ظاهرا آخرین ورژن (redactor803) که شما گذاشتید با آخرین ورژن کنونی جیکوری (jquery-1.9.1.min.js) سازگاری ندارد و با زدن روی دکمه image پنجره ای برای ارسال باز نمی‌شود ولی با jquery-1.8.1.min.js تست کردم مشکلی نبود
آیا راهی برای حل این مشکل وجود دارد؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۲/۲۲ ۱۳:۱۳

- من دقیقا از آخرین نسخه رایگان این ادیتور استفاده می‌کنم. نیازی هم به نگارش‌های بعدی آن ندارم، چون نگارش مورد استفاده بدون مشکل کار می‌کند.
- همچنین از jquery-1.8.3.min.js برای کار با این نسخه استفاده می‌کنم.
- اگر نیاز به نگارش دیگری دارید بهتر است در انجمن تهیه کنندگان آن این مسایل رو مطرح کنید. البته ابتدا باید هزینه لایسنس نگارش‌های جدید آن‌را پرداخت کنید.
- بسیاری از افزونه‌های jQuery، با نگارش‌های جدید بعد از 1.9 آن سازگار نیستند و فقط این یک مورد نیست. بهتر است عجله نکنید و حداقل 6 ماهی برای ارتقاء صبر کنید.
- پروژه‌ای وجود دارد به نام jQuery Migrate برای پوشش مواردی که از جی‌کوئری 1.9 به بعد حذف شدن. این مورد رو باید به پروژه اضافه کنید تا با افزونه‌های قدیمی بتونید کار کنید.

نویسنده: میهمان
تاریخ: ۱۳۹۱/۱۲/۲۲ ۱۳:۳۳

سلام جناب آقای مهندس
در لینکی که در اول پست گذاشته اید ، وقتی دکمه Get Redactor رو میزنیم ، هیچ نسخه free وجود ندارد!
3 تا گزینه buy a licence وجود دارد و یک گزینه trial version
ممنون میشم اگر لینک مستقیم نسخه رایگان رو بگذارید
با سپاس

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۲/۲۲ ۱۳:۳۹

کمی بالاتر در بین نظرات ارسال شده.

نویسنده: مولانا
تاریخ: ۱۳۹۲/۰۲/۱۷ ۲۱:۵۴

با سلام.
من خاصیتی از ویومدل را بصورت زیر تعریف کردم:

```
[Required(ErrorMessageResourceType = typeof(ValidationErrorsResource),
            ErrorMessageResourceName = ResourceKeys.ValidationErrorsResource.RequiredField)]
        [MaxLength]
        [DisplayName("متن")]
        [AllowHtml]
        public string FullDescription { get; set; }
```

در ویو هم بصورت زیر:

```
<div>متن: </div>
                <div>@Html.TextAreaFor(p => p.FullDescription)</div>
                <div>@Html.ValidationMessageFor(p => p.FullDescription)</div>

$('#FullDescription').redactor({
            autoformat: false,
            convertDivs: false
        });
```

ولی وقتی کاربر حتی چیزی وارد نمیکند در دیتابیس مقدار "
 " ذخیره می‌شود و همین باعث میشود Required کار نکند و پیغامی به کاربر نمایش نمی‌دهد.
ایراد کارم از کجاست؟ باتشکر.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۲/۱۷ ۲۲:۴۴

اینطور نیست. مقداری که عنوان کردید به معنای new line وارد شده است که در مرورگرهای مختلف مقدار بومی آن متفاوت است.

نویسنده: مولانا
تاریخ: ۱۳۹۲/۰۲/۱۸ ۶:۳

با سلام
من پوشه ای بنام upload درست کردم و کد مربوط به کنترل من هم بصورت زیر است:

```
[HttpPost]
        [AllowUploadImagesOnly(".jpg,.gif,.png")]
```

```
        public virtual ActionResult ImageUpload(HttpPostedFileBase file)
        {
             var newFileName = Server.MapPath(Path.Combine(Links.SiteContents.Upload.Url(),
file.FileName));
            file.SaveAs(newFileName);
            var array = new { filelink = newFileName };
            return Json(array, MediaTypeNames.Text.Plain, JsonRequestBehavior.AllowGet);
        }
```

فایل به درستی درون پوشه آپلود قرار میگیرد ولی درون ادیتور نمایش داده نمی‌شود. علت چیست؟
خطایی هم دریافت نمیکنم.
باتشکر.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۲/۱۸ ۸:۵۴

علت این است که نباید مستقیما newFileName را بازگشت دهید. newFileName در اینجا چیزی است مثل c:\path\folder\site\images\img01.png. یعنی مسیر مطلق است که باید تبدیل به مسیر نسبی images\img01.png شود برای استفاده در src تگ img.

نویسنده: افشار محبی
تاریخ: ۱۳۹۲/۰۲/۱۸ ۱۲:۱۰

اگر نخواهیم از نسخه‌های قدیمی‌تر (که رایگان بودند) استفاده کنیم و همیشه به روز باشیم باید دست به دامان نسخه‌های کرک یا نسخه‌هایی شویم که از این طرف و اون طرف به صورت غیر قانونی برای دانلود گذاشته می‌شوند.

اگر این موضوع درست باشد پس نمی‌توان از RedActor به صورت به روز و بی‌دردسر استفاده کرد. در نتیجه شاید بهتر باشد از رقبای رایگان مثل CKEditor استفاده شود.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۲/۱۸ ۱۲:۲۲

سایت جاری از این ادیتور (نگارش قدیمی آن) استفاده می‌کند و به شخصه مشکلی با کیفیت آن ندارم و پاسخگوی نیاز کاری ما است.

نویسنده: افشار محبی
تاریخ: ۱۳۹۲/۰۲/۱۸ ۱۲:۳۵

آن طور که گفته می‌شود RedActor خیلی بهتر از CKEditor است. در همین سایت هم بدون مشکل در حال کار است. اما امکان به روز رسانی همیشگی باعث می‌شود خیال آدم از آینده راحت‌تر باشد.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۲/۱۸ ۱۲:۳۸

- هر طور صلاح می‌دونید.
- هدف از این بحث صرفا به اشتراک گذاشتن یک سری نکته بود که شاید هرجایی به این سادگی یافت نشود.

نویسنده: افشار محبی
تاریخ: ۱۳۹۲/۰۲/۱۸ ۱۲:۳۹

بله. مطمئناً بحث مفیدی بوده است. علاوه بر من، خیلی‌های دیگر استفاده کرده‌اند.

نویسنده: محمدرضا برنتی
تاریخ: ۱۳۹۲/۰۹/۱۴ ۱۲:۳۴

سلام آقای نصیری
امکان داره توضیح بدید چگونه دکمه Code و به این ادیتور اضافه کنیم ؟!

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۹/۱۴ ۱۳:۰

باید افزونه بنویسید. فایل paste_code.html آن در مسیر plugins:

```
<label>
    Code:</label>
<textarea id="redactor_insert_code_area" name="redactor_insert_code_area" style="height: 211px; width:
538px;" />
<label>
    Language:</label>
<select id="redactor_insert_code_lang">
    <option>CSharp</option>
    <option>VB</option>
    <option>JScript</option>
    <option>Sql</option>
    <option>XML</option>
    <option>CSS</option>
    <option>Java</option>
    <option>Delphi</option>
</select>
<br />
<input type="button" name="insert" id="redactor_insert_btn" value="%RLANG.insert%" />
```

و قسمت فعال سازی آن در فایل redactor.js ذیل setColorNone

```
showCodesPage: function () {
            this.modalInit('Insert Code', this.opts.path + '/plugins/paste_code.html', 600,
$.proxy(function () {
                var sel = this.getSelection();
                var currentCode = '';

                this.opts.codeElement = false;
                if ($.browser.msie) {
                    var parent = this.getParentNode();
                    if (parent.nodeName === 'PRE') {
                        this.opts.codeElement = parent;
                        currentCode = $(parent).text();
                    } else {
                        if (this.oldIE()) {
                            currentCode = sel.text;
                        } else {
                            currentCode = sel.toString();
                        }
                    }
                } else {
                    if (sel && sel.anchorNode && sel.anchorNode.parentNode.tagName === 'PRE') {
                        this.opts.codeElement = sel.anchorNode.parentNode;
                        currentCode = $(sel.anchorNode.parentNode).text();
                    } else {
                        currentCode = sel.toString();
                    }
                }

                if (this.opts.codeElement) {
                    $("#redactor_insert_btn").val("Update");
                }

                if (currentCode) $('#redactor_insert_code_area').val(currentCode);


                $('#redactor_insert_code_area').focus();
                $('#redactor_insert_btn').click($.proxy(this.insertCodesPage, this));

            }, this));
        },
        insertCodesPage: function () {
```

```
            var lang = $("#redactor_insert_code_lang").val();
            var code = $("#redactor_insert_code_area").val();
            code = code.replace(/\s+$/, ""); //rtrim;
            code = $('<span/>').text(code).html(); // encode

            this.$editor.focus();

            var preBlock;
            if (this.opts.codeElement) {
                preBlock = $(this.getParentNode());
            } else {
                preBlock = $("<pre/>");
            }
            preBlock.replaceWith("");

            var htmlCode = "<pre language='" + lang + "' name='code'>" + code + "</pre></div>";
            var codeBlock = "<div align='left' dir='ltr'>" + htmlCode + "</div><br/>";
            this.execCommand('inserthtml', codeBlock);

            this.modalClose();
        },
```

و بعد ثبت آن در فایل‌های public.js و default.js ذیل دکمه justify

```
code:
{
    title: 'Code',
    func: 'showCodesPage'
},
```

به css آن هم باید یک سطر ذیل را اضافه کنید:

```
body .redactor_toolbar li a.redactor_btn_code span          { background: url(../img/code_red.png) no-repeat center; }
```

نویسنده: محمدرضا برنتی
تاریخ: ۱۳۹۲/۰۹/۱۴ ۱۹:۱۳

در این خط :

```
this.modalInit('Insert Code', this.opts.path + '/plugins/paste_code.html', 600, $.proxy(function () {
```

مقدار

```
this.opts.path
```

برابر undefined هست و در نتیجه بعد از کلیک روی دکمه code فقط تب خالی نمایش داده می‌شه !
و متاسفانه از این مرحله بیشتر نتونستم پیش برم ...

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۹/۱۴ ۲۰:۰

ممکنه. چون نگارش‌های مختلفی ازش وجود داره و من از آخرین نگارش آن استفاده نمی‌کنم. برای مثال در نگارش‌های جدیدتر دنبال modal_table بگردید تا نحوه تعریف صفحه نمایش اطلاعات را به صورت توکار در همان فایل js نهایی، مشاهده کنید. کدهای ارسال شده با مقداری تغییر در نگارش‌های بعدی قابل استفاده هستند.

نویسنده: محمدرضا برنتی

تاریخ: ۱۳۹۲/۰۹/۱۵ ۱:۳۱

همین الان کارم تموم شد و تونستم دکمه code و با نگارش جدید این ادیتور هماهنگ کنم و در راه حل مشکلم تقریبا یاد گرفتم چجوری باید برای این ادیتور افزونه بنویسم ... و این بدون کمک شما امکان پذیر نبود ...
خیلی ممنونم ...

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۹/۱۵ ۱۰:۲۳

اگر دوست داشتید فایل نهایی خودتون رو پیوست کنید. ممنون.

نویسنده: پرویز
تاریخ: ۱۳۹۲/۰۹/۲۷ ۲۲:۲۷

سلام؛ این آموزش را به صورت زیر انجام دادم. فایل آپلود میشه اما تو ادیتور نمایش داده نمیشه. اگه دوستان لطف کنند راهنمایی کنند ممنون میشم. در ضمن از نسخه ای که تو بخش نظرات ارسال شده، استفاده کردم. با تشکر.

کدها کنترلر

```
public ActionResult FileUpload(HttpPostedFileBase file)
            {

                if (file.ContentLength > 0)
                {
                  string   filePath = Path.Combine(HttpContext.Server.MapPath("../uploade/"),
Path.GetFileName(file.FileName));
                     file.SaveAs(filePath);
                }
              return Content("<img src='/uploade/" + file.FileName + "' />");
            }
```

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۹/۲۷ ۲۲:۳۲

- نسخه جدیدتر return Json دارد، بجای return Content. کمی بالاتر بحث شده.
+ مسیر نسبی هم باید برای تصاویر بازگشت داده شود.

نویسنده: فرزین بیاتی
تاریخ: ۱۳۹۲/۱۲/۰۱ ۱۶:۵۰

با سلام؛ فایل های public.js و default.js کجا قرار دارن؟ چون توی اون ورژنی که خودتون واسه دانلود گذاشتید و گفتید که از اون استفاده میکنید، من فقط فایل redactor.js رو پیدا کردم!

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۲/۰۱ ۱۶:۵۳

در نگارش جدیدتر آن این فایل‌ها با همان فایل اصلی ادغام و یکی شده‌اند.

نویسنده: فرزین بیاتی
تاریخ: ۱۳۹۲/۱۲/۰۱ ۱۷:۱۷

ممنون از پاسختون
من از ورژن v8.0.3 استفاده میکنم
میشه لطفاً بیشتر توضیح بدید که مثلاً چه جوری از فایل paste_code.html باید استفاده کرد یا اینکه code:

```
{
```

```
    title: 'Code',
    func: 'showCodesPage'
},
```

رو باید کجا اضافه کرد؟ تو سایت خودش واسه نوشتن پلاگین از یه روش دیگه استفاده کرده

نویسنده: فرزین بیاتی
تاریخ: ۱۳۹۲/۱۲/۰۴ ۱۲:۵۶

سلام میشه لطف کنید مراحل اضافه کردن دکمه‌ی کد رو بفرمایید. با اون روشی که آقای نصیری توضیح داده بودن من نتونستم انجامش بدم.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۲/۰۴ ۱۳:۵۹

یک مثال کامل برای ارسال کد؛ کار آقای احمدی هست: RedActor-InsertCode.rar
کدهای آن‌را با کدهایی که کمی بالاتر ارسال کردم، تطابق داده و تکمیل کنید.

نویسنده: فرزین بیاتی
تاریخ: ۱۳۹۲/۱۲/۰۶ ۱۴:۳۷

دوستان فرق linkFileUpload و fileUpload چیه و با کدوم دکمه فعال میشه؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۲/۰۶ ۱۷:۳

دکمه insert link چند برگه‌ی مختلف دارد که یکی از آن‌ها مربوط به linkFileUpload است. fileUpload مستقلا دکمه‌ی سنجاق مانندی در نوار ابزار دارد.

نویسنده: فرزین بیاتی
تاریخ: ۱۳۹۲/۱۲/۱۴ ۱۵:۳۳

چطوری میشه redactor رو readonly کرد؟ من readonly و disabled رو به TextArea اعمال کردم که جواب نداد.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۲/۱۴ ۱۶:۱۳

برای نمایش فقط خواندنی یک متن HTML از Html.Raw استفاده کنید؛ مانند نمایش متن یک مقاله در یک سایت.

نویسنده: جوادنبی
تاریخ: ۱۳۹۲/۱۲/۲۵ ۱۸:۵۳

این کد من که نوشتمش و جواب گرفتم.

```
[HttpPost]
public ActionResult ImageUpload(HttpPostedFileBase file)
{
   string path = "";
   var FileName = "";
   if (file.ContentLength > 0)
   {
     FileName = Path.GetFileName(file.FileName);
     string Extension = file.ContentType;
     string[] ExtensionList = { "image/jpg", "image/jpeg", "image/gif", "image/pmg" };
     if (ExtensionList.Contains(Extension.ToLower()))
     {
```

```
            path = Path.Combine(Server.MapPath("~/Images/uploads"), FileName);
            file.SaveAs(path);
        }
    }

    var array = new { filelink = @"Images\uploads\" + FileName };
    return Json(array, System.Net.Mime.MediaTypeNames.Text.Plain, JsonRequestBehavior.AllowGet);
}
```

امید وارم به کارت بیاد.

نویسنده: پویا امینی
تاریخ: ۱۳۹۲/۱۲/۲۶ ۲۲:۲۲

با سلام ، من می‌خواستم قسمت ارسال فایل را همانند شما فعال کنم مطابق کدی که فرمودید تنظیمات رو انجام دادم ولی هنگام انتخاب عکس از قسمت زیر

```
if (this.uploadOptions.success)
{
if (typeof d !== 'undefined')
{
// Remove bizarre <pre> tag wrappers around our json data:
var rawString = d.body.innerHTML;
var jsonString = rawString.match(/\{.*\}/)[0];
this.uploadOptions.success(jsonString);
}
```

خط

```
var jsonString = rawString.match(/\{.*\}/)[0];
```

خطای زیر رو مگیره

```
0x800a138f - JavaScript runtime error: Unable to get property '0' of undefined or null reference
```

ممنون میشم راهنمایی کنید

نویسنده: جوادنبی
تاریخ: ۱۳۹۲/۱۲/۲۷ ۲۳:۱۰

با سلام؛ من یک ویو دارم که در آن از دوتا redactor استفاده کردم. در واقع سناریوی من به این شکل است. یکی از این redactor برای ورود خلاصه‌ی متن است و در زیر آن یک لینک ایجکسی وجود دارد که میره و یک partial را می‌یاره. توی partial اونredactor دومی قرار دارد و جالب وقتی که من یک فایلی را در red actor که در پارشال هست آپلود می‌کنم فایل آپلود می‌شود ولی هنگامی که در صفحه‌ی سایتم می‌خوام دانلود کنم Error زیر را می‌دهد.
.The resource you are looking for has been removed, had its name changed, or is temporarily unavailable
حالا باید چه کار کنم؟ ممنون

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۲/۲۷ ۲۳:۳۶

لینک تولید شده نهایی در اکشن متد را با لینک واقعی فایل بر روی سرور تطابق دهید.

نویسنده: پرویز
تاریخ: ۱۳۹۳/۱۰/۱۶ ۱۲:۳۴

سلام
فرض کنید یه تعداد عکس را با این ویرایشگر آپلود کردیم ، بعد یه تعداد از عکس رو از درون ویرایشگر حذف کردیم، چون از درون ویرایشگر مسیر عکس‌ها حذف میشه خود عکس رو هاست باقی می‌مونه.
سوال: چطوری میشه با حذف عکس از درون ویرایشگر خود عکس هم از روی هاست پاک بشه؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۱۰/۱۶ ۱۲:۳۸

چنین قابلیتی را ندارد. برای این مورد از Froala WYSIWYG Editor استفاده کنید (^).

عنوان: **سازگار کردن لینک‌های قدیمی یک سایت با ساختار جدید آن در ASP.NET MVC**
نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۵/۱۴ ۲۱:۲۴
آدرس: www.dotnettips.info
برچسب‌ها: ASP.Net, MVC, ASP.Net MVC

اگر پیشتر سایتی را در آدرس مشخصی در اینترنت داشته‌اید و اکنون تنها نرم افزار آن تغییر کرده است، اما نحوه ارائه خدمات آن خیر، لازم است بتوانید شرایط ذیل را مدیریت کنید:
- موتورهای جستجو مدام اطلاعات قبلی خود را به روز می‌کنند. اگر آدرس قبلی مقاله‌ای در سایت شما http://site/year/month/day/title بوده، برای نمونه گوگل هر از چندگاهی مجددا به این آدرس مراجعه می‌کند تا حداقل مطمئن شود وجود خارجی دارد یا خیر (این نکته را از لاگ‌های خطای سایت استخراج کردم).
- سایت‌های زیادی هستند که پیشتر به سایت شما و مطالب آن لینک داده‌اند. نمی‌توانید از آن‌ها درخواست کنید لطفا بانک اطلاعاتی خود را به روز کنید.
- اگر فید قبلی سایت شما http://site/feeds/posts بوده و اکنون چیز دیگری است، باز هم نمی‌توانید از همه درخواست کنید اطلاعات خود را به روز کنند. عده‌ای اینکار را خواهند کرد و تعداد زیادی هم خیر.

برای مدیریت یک چنین مواردی می‌توان از امکانات مسیریابی موجود در ASP.NET MVC استفاده کرد؛ که نمونه‌ای عملی از آن‌را جهت سازگاری سایت جاری با هاست قبلی آن (بلاگر) در ادامه مطالعه خواهید نمود:

الف) سازگار سازی لینک‌های قدیمی برچسب‌های سایت با ساختار جدید آن

در بلاگر آدرس‌های برچسب‌ها، به صورت http://site/search/label/name تعریف شده است. در سایت جاری برچسب‌ها توسط کنترلر Tag مدیریت می‌شوند. برای هدایت آدرس‌های قدیمی (موجود در موتورهای جستجو یا ثبت شده در سایت‌هایی که به ما لینک داده‌اند) می‌توان از تعریف مسیریابی ذیل در فایل global.asax استفاده کرد:

```
routes.MapRoute(
                "old_bloger_tags_list", // Route name
                "search/label/{name}", // URL with parameters
                new { controller = "Tag", action = "Index", name = UrlParameter.Optional, area = "" }
// Parameter defaults
            );
```

به این ترتیب به صورت خودکار تمامی آدرس‌های شروع شده با http://site/search/label پالایش شده و سپس قسمت name آن‌ها جدا سازی می‌شود. این نام به متدی به نام Index در کنترلر Tag که دارای پارامتری به نام name است ارسال خواهد شد.

ب) از دست ندادن خوانندگان قدیمی فیدهای سایت

دو نوع فید کلی در بلاگر وجود دارد: http://site/feeds/posts/default و http://site/feeds/comments/default؛ اما در سایت جاری فیدها توسط کنترلری به نام Feed ارائه می‌شوند. برای سازگار سازی آدرس‌های قدیمی و هدایت آن‌ها به صورت خودکار به کنترلر فید می‌توان از دو تعریف مسیریابی ذیل استفاده کرد:

```
routes.MapRoute(
                "old_bloger_posts_feeds_list", // Route name
                "feeds/posts/default", // URL with parameters
                new { controller = "Feed", action = "Posts", name = UrlParameter.Optional, area = "" }
// Parameter defaults
            );

routes.MapRoute(
                "old_bloger_comments_feeds_list", // Route name
                "feeds/comments/default", // URL with parameters
                new { controller = "Feed", action = "Comments", name = UrlParameter.Optional, area = ""
} // Parameter defaults
            );
```

در اینجا دو آدرس ذکر شده به کنترلر Feed و متدهای Posts و Comments آن هدایت خواهند شد و به این نحو کاربران قدیمی سایت هیچگونه تغییری را احساس نکرده و باز هم فیدخوان‌های آن‌ها، بدون مشکل کار خواهند کرد.

ج) پردازش لینک‌های قدیمی مطالب سایت و هدایت آن‌ها به آدرس‌های جدید

این مورد اندکی مشکل‌تر از موارد قبلی است:

```
routes.MapRoute(
                "old_bloger_post_urls",
                "{yyyy}/{mm}/{title}",
                new { controller = "Post", action = "OldBloggerLinks" },
                new { yyyy = @"\d{4}", mm = @"\d{1,2}" }
            );
```

برای نمونه آدرس مقاله‌ای مانند http://site/2012/05/ef-code-first-15.html را درنظر بگیرید. سه قسمت سال، ماه و عنوان آن، حائز اهمیت هستند. این‌ها را در اینجا به کنترلر Post و متد OldBloggerLinks آن هدایت خواهیم کرد. همچنین برای سال و ماه آن نیز قید تعریف شده است. سال عددی 4 رقمی است و ماه عددی یک تا دو رقمی.

کدهای متد OldBloggerLinks را در اینجا مشاهده می‌کنید:

```
public virtual ActionResult OldBloggerLinks(int yyyy, int mm, string title)
        {
            var oldUrl = string.Format(CultureInfo.InvariantCulture,
"http://www.dotnettips.info/{0}/{1}/{2}", yyyy, mm.ToString("00"), title);
            var blogPost = _blogPostsService.FindBlogPost(oldUrl);
            if (blogPost != null)
                return RedirectToActionPermanent(actionName: ActionNames.Index, controllerName:
MVC.Post.Name,
                                                 routeValues: new { id = blogPost.Id, name =
blogPost.Title.GetPostSlug() });
            return this.Redirect("/");
        }
```

در اینجا چون ساختار لینک‌ها کلا تغییر کرده است، ابتدا بر اساس پارامترهای دریافت شده، لینک قدیمی بازسازی می‌شود. سپس به بانک اطلاعاتی مراجعه شده و لینک قدیمی به همراه شماره مطلب مرتبط با آن یافت می‌شود (یک فیلد oldUrl برای مطالب قدیمی در بانک اطلاعاتی وجود دارد). در آخر هم به کمک متد RedirectToActionPermanent آدرس رسیده به آدرس جدید مطلب در سایت ترجمه و هدایت خواهد شد. Permanent بودن آن برای به روز رسانی خودکار اطلاعات موتورهای جستجو مفید است.

نتیجه گیری

به کمک امکانات مسیریابی توکار ASP.NET MVC می‌توان ساختار قدیمی یک سایت را به ساختار جدید آن ترجمه کرد. به این ترتیب لینک‌های قدیمی ثبت شده در صدها سایت اینترنتی که به سایت ما اشاره می‌کنند، مجددا بدون مشکل قابل استفاده بوده و همچنین موتورهای جستجو نیز امکان به روز رسانی اطلاعات خود را خواهند یافت.

عنوان: CAPTCHAfa
نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۵/۱۵ ۱۲:۴۶
آدرس: www.dotnettips.info
گروه‌ها: CAPTCHA, reCAPTCHA, CAPTCHAfa, ASP.Net, MVC, ASP.Net MVC

حتماً با CAPTCHA آشنا هستید. فرایندی که در طی آن متنی نمایش داده می‌شود که عمدتاً فقط یک انسان قادر به درک و پاسخگویی به آن است. با این کار از ارسال داده‌های بیهوده توسط ربات‌ها جلوگیری می‌شود.
reCAPTCHA ایده‌ای است که با نمایش کلمات واقعی و اسکن شده از کتاب‌های قدیمی، بخشی از مشکلات را حل کرده و از کاربران اینترنت برای شناسایی کلماتی که رایانه توانایی خواندن آن‌ها را ندارد استفاده می‌کند. (شکل زیر)

با وارد کردن درست هر کلمه، بخشی از یک کتاب، روزنامه، و یا مجله‌ی قدیمی در رایانه شناسایی و به فرمت دیجیتال ذخیره می‌شود. به این شکل شما در دیجیتالی کردن متون کاغذی سهیم هستید.
پروژه‌ی reCAPTCHA توسط گوگل حمایت می‌شود و در این آدرس قرار دارد.
به تازگی تیمی از دانشکده فنی دانشگاه تهران به همراه انستیتو تکنولوژی ایلینویز شیکاگو، پروژه ای بر همین اساس اما برای متون فارسی با عنوان CAPTCHAfa تولید کرده اند (شکل زیر) که در این آدرس در دسترس است. امیدوارم این پروژه به گونه ای تغییر کنه که برای دیجیتالی کردن متن‌های پارسی استفاده بشه. در حال حاضر، این پروژه از کلماتی از پیش تعریف شده استفاده می‌کنه.

متاسفانه این پروژه در حال حاضر فقط توسط برنامه‌های PHP قابل استفاده است. از این رو بر آن شدم تا اون رو برای برنامه‌های ASP.NET (هم Web Forms و هم MVC) آماده کنم. برای استفاده از CAPTCHAfa نیاز به یک کلید خصوصی و یک کلید عمومی دارید که از این آدرس قابل دریافت است.
کدهای پروژه‌ی Class Library به شرح زیر است.

```
// =-=-=-=
```

```
            Stream dataStream = null;
            WebResponse response = null;
            string responseFromServer = string.Empty;

            try
            {
                WebRequest request = WebRequest.Create(CAPTCHAFA_VERIFY_SERVER);

                request.Method = "POST";
                request.ContentType = "application/x-www-form-urlencoded";

                byte[] byteData = Encoding.UTF8.GetBytes(data);

                request.ContentLength = byteData.Length;

                dataStream = request.GetRequestStream();

                dataStream.Write(byteData, 0, byteData.Length);

                dataStream.Close();

                response = request.GetResponse();

                dataStream = response.GetResponseStream();

                reader = new StreamReader(dataStream);

                responseFromServer = reader.ReadToEnd();
            }
            finally
            {
                if (reader != null)
                {
                    reader.Close();
                }

                if (dataStream != null)
                {
                    dataStream.Close();
                }

                if (response != null)
                {
                    response.Close();
                }
            }

            return responseFromServer;
        }
    }
}
```

استفاده در پروژه‌های ASP.NET Web Forms

ابتدا ارجاعی به فایل Captchafa.dll ایجاد کنید و سپس در روال Page_Load، کد زیر را قرار دهید. این کار برای تزریق اسکریپ CAPTCHAfa به صفحه استفاده می‌شود.

```
if (!IsPostBack)
{
    litCaptcha.Text = Captchafa.CaptchafaGetHtml();
}
```

litCaptcha، یک کنترل Literal است که اسکریپت تولید شده، به عنوان متن آن معرفی می‌شود.

بررسی صحت مقدار وارد شده توسط کاربر (مثلاً در روال Click یک دکمه) به صورت زیر است.

```
Captchafa captchaFa = new Captchafa();

bool isAnswerCorrect = captchaFa.IsAnswerCorrect();

if (isAnswerCorrect)
{
    // پاسخ صحیح است
}
else
{
   // پاسخ صحیح نیست
```

```
}
```

استفاده در پروژه‌های ASP.NET MVC

ابتدا ارجاعی به فایل Captchafa.dll ایجاد کنید. در ASP.NET MVC بهتره تا فرایند کار رو در یک HTML helper کپسوله کنیم.

```
public static class CaptchaHelper
{
    public static MvcHtmlString Captchafa(this HtmlHelper htmlHelper)
    {
        return MvcHtmlString.Create(Captcha.Captchafa.CaptchafaGetHtml());
    }
}
```

بررسی صحت مقدار وارد شده توسط کاربر (پس از ارسال فرم به Server) به صورت زیر است.

```
[HttpPost]
[ActionName("Index")]
public ViewResult CaptchaCheck()
{
    Captchafa captchaFa = new Captchafa();

    bool isAnswerCorrect = captchaFa.IsAnswerCorrect();

    if (isAnswerCorrect)
    {
        ViewBag.IsAnswerCorrect = true;
    }
    else
    {
        ViewBag.IsAnswerCorrect = false;
    }

    return View();
}
```

و در نهایت، کدهای View (از سینتکس موتور Razor استفاده شده است).

```
@using CaptchafaDemoMvc.Helpers;

@{
    ViewBag.Title = "Index";
}

<form action="/" method="post">

@Html.Captchafa();

<input type="submit" id="btnCaptchafa" name="btnCaptchafa" value="آزمایش" />

@{
    bool isAnswerExists = ViewBag.IsAnswerCorrect != null;
}

@if (isAnswerExists)
{
    if ((bool)ViewBag.IsAnswerCorrect == true)
    {
       <span id="lblResult">پاسخ صحیح است</span>
    }
    else
    {
       <span id="lblResult">پاسخ صحیح نیست</span>
    }
}
</form>
```

دموی پروژه رو در این آدرس قرار دادم. پروژه‌ی نمونه نیز از این آدرس قابل دریافت است.

پ.ن: به زودی برخی بهبودها رو بر روی این پروژه انجام میدم.

نظرات خوانندگان

نویسنده: امیرحسین جلوداری
تاریخ: ۱۳۹۱/۰۵/۱۵ ۱۵:۹

اول دیدم هر چی تست میکنم میگه صحیح نیست! ... بعد دیدم باید فاصله باشه بین دو کلمه که صحیح است! در صورتی که تو recaptcha اصلی فاضله اهمیت نداره! ... چرا اینجوریه؟!

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۵/۱۵ ۱۵:۳۸

چون در زبان فارسی، حروف وقتی به هم بچسبن، شکلشون تغییر می‌کنه. بنابراین طبیعیه که "مسلسل" با "مسل سل" فرق کنه.

نویسنده: ایمان نعمتی
تاریخ: ۱۳۹۱/۰۵/۱۵ ۱۵:۴۰

کار جالبی کردی بهروز جان
اما به نظر میرسه خود پروژه خیلی جالب نباشه
اینکه از یه سری کلمات از پیش تعیین شده استفاده میکنه و برای نمایش هم فقط دوتا خط به شکل بیضی روی کلمات میندازه زیاد جالب نیست به نظرم. شبیه به ری‌کپتچا خواسته کار کنه اما خب چیز خوبی از کار درنیامده

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۵/۱۵ ۱۵:۵۲

برای شروع، کار بسیار ارزشمندیه. ضمن اینکه تنوع کلمات مورد استفاده بسیار بالاست. خیلی مشتاقم که زودتر برای دیجیتالی کردن متون فارسی بشه ازش استفاده کرد، اما نیاز به یک متولی قوی و پشتیبانی خوب داره. مثل کاری که سایت گنجور انجام میده.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۵/۱۵ ۱۵:۵۳

نوشتن مشابه آن با دات نت زیاد سخت نیست. برای نمونه « بررسی نحوه برنامه نویسی سایت نستعلیق آنلاین » می‌تونه ایده خوبی برای شروع باشه. من این مطلب رو به همراه مطلب « تبدیل عدد به حروف » کنار هم گذاشتم و یک captcha فارسی ازش تهیه کردم.

نویسنده: ایمان نعمتی
تاریخ: ۱۳۹۱/۰۵/۱۵ ۱۹:۵۳

چه جالب!
امیدوارم استفاده از کپتچای فارسی در سایت‌های فارسی مرسوم بشه

نویسنده: علیرضا اسم‌رام
تاریخ: ۱۳۹۱/۰۵/۱۶ ۹:۵۶

با سلام. جهت حل مشکل کلمات از پیش تعریف شده، می‌توان از Google Translate استفاده کرد.
متاسفانه هنوز فرصت نکردم تا جزییات را دقیق ببینم، اما امیدوارم تلفظ و پخش کلمات بصورت صوتی خوب عمل کند!

نکته دیگر اینکه در مورد پیچیدگی تصویر CAPTCHA، بهترین روش استفاده از نقطه و پاره خط جهت تشویش تصویر است. معمولاً

بیشترین خطاها در OCR مربوط به این دو شکل پایه‌ی هندسی هستند. البته این موضوع بیشتر در مورد کلمات لاتین مطرح است.
شاید در مورد کلمات فارسی بیضی هایی که دوستان روی متن انداختند پیشچیدگی را افزایش دهد. آقای ایمان نعمتی لطفاً فراموش نکنیم که هدف از تشویش تصویر، سختی کار برای انسان نیست بلکه پیچیده کردن کار روبات است.
در مجموعه بسیار موضوع خوبی است. اگر دوستانی علاقه به این موضوع داشته باشند ایده و مطلب بسیار خوبی وجود دارد، هرچند پیاده سازی نشده است اما فکر می‌کنم ارزش انتشار و مطالعه را داشته باشد. البته جهت پیاده سازی آن چالشی بر سر راه نیست و فقط کمی وقت آزاد و البته تخصص نیاز دارد.

این مطلب را در این سایت میشه منتشر کرد؟ (با توجه به اینکه هنوز پیاده سازی ندارد و صرفاً بیان الگوریتم است.)

نویسنده: هوشنگ
تاریخ: ۱۳۹۱/۰۵/۱۸ ۱۱:۱۱

"با وارد کردن درست هر کلمه، بخشی از یک کتاب، روزنامه، و یا مجله‌ی قدیمی در رایانه شناسایی و به فرمت دیجیتال ذخیره می‌شود. به این شکل شما در دیجیتالی کردن متون کاغذی سهیم هستید. "

این منطقی نیست ، با همان شیوه ای که کلمه تایپ شده کاربر با عکس مقایسه میشه ، به همان طریق میشه متون کاغذی رو دیجیتالی کرد و نیازی به استفاده از این شیوه برای دیجتالی کردن نیست

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۵/۱۸ ۱۲:۴

دوست من.
تمامی کلمات در یک متن (مخصوصاً اگر اون متن قدیمی باشه و برخی حروف یک کلمه پاک یا ناخوانا باشند) توسط رایانه و برنامه‌های OCR قابل تشخیص نیست.
مدل‌های CAPTCHAیی مانند reCAPTCHA که بر اساس دیجیتالی کردن متون پیاده سازی شدند، دو کلمه رو به کاربر نشان میدن:
1) کلمه ای که توسط رایانه به درستی تشخیص داده شده
2) کلمه یا بخشی از اون که رایانه نتونسته اون رو به درستی تشخیص بده.
در صورتی که کاربر، کلمه ای که توسط رایانه تونسته تشخیص داده بشه و برای رایانه مشخص هست رو صحیح وارد کنه، فرض بر این گرفته میشه که کلمه‌ی دوم یا بخشی از کلمه‌ی دوم که قابل تشخیص نبوده رو هم تونسته به درستی وارد کنه.
پس از این، میانگینی از تعداد عبارت ورودی کاربران برای کلمه‌ی نامشخص گرفته میشه و عبارتی که بیشترین فراوانی رو داشته به عنوان مورد قابل قبول ثبت میشه.

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۵/۲۸ ۱۲:۵۰

کامپوننت رو آپدیت کردم. نسخه‌ی اولیه بر اساس وفاداری به کدهای PHP و برگردانی از اونها بود. در نسخه‌ی جدید، متد CaptchafaGetHtml به GetHtml تغییر پیدا کرد، مدیریت خطاها اضافه شد و کلیدهای خصوصی و عمومی باید در زمان استفاده به کامپوننت پاس داده شوند. نسخه‌ی جدید را از لینک انتهای مقاله دریافت کنید.

نویسنده: بابک
تاریخ: ۱۳۹۳/۰۸/۲۸ ۱۵:۲۳

آدرس ظاهرا به آدرس زیر تغییر کرده است:
http://www.captchafa.net

عنوان: **اهمیت Controller های ساده در ASP.NET MVC**
نویسنده: شاهین کیاست
تاریخ: ۱۳۹۱/۰۵/۳۱ ۱۴:۳۳
آدرس: www.dotnettips.info
گروه‌ها: Refactoring, ASP.Net MVC

Controllerها به نوعی رابط بین View و Model هستند. *ساده ترین* محل برای قرار دادن کدهای تصمیم گیری (decision-making code) ، قرار دادن منطق تجاری و یا فراهم ساختن داده برای View مثل ایجاد یک لیست از Select List برای یک DropDownList می‌باشند. اما انجام این کارها به نرم افزار ما پیچیدگی تحمیل می‌کند. Controllerها باید در طول زمان توسعه‌ی یک نرم افزار کم حجم و سبک باقی بمانند. در این مطلب بحث شد که یکی از اهداف استفاده از ASP.NET MVC نوشتن نرم افزار هایی تست پذیر می‌باشد. نوشتن آزمون واحد و نگهداری Controller هایی که مسئولیت زیادی بر عهده دارند سخت می‌باشد. Controllerها باید به نحوی توسعه پیدا کنند که نگهداری آن‌ها ساده باشد ، تست پذیر باشند و اصل SRP را رعایت کرده باشند.

نگهداری آسان

کدی که درک آن سخت باشد ، تغییر دادن آن نیز سخت است ، هنگامی که درک کدی سخت باشد زمینه برای به وجود آمدن خطاها ، دوباره کاری و سردرد فراهم می‌شود. هنگامی که مسئولیت یک Action به صورت شفاف مشخص نباشد و انواع و اقسام کارها به آن سپرده شده باشد تغییر در آن سخت می‌شود ، ممکن است این تغییر باید چند جای دیگر هم داده شود در نتیجه فاز نگهداری هزینه و زمان اضافی به نرم افزار تحمیل می‌کند.

تست پذیری

بهترین راه برای اطمینان از این که درک و تست پذیری سورس کد ما ساده هست انجام و تمرین توسعه‌ی تست محور (TDD) می‌باشد. هنگامی که از روش TDD استفاده می‌کنیم با سورسی کدی کار می‌کنیم که هنوز وجود ندارد. در این مرحله از به وجود آمدن کلاس هایی که تست آن‌ها دشوار یا غیر ممکن است (به دلیل داشتن مسئولیت‌های اضافی) از جمله Controllerها جلوگیری می‌شود. نوشتن آزمون‌های واحد برای Controllerهای کم حجم ساده می‌باشد.

تمرکز بر روی یک مسئولیت

بهترین راه برای ساده سازی Controllerها گرفتن مسئولیت‌های اضافی از آن می‌باشد به Action زیر توجه کنید :

```
public RedirectToRouteResult Ship(int orderId)
{
   User user = _userSession.GetCurrentUser();
   Order order = _repository.GetById(orderId);
   if (order.IsAuthorized)
   {
      ShippingStatus status = _shippingService.Ship(order);
      if (!string.IsNullOrEmpty(user.EmailAddress))
      {
         Message message = _messageBuilder
            .BuildShippedMessage(order, user);
         _emailSender.Send(message);
      }
      if (status.Successful)
      {
         return RedirectToAction("Shipped", "Order", new {orderId});
      }
   }
   return RedirectToAction("NotShipped", "Order", new {orderId});
}
```

این Action کار زیادی انجام می‌دهد ، تقریبا می‌توان تعداد مسئولیت‌های این Action را با شمارش تعداد Ifها پیدا کرد . مسئولیت تصمیم گیری درباره این که آیا Order مورد نظر آماده برای تحویل است یا خیر به این Action سپرده شده ، همچنین تصمیم گیری درباره اینکه آیا کاربر دارای آدرس ایمیل جهت ارسال ایمیل می‌باشد یا خیر به این Action سپرده شده است. منطق‌های domain logic و business logic نباید در کلاس‌های presentation همانند Controllerها قرار داده شود. تست و نگهداری کدی مثل کد بالا دشوار خواهد بود. Refactoring که باید در این Code اعمال شود Refactor Architecture by Tiers نام دارد. این Refactoring به توسعه دهنده می‌گوید که منطقی که در لایه‌ی نمایش (Presentation) پیاده کرده را به لایه‌ی Business انتقال دهد. پس از انتقال منطق کد بالا به OrderShippingService ، کد Action ما ساده‌تر می‌شود :

```
public RedirectToRouteResult Ship(int orderId)
{
   var status = _orderShippingService.Ship(orderId);
   if (status.Successful)
```

```
    {
        return RedirectToAction("Shipped", "Order", new {orderId});
    }
    return RedirectToAction("NotShipped", "Order", new {orderId});
}
```

پس از انتقال منطق تجاری به محل مناسب خودش تنها مسئولیتی که برای برای Controller باقی مانده این است که کاربر را به کجا Redirect کند. پس از این Refactoring علاوه برا اینکه مسئولیت‌ها در جای مناسب خود قرار گرفتند ، اکنون می‌توان به سادگی منطق کار را بدون تحت تاثیر قرار گرفتن کدهای لایه‌ی نمایش تغییر داد.
در آینده به تکنیک‌های ساده سازی Controllerها خواهیم پرداخت.

نظرات خوانندگان

نویسنده: محمد صاحب
تاریخ: ۱۳۹۱/۰۵/۳۱ ۱۸:۹

ممنون موضوع جالبیه...
فصل 7وم کتاب Programming Microsoft ASP.NET MVC هم به همین موضوع اختصاص داده شده که برای سبک کردن کنترلر دوتا الگوی Responsibility-Driven Design و iPODD رو معرفی میکنه و...

نویسنده: شاهین کیاست
تاریخ: ۱۳۹۱/۰۵/۳۱ ۱۸:۱۳

ممنون می‌شم اگر درباره الگوی iPODD لینک بدید.

به این موضوع در سری کتاب‌های MVC In Action هم یک فصل کامل اختصاص داده شده .

نویسنده: حسین مرادی نیا
تاریخ: ۱۳۹۱/۰۵/۳۱ ۲۰:۳۵

ممنون
واقعا لذت بردیم
امیدوارم این مباحث رو همچنان ادامه بدید ...

نویسنده: رضا بزرگی
تاریخ: ۱۳۹۱/۰۵/۳۱ ۲۰:۳۶

آیا در معماری چندلایه (N-Tier arch) مرزبندی شفافی وجود دارد که از نظر شی‌گرایی روش یا روش‌های مرجح وجود داشته باشند؟
مثلا علت اینکه شما کنترل‌ها در MVC رو لایه نمایش به‌حساب میارین متوجه نشدم و مگر در واقع BL ما در کنترلرها اتفاق نمیوفته؟
یا مثلا ViewModel ما اگر با View در دولایه متفاوت هستند_(که هستند) جزو همان لایه نمایش به‌حساب میان؟
اگر برایتان مقدور است در مورد مرزهای شفاف تفکیک منطقی و فیزیکی (Layer & Tier) در MVC توضیح بدین،ممنون میشم.

نویسنده: حسین مرادی نیا
تاریخ: ۱۳۹۱/۰۵/۳۱ ۲۰:۴۹

طبق چیزی که من میدونم Controller محتوای لازم جهت نمایش در View رو آماده میکنه اما با لایه View متفاوت است. هدف ما از این پیاده سازی همین است (آماده سازی محتوای لازم جهت نمایش در View). با این حال Controller این محتوا را از منبع یا منابع مختلف دریافت میکند. به عبارتی اطلاعات خود را از لایه Service یا همان BL دریافت میکند و کدهای لایه Business درون Controller قرار نمیگیرند.

نویسنده: میثم
تاریخ: ۱۳۹۱/۰۵/۳۱ ۲۰:۵۵

مرسی ، جالب بود

نویسنده: شاهین کیاست
تاریخ: ۱۳۹۱/۰۶/۰۱ ۰:۳۱

Controller فقط مصرف کننده‌ی منطق نهایی است ، بدنه‌ی کنترلرها جای مناسبی برای پیاده سازی منطق تجاری نیست. BL که شما از آن یاد می‌کنید در لایه‌ی دیگری رخ می‌دهد ، یکی اسم آم را Task می‌گذارد ، یکی Service و دیگری BlaBla ..
MVC جایگزینی برای N-Tier نیست ، بلکه روشی برای سازماندهی لایه‌ی نمایش می‌باشد.
ViewModelها (ViewModel === Model Of View) اشیایی هستند که از طریق لایه‌ی سرویس تولید می‌شوند و در واقع داده ای که باید در View به کاربر نمایش داده شود را نگهداری می‌کنند.
نگرانی و مسئولیت Controller فراهم کردن داده (از طریق اجرای Business logic) برای UI می‌باشد.

نویسنده: رضا بزرگی
تاریخ: ۱۳۹۱/۰۶/۰۱ ۱:۱۴

- تصور من اینست که معماری چندلایه یک stack است از لایه‌های مختلف با وظایف متفاوت. که پایین‌ترین سطح(دیتا سورس) تا بالاترین سطح(رابط کاربری) هست. در این بین بقیه اجزا مثل BL و غیره قرار میگیره. سوالم اینه که اگر بخواهیم یه الگوی نسبتا کلی ارائه بدیم این لایه‌ها چطور روی هم چیده میشه.
- تجسم این لایه‌ها شاید از این لحاظ مهم باشه که یکی از مهمترین مفاهیم شی‌گرایی یعنی کپسوله‌سازی و Information hidding باعث تولید ایده‌ی چند لایه‌ای هست. و دانستن و اجرای درست اون خیلی کار توسعه را آسانتر میکنه.
- N-Tier یک معماری برای طراحی هست. ولی MVC یه الگوی طراحی. و این‌ها جایگزین که نه، بلکه به نوعی با هم‌پوشانی کار را اصولی‌تر میکند. همانطور که جناب نصیری در سری MVC اشاره‌ای موکد داشتند که Model در MVC در واقع همان ViewModel است .
که این ViewModel از الگوی MVVM آمده.

نویسنده: شاهین کیاست
تاریخ: ۱۳۹۱/۰۶/۰۱ ۱:۴۸

-معماری n-tier یک مدل برای توسعه ی نرم افزارهای انعطاف پذیر و با قابلیت استفاده‌ی مجدد می‌باشد. MVC یک Architectural_pattern است.
-MVC در یک معماری چند لایه وظیفه‌ی لایه‌ی نمایش (Presentation) را بر عهده دارد. همانند MVP در Windows Forms یا Web Forms. http://allthingscs.blogspot.de/2011/03/mvc-vs-3-tier-pattern.html

-ViewModel الگوی MVVM با ViewModel که ما در MVC داریم تفاوت زیادی دارند و ارتباطی با هم ندارند.

نویسنده: اصغر ترابی
تاریخ: ۱۳۹۱/۰۶/۰۱ ۴:۷

با عذر خواهی
ولی متاسفانه این ایده که منطق تجاری را در کنترلر قرار دهیم کاملا غلط است.واضح‌ترین دلیل هم این است که منطق تجاری ممکن است بین چندین استفاده کننده مشترک باشد، خیلی ساده فرض کنید قرار است نسخه وب و ویندوز یک سیستم ساخته شود.

نویسنده: رضا
تاریخ: ۱۳۹۱/۰۶/۰۱ ۹:۶

با سلام، معماری لایه ای طراحی‌های متفاوتی داره
و هر طراحی تو سناریوی خاصی ممکنه پیگیری بشه
ولی دو نکته همیشه پابرجا هستش
اول این که چون لایه UI پیچیده‌ترین لایه هستش، با یکی از **الگوهای معماری** زیر سر و سامانی پیدا می‌کنه :
Model - View - Controller
Model View - View Model

Model View Presenter
نکته دوم این هستش که سعی می‌کنن معماری لایه ای رو به صورتی پیاده سازی کنند که بر **N-Tier Development** و Tierهای فیزیکی برنامه منطبق بشه
مثلا DA و BL و Service Layer در سمت سرور، و View و View Model در سمت کلاینت در یک برنامه Desktop
امیدوارم مطلبم شفافیت رو افزایش داده باشه

نویسنده: محمد صاحب
تاریخ: ۱۳۹۱/۰۶/۰۱ ۹:۲۴

iPODD سرنام Idiomatic Presentation, Orchestration, Domain and Data هست من لینک خاصی تو نت ازش ندیدم تو همون کتاب که معرفی کردم باهاش آشنا شدم...
کتاب رو اگه پیدا نکردی بگو برات میل کنم.

نویسنده: شاهین کیاست
تاریخ: ۱۳۹۱/۰۶/۰۱ ۱۱:۵

در مطلب فعلی هم سعی شده به همین موضوع پرداخته شه که منطق تجاری را در Controller قرار ندهید.
گفته شده Controller باید محل استفاده از خروجی منطق تجاری باشد.

نویسنده: امیرحسین جلوداری
تاریخ: ۱۳۹۱/۰۶/۰۲ ۰:۶

الان برای من یه سوالی پیش اومد :
طبق حرفایه شاهین و بقیه‌ی بچه‌ها الان MVC از دید معماری سه لایه داره رو لایه‌ی نمایش مانور میده ... خوب برا چی نیومدن بگن VMVC ؟! (ViewModel-View-Controller) ...چون الان Model در واقع همون ViewModel هست! (اون چیزی که تو View استفاده میشه نه Domain Model!)

نویسنده: شاهین کیاست
تاریخ: ۱۳۹۱/۰۶/۰۲ ۰:۲۵

در مطالب قبلی هم گفته شده بود که M در ASP.NET MVC همان ViewModel هست یا Model Of View .
M در الگوی MVC (الگوی MVC به صورت عمومی در همه‌ی چارچوب ها) هم تنها وظیفه‌ی تامین داده‌ی View را دارد. در صفحه‌ی Wikipedia الگوی MVC نوشته شده :

```
A view requests from the model the information that it needs to generate an output representation.
```

تعریف Domain Entity چیز دیگری هست و برای استفاده از الگوی MVC لزوما نیازی نیست که برنامه‌ی ما دیتابیس داشته باشد.
این لینک
هم مفید است.

نویسنده: حسین مرادی نیا
تاریخ: ۱۳۹۱/۰۸/۱۹ ۲۲:۳۱

کدهای بالا رو در نظر بگیرید؛
ممکن است در حین عملیات‌های بررسی آماده بودن یک سفارش ، بررسی معتبر بودن ایمیل کاربر و ... در صورتی که نتیجه عملیات false باشد بخواهیم پیامی به کاربر نمایش داده شود(نمایش دلیل خطا به کاربر خطا(سفارش آماده نیست - ایمیل وارد نشده و ...)). در این حالت چگونه می‌توان این عملیات را به درستی در کد پیاده کرد؟

نویسنده: شاهین کیاست

تاریخ: ۱۳۹۱/۰۸/۱۹ ۲۳:۲۷

برای نمایش پیام به کاربر در ASP.NET MVC روش‌های زیادی وجود دارد ، مثلا می‌توان در ViewModel خود یک پراپرتی جهت نمایش پیغام به کاربر تعبیه کرد و در صورت نیاز به کمک یک Helper در View پیام مورد نظر را نشان داد یا از شیء TempData استفاده کرد.

نویسنده: محسن درپرستی
تاریخ: ۱۳۹۲/۱۲/۰۶ ۲۰:۳۱

اگر طبق راهنمایی‌های همین پست پیش بریم ، من می‌تونم این روش رو پیشنهاد بدم که مقدار متغیر status اگر برابر با successful نبود . به یک متد دیگه ای ارسال بشه که مسئولیت ساخت و برگرداندن پیام مناسب رو برعهده داره . اون پیام‌ها هم بهتره که به جای اینکه در قالب یک سری if یا switch محاسبه بشن در قالب یک dictionary نگهداری بشن .

```
public RedirectToRouteResult Ship(int orderId)
{
   var status = _orderShippingService.Ship(orderId);
   if (status.Successful)
   {
      return RedirectToAction("Shipped", "Order", new {orderId});
   }
   return RedirectToAction("NotShipped", "Order", new {id = orderId, desc = status});
}
```

ولی محل فراخوانی این تابع دوم کجا باید باشه ؟ نظر من اینه که نباید توی این اکشن باشه چون وظیفه‌ی این اکشن چیز دیگری است . بهتره که مقدار status به اکشن NotShipped ارسال بشه و در اونجا پیام استخراج و به view ارسال بشه . چون در اون اکشن احتمالا دلیل ship نشدن سفارش باید به کاربر نمایش داده بشه.

عنوان: **نحوه ارتقاء برنامه‌های موجود MVC3 به MVC4**
نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۶/۰۳ ۱۰:۴۸
آدرس: www.dotnettips.info
گروه‌ها: Entity framework, MVC, ASP.Net MVC, MS Ajax Minifier

در ادامه، مراحل ارتقاء پروژه‌های قدیمی MVC3 را به ساختار جدید پروژه‌های MVC4 مرور خواهیم کرد .

1) نصب پیشنیاز

الف) نصب VS 2012

و یا

ب) نصب بسته MVC4 مخصوص VS 2010 (این مورد جهت سرورهای وب نیز توصیه می‌شود)

پس از نصب باید به این نکته دقت داشت که پوشه‌های زیر حاوی اسمبلی‌های جدید MVC4 هستند و نیازی نیست الزاما این موارد را از NuGet دریافت و نصب کرد:

```
C:\Program Files\Microsoft ASP.NET\ASP.NET Web Pages\v2.0\Assemblies
C:\Program Files\Microsoft ASP.NET\ASP.NET MVC 4\Assemblies
```

پس از نصب پیشنیازها

2) نیاز است نوع پروژه ارتقاء یابد

به پوشه پروژه MVC3 خود مراجعه کرده و تمام فایل‌های csproj و web.config موجود را با یک ادیتور متنی باز کنید (از خود ویژوال استودیو استفاده نکنید، زیرا نیاز است محتوای فایل‌های پروژه نیز دستی ویرایش شوند).

در فایل‌های csproj (یا همان فایل پروژه؛ که vbproj هم می‌تواند باشد) عبارت

```
{E53F8FEA-EAE0-44A6-8774-FFD645390401}
```

را جستجو کرده و با

```
{E3E379DF-F4C6-4180-9B81-6769533ABE47}
```

جایگزین کنید. به این ترتیب نوع پروژه به MVC4 تبدیل می‌شود.

3) به روز رسانی شماره نگارش‌های قدیمی

سپس تعاریف اسمبلی‌های قدیمی نگارش سه MVC و نگارش یک Razor را یافته (در تمام فایل‌ها، چه فایل‌های پروژه و چه تنظیمات):

```
System.Web.Mvc, Version=3.0.0.0
System.Web.WebPages, Version=1.0.0.0
System.Web.Helpers, Version=1.0.0.0
System.Web.WebPages.Razor, Version=1.0.0.0
```

و این‌ها را با نگارش چهار MVC و نگارش دو Razor جایگزین کنید:

```
System.Web.Mvc, Version=4.0.0.0
System.Web.WebPages, Version=2.0.0.0
System.Web.Helpers, Version=2.0.0.0
System.Web.WebPages.Razor, Version=2.0.0.0
```

این کارها را با replace in all open documents توسط notepad plus-plus به سادگی می‌توان انجام داد.

4) به روز رسانی مسیرهای قدیمی

به علاوه اگر در پروژه‌های خود از اسمبلی‌های قدیمی به صورت مستقیم استفاده شده:

```
C:\Program Files\Microsoft ASP.NET\ASP.NET Web Pages\v1.0\Assemblies
C:\Program Files\Microsoft ASP.NET\ASP.NET MVC 3\Assemblies
```

این‌ها را یافته و به نگارش MVC4 و Razor2 تغییر دهید:

```
C:\Program Files\Microsoft ASP.NET\ASP.NET Web Pages\v2.0\Assemblies
C:\Program Files\Microsoft ASP.NET\ASP.NET MVC 4\Assemblies
```

5) به روز رسانی قسمت appSettings فایل‌های کانفیگ

در کلیه فایل‌های web.config برنامه، webpages:Version را یافته و شماره نگارش آن‌را از یک به دو تغییر دهید:

```
<appSettings>
  <add key="webpages:Version" value="2.0.0.0" />
  <add key="PreserveLoginUrl" value="true" />
</appSettings>
```

همچنین یک سطر جدید PreserveLoginUrl را نیز مطابق تنظیم فوق اضافه نمائید.

6) رسیدگی به وضعیت اسمبلی‌های شرکت‌های ثالث

ممکن است در این زمان از تعدادی کامپوننت و اسمبلی MVC3 تهیه شده توسط شرکت‌های ثالث نیز استفاده نمائید. برای اینکه این اسمبلی‌ها را وادار نمائید تا از نگارش‌های MVC4 و Razor2 استفاده کنند، نیاز است bindingRedirectهای زیر را به فایل‌های web.config برنامه اضافه کنید (در فایل کانفیگ ریشه پروژه):

```
<configuration>
  <!--... elements deleted for clarity ...-->

  <runtime>
    <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
      <dependentAssembly>
        <assemblyIdentity name="System.Web.Helpers"
             publicKeyToken="31bf3856ad364e35" />
        <bindingRedirect oldVersion="1.0.0.0" newVersion="2.0.0.0"/>
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="System.Web.Mvc"
             publicKeyToken="31bf3856ad364e35" />
        <bindingRedirect oldVersion="1.0.0.0-3.0.0.0" newVersion="4.0.0.0"/>
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="System.Web.WebPages"
             publicKeyToken="31bf3856ad364e35" />
        <bindingRedirect oldVersion="1.0.0.0" newVersion="2.0.0.0"/>
      </dependentAssembly>
    </assemblyBinding>
  </runtime>
</configuration>
```

اکنون فایل solution را در VS.NET گشوده و یکبار گزینه rebuild را انتخاب کنید تا پروژه مجددا بر اساس اسمبلی‌های جدید معرفی شده ساخته شود.

7) استفاده از NuGet برای به روز رسانی بسته‌های نصب شده

یک سری از بسته‌های تشکیل دهنده MVC3 مانند موارد ذیل نیز به روز شده‌اند که لازم است از طریق NuGet دریافت و جایگزین شوند:

Unobtrusive.Ajax.2

Unobtrusive.Validation.2

Web.Optimization.1.0.0

و ....

برای اینکار در solution explorer روی references کلیک راست کرده و گزینه Manage NuGet Packages را انتخاب کنید. در صفحه باز شده گزینه updates/all را انتخاب کرده و مواردی را که لیست می‌کند به روز نمائید (شامل جی کوئری، EF، structureMap و غیره خواهد بود).

8) اضافه کردن یک فضای نام جدید

بسته Web Optimization را از طریق NuGet دریافت کنید (برای یافتن آن bundling را جستجو کنید؛ نام کامل آن Microsoft ASP.NET Web Optimization Framework 1.0.0 است). این مورد به همراه پوشه MVC4 نیست و باید از طریق NuGet دریافت و نصب شود. (البته پروژه‌های جدید MVC4 شامل این مورد هستند)

در فایل وب کانفیگ، فضای نام System.Web.Optimization را نیز اضافه نمائید:

```
<pages>
      <namespaces>
        <add namespace="System.Web.Optimization" />
      </namespaces>
    </pages>
```

پس از ارتقاء

اولین مشکلی که مشاهده شد:

بعد از rebuild به مقدار پارامتر salt که به نحو زیر در MVC3 تعریف شده بود، ایراد خواهد گرفت:

```
[ValidateAntiForgeryToken(Salt = "data123")]
```

Salt را در MVC4 منسوخ شده معرفی کرده‌اند: (^)

علت هم این است که salt را اینبار به نحو صحیحی خودشان در پشت صحنه تولید و اعمال می‌کنند. بنابراین این یک مورد را کلا از کدهای خود حذف کنید که نیازی نیست.

مشکل بعدی:

در EF 5 جای یک سری از کلاس‌ها تغییر کرده. مثلا ویژگی‌های ForeignKey، ComplexType و ... به فضای نام System.ComponentModel.DataAnnotations.Schema منتقل شده‌اند. در همین حد تغییر جهت کامپایل مجدد کدها کفایت می‌کند.

همچنین فایل‌های پروژه موجود را باز کرده و EntityFramework, Version=4. **1**. 0.0 را جستجو کنید. نگارش جدید 4. **4**. 0.0. است که باید اصلاح شود (این موارد را بهتر است توسط یک ادیتور معمولی خارج از VS.NET ویرایش کنید).

در زمان نگارش این مطلب EF Mini Profiler با EF 5 سازگار نیست. بنابراین اگر از آن استفاده می‌کنید نیاز است غیرفعالش کنید.

اولین استفاده از امکانات جدید MVC4:

استفاده از امکانات System.Web.Optimization که ذکر گردید، می‌تواند اولین تغییر مفید محسوب شود.

برای اینکه با نحوه کار آن بهتر آشنا شوید، یک پروژه جدید MVC4 را در VS.NET (از نوع basic) آغاز کنید. به صورت خودکار یک پوشه جدید را به نام App_Start به ریشه پروژه اضافه می‌کند. داخل آن فایل مثال BundleConfig قرار دارد. این کلاس در فایل global.asax برنامه نیز ثبت شده‌است. باید دقت داشت در حالت دیباگ (compilation debug=true در وب کانفیگ) تغییر خاصی را ملاحظه نخواهید کرد.

تمام این‌ها خوب؛ اما من به نحو زیر از این امکان جدید استفاده می‌کنم:

```
using System.Collections.Generic;
using System.IO;
using System.Web;
using System.Web.Optimization;

namespace Common.WebToolkit
{
    /// <summary>
```

```
    /// A custom bundle orderer (IBundleOrderer) that will ensure bundles are
    /// included in the order you register them.
    /// </summary>
    public class AsIsBundleOrderer : IBundleOrderer
    {
        public IEnumerable<FileInfo> OrderFiles(BundleContext context, IEnumerable<FileInfo> files)
        {
            return files;
        }
    }

    public static class BundleConfig
    {
        private static void addBundle(string virtualPath, bool isCss, params string[] files)
        {
            BundleTable.EnableOptimizations = true;

            var existing = BundleTable.Bundles.GetBundleFor(virtualPath);
            if (existing != null)
                return;

            var newBundle = isCss ? new Bundle(virtualPath, new CssMinify()) : new Bundle(virtualPath,
new JsMinify());
            newBundle.Orderer = new AsIsBundleOrderer();

            foreach (var file in files)
                newBundle.Include(file);

            BundleTable.Bundles.Add(newBundle);
        }

        public static IHtmlString AddScripts(string virtualPath, params string[] files)
        {
            addBundle(virtualPath, false, files);
            return Scripts.Render(virtualPath);
        }

        public static IHtmlString AddStyles(string virtualPath, params string[] files)
        {
            addBundle(virtualPath, true, files);
            return Styles.Render(virtualPath);
        }

        public static IHtmlString AddScriptUrl(string virtualPath, params string[] files)
        {
            addBundle(virtualPath, false, files);
            return Scripts.Url(virtualPath);
        }

        public static IHtmlString AddStyleUrl(string virtualPath, params string[] files)
        {
            addBundle(virtualPath, true, files);
            return Styles.Url(virtualPath);
        }
    }
}
```

کلاس BundleConfig فوق را به مجموعه کلاس‌های کمکی خود اضافه کنید.

چند نکته مهم در این کلاس وجود دارد:

الف) توسط AsIsBundleOrderer فایل‌ها به همان ترتیبی که به سیستم اضافه می‌شوند، در حاصل نهایی ظاهر خواهند شد. حالت پیش فرض مرتب سازی، بر اساس حروف الفباء است و ... خصوصا برای اسکریپت‌هایی که ترتیب معرفی آن‌ها مهم است، مساله ساز خواهد بود.

ب)BundleTable.EnableOptimizations سبب می‌شود تا حتی در حالت debug نیز فشرده سازی را مشاهده کنید.

ج) متدهای کمکی تعریف شده این امکان را می‌دهند تا بدون نیاز به کامپایل مجدد پروژه، به سادگی در کدهای Razor بتوانید اسکریپت‌ها را اضافه کنید.

سپس نحوه جایگزینی تعاریف قبلی موجود در فایل‌های Razor با سیستم جدید، به نحو زیر است:

```
@using Common.WebToolkit

<link href="@BundleConfig.AddStyleUrl("~/Content/blueprint/print", "~/Content/blueprint/print.css")"
```

```
rel="stylesheet" type="text/css" media="print"/>

@BundleConfig.AddScripts("~/Scripts/js",
                            "~/Scripts/jquery-1.8.0.min.js",
                            "~/Scripts/jquery.unobtrusive-ajax.min.js",
                            "~/Scripts/jquery.validate.min.js")

@BundleConfig.AddStyles("~/Content/css",
                            "~/Content/Site.css",
                            "~/Content/buttons.css")
```

پارامتر اول این متدها، سبب تعریف خودکار routing می‌شود. برای مثال اولین تعریف، آدرس خودکار زیر را تولید می‌کند:

```
http://site/Content/blueprint/print?v=hash
```

بنابراین تعریف دقیق آن مهم است. خصوصا اگر فایل‌های شما در پوشه‌ها و زیرپوشه‌های متعددی قرار گرفته نمی‌توان تمام آن‌ها را در طی یک مرحله معرفی نمود. هر سطح را باید از طریق یک بار معرفی به سیستم اضافه کرد. مثلا اگر یک زیر پوشه به نام noty دارید (Content/noty)، چون در یک سطح و زیرپوشه مجزا قرار دارد، باید نحوه تعریف آن به صورت زیر باشد:

```
@BundleConfig.AddStyles("~/Content/noty/css",
                                "~/Content/noty/jquery.noty.css",
                                "~/Content/noty/noty_theme_default.css")
```

این مورد خصوصا در مسیریابی تصاویر مرتبط با اسکریپت‌ها و شیوه نامه‌ها مؤثر است؛ وگرنه این تصاویر تعریف شده در فایل‌های CSS یافت نخواهند شد (تمام مثال‌های موجود در وب با این مساله مشکل دارند و فرض آن‌ها بر این است که کلیه فایل‌های خود را در یک پوشه، بدون هیچگونه زیرپوشه‌ای تعریف کرده‌اید).
پارامترهای بعدی، محل قرارگیری اسکریپت‌ها و CSSهای برنامه هستند و همانطور که عنوان شد اینبار با خیال راحت می‌توانید ترتیب معرفی خاصی را مدنظر داشته باشید؛ زیرا توسط AsIsBundleOrderer به صورت پیش فرض لحاظ خواهد شد.

نظرات خوانندگان

نویسنده: امیرحسین جلوداری
تاریخ: ۱۳۹۱/۰۸/۲۵ ۱۳:۶

سلام ... من این کارارو کردم به امید اینکه بتونم برنامرو رو هاستی که تا mvc3 ساپورت کنه اجراش کنم! ... ولی به این خط گیر میده :

```
<compilation targetFramework="4.0" />
```

و پروژم دیگه رو vs2010 ای که mvc3 ساپورت میکنه لود نمیشه! ... مشکل چیه؟! ... اگه مشکلو پیدا نکنم رسما بیچارم! و باید یه پروژه‌ی جدید mvc3 بسازم و فایلارو توش کپی کنم! ... بعد یه سوال دیگه:
اگه نشه اونوقت نمیتونم تو برنامه‌ی mvc3 ، وب api و bundling and minification رو داشته باشم؟!
خواهشا کمک کنید! خیلی خیلی ضروریه!

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۸/۲۵ ۱۳:۴۳

- اگر به اون خط ایراد گرفته یعنی تنظیمات IIS آن روی ASP.NET 4.0 و کلا دات نت 4 نیست: (^). هاست باید این مساله را بررسی و تنظیم کند (بررسی هر سه نکته یاد شده در مقاله « نکات نصب برنامه‌های ASP.NET 4.0 بر روی IIS 6 » الزامی است).
- به علاوه MVC4 باید روی هاست و همچنین روی کامپیوتر توسعه نصب باشد.

نویسنده: امیرحسین جلوداری
تاریخ: ۱۳۹۱/۰۸/۲۵ ۱۵:۲

قبل از اینکه جوابتونو ببینم تماس گرفتم به سرویس دهنده‌ی هاست که مشکلم کامل حل شد :ایکس ... مشکل همونی هست که شما میگین ولی رو وب سایت پنل خودشون رو asp.net 2 تنظیم بود که درسش کردم.
الان با این شرایط که من میبینم و اونا هم ساپورت mvc 4 ندارند پس نیازی به روی هاست بودن mvc 4 نیست چون پروژم کاملا داره کار میکنه :دی

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۸/۲۵ ۱۶:۴۴

- اگر کار می‌کنه یعنی قسمت‌های 3 و 4 مطلب فوق در حین ارتقاء، ناقص انجام شده یا هنوز تغییری نکرده.
- ضمن اینکه حتما برای مدیریت فایل‌های پروژه‌های خودتون از سورس کنترل استفاده کنید تا نگران تغییرات و بازگشت به قبل نباشید.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۳/۲۴ ۹:۲۳

یک نکته تکمیلی

اولین کاری که باید پس از آغاز یک پروژه جدید MVC4 انجام داد :

خط فرمان پاورشل نیوگت را باز کنید و دستور Update-Package را صادر کنید. تقریبا تمام اجزای MVC4 مرتبا به روز می‌شوند:

```
PM> Update-Package
```

خصوصا اسکریپت‌های اعتبارسنجی همراه آن که با نگارش‌های جدیدتر jQuery سازگار شده‌اند.

نویسنده: سیروس

تاریخ: ۱۳۹۲/۰۵/۱۱ ۲۲:۱۴

دقیقا تفاوت BundleConfig.AddStyleUrl و BundleConfig.AddStyles چیه؟

نویسنده: مجتبی فخاری
تاریخ: ۱۳۹۲/۱۲/۱۶ ۲۳:۵۲

با سلام
زمانی که در PartialViewها نیاز به Jquery Validator است در صورتی که آنها را در فایل Layout و در قسمت Body به صورت

```
@Scripts.Render("~/bundles/jquery")
 @Scripts.Render("~/bundles/jqueryval")
```

تعریف نماییم و آنها نیز در فایل BundleConfig تعریف شده باشند، آیا نیاز است که ما دوباره آنها را بصورت

```
<script src="~/Scripts/jquery-1.8.2.min.js"></script>
<script src="~/Scripts/jquery.validate.min.js"></script>
<script src="~/Scripts/jquery.validate.unobtrusive.min.js"></script>
```

در PartialViewها تعریف نماییم؟ اگه خیر پس چرا جواب نمیده؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۲/۱۷ ۰:۱۷

نیاز هست با روش دیباگ این نوع برنامه‌ها بیشتر آشنا شوید. اطلاعات بیشتر
توسط فایرباگ یا حتی کنسول کروم بررسی کنید چه خطاهایی گرفتید، چه فایل‌هایی را گزارش می‌دهد که یافت نشد و امثال آن.
بر این اساس باید برنامه را دیباگ و رفع اشکال کرد.

نویسنده: علی فخرایی
تاریخ: ۱۳۹۲/۱۲/۲۸ ۰:۳۳

ممنون از مطلبتون. پروژه ام را طبق آموزش شما ارتقاء دادم و به درستی هم کار کرد اما زمانی که از Bundle خواستم استفاده کنم به مشکل زیر برخوردم.
آیا نوع خروجی متد زیر تغییر کرده است؟

```
public IEnumerable<FileInfo> OrderFiles(BundleContext context, IEnumerable<FileInfo> files)
{
     return files;
}
```

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۲/۲۸ ۰:۴۴

بله. FileInfo به BundleFile تبدیل شده.

نویسنده: م کریمی
تاریخ: ۱۳۹۳/۰۱/۲۰ ۱۱:۱۷

من از این کلاس شما در mvc 4 استفاده کردم اوکی بودش اما الان در 5 دارم استفاده میکنم تمام فایل ها
این شکلی میشن.
debug هم false کردم

```
<script src="/Asset/Scripts/js?v="></script>
 <script src="/Asset/Scripts/JQuery/Zebra/js?v="></script>
```

```
<script src="/Asset/Scripts/Angular/Asset/js?v=""></script>
```

مشکل از کجا میتونه باشه ؟ ممنون

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۱/۲۰ ۱۱:۵۰

مشکلی مشاهده نشد. با دات نت‌های 4.5 و 4.5.1 در یک برنامه MVC 5.x تست شد و کار می‌کند.
اگر v خالی است، ممکن است اسکریپت‌های مورد استفاده مشکل دارند. آدرس نهایی تولید شده را مستقلا در مرورگر باز کرده و خروجی آن‌را بررسی کنید. اگر مشکلی باشد، در ابتدای خروجی، خطاها را ذکر می‌کند.

نویسنده: محسن موسوی
تاریخ: ۱۳۹۳/۰۷/۱۲ ۱۰:۵

میتونه به دلیل این باشه که فایل‌های js مربوطه رو پیدا نکرده باشه.
جهت اطلاعات بیشتر
Unable to generate 'VersionQueryString' when using bundle

نویسنده: حسن اسلامی
تاریخ: ۱۳۹۳/۰۸/۱۱ ۱۳:۵۲

سلام. میخواستم بدونم دلیل خاصی داشته که شما توی مثال، هنگام استفاده از AddScripts فایل‌های .min رو ذکر کردید؟ یعنی حتماً باید خودمون فایل minify شده رو اضافه کنیم یا میشه فایل غیر minify شده رو اضافه کنیم تا خود سیستم Bundling & Minification عمل minify کردن رو انجام بده؟
ممنون

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۸/۱۱ ۱۴:۳۶

- اگر فایل min موجود باشد، این سیستم کار کمتری را انجام داده و از همان نمونه‌ی موجود استفاده می‌کند.
- برای سایر حالات سطر ذیل که در کدهای فوق موجود است:

```
var newBundle = isCss ? new Bundle(virtualPath, new CssMinify()) : new Bundle(virtualPath, new
JsMinify());
```

کار فشرده سازی را انجام می‌دهد.

عنوان: **آپلود فایل ها با استفاده از PlUpload در Asp.Net Mvc**
نویسنده: امیرحسین جلوداری
تاریخ: ۱۳۹۱/۰۶/۰۶ ۱۳:۴۳
آدرس: www.dotnettips.info
گروه‌ها: JavaScript, ASP.Net MVC, jQuery, Ajax

امروزه بازار برنامه‌های تماما ajax و بدون Postback شدن صفحه بسیار داغ میباشد که از این موارد میتوان به برنامه‌های تحت وب گوگل اشاره کرد. (gmail ، googlePlus ، Google Reader)
در این میان یکی از دغدغه‌های توسعه دهندگان وب ، آپلود فایل‌ها به صورت آنی (مثل attach files گوگل) میباشد. برای حل این مسئله ، ابزارها و پلاگین‌های متعددی وجود دارد که در اینجا به 10 تا از پلاگین‌های Jquery اشاره شده است.
به شخصه با پلاگین Uploadify کار کرده ام و از استفاده از آن راضی هستم ولی همین دیشب برای قسمتی از یک پروژه نیاز به ابزاری جهت آپلود فایل‌ها با امکانات مورد نظرم داشتم که به PlUpload برخورد کردم.

از امکاناتی که این ابزار در اختیار شما قرار میدهد :
- یک اینترفیس زیبا جهت آپلود و افزودن فایل ها
- پشتیبانی از زبان‌های مختلف و همین طور زبان فارسی
- امکان استفاده از قالب Jquery UI
- Drag&Drop برای مرورگرهایی که از Html5 پشتیبانی میکنند

حال که با امکانات این ابزار بیشتر آشنا شدید بریم سراغ استفاده از این ابزار در asp.net mvc :)
ابتدا پروژه را از اینجا دانلود کنید. سپس یک پروژه‌ی جدید mvc 3 بسازید (از نوع Internet Application و با نام دلخواه). سپس پوشه‌ی plupload را در قسمت سلوشن برنامه کپی کنید.
حال در فایل Views->Shared->_Layout.cshtml ، تگ head را جهت افزودن امکانات پلاگین این گونه تغییر دهید :

```
<title>@ViewBag.Title</title>

    <link href="@Url.Content("~/Content/Site.css")" rel="stylesheet" type="text/css" />
    <link href="../../plupload/js/jquery.plupload.queue/css/jquery.plupload.queue.css" rel="stylesheet"
/>
    <script src="@Url.Content("~/Scripts/jquery-1.7.1.min.js")" type="text/javascript"></script>
    <script type="text/javascript" src="http://bp.yahooapis.com/2.4.21/browserplus-min.js"></script>

    <script src="../../plupload/js/plupload.full.js"></script>
    <script src="../../plupload/js/jquery.plupload.queue/jquery.plupload.queue.js"></script>
    <script src="../../plupload/js/i18n/fa.js"></script>
```

نکته : فایل fa.js که جهت استفاده از زبان فارسی در اینترفیس آپلود فایل‌ها میباشد، که وجود آن در آدرس واضح میباشد.
سپس به فایل Views->Home->Index.cshtml بروید و آن را این گونه دوباره نویسی کنید :

```
@{
    ViewBag.Title = "Uploading Files using PlUpload";
}
<h2>@ViewBag.Message</h2>

@using (Html.BeginForm("Post", "home", FormMethod.Post,
    new { enctype = "multipart/form-data" }))
{
    <div id="uploader">
        <p>You browser doesn't have Flash, Silverlight, Gears, BrowserPlus or HTML5 support.</p>
    </div>
}

<script>
    $(function () {

        $("#uploader").pluploadQueue({
            // General settings
            runtimes: 'html5,gears,flash,silverlight,browserplus,html4',
            url: '@Url.Action("Upload" , "Home")',
            max_file_size: '10mb',
            chunk_size: '1mb',
```

```
            unique_names: true,

            // Resize images on clientside if we can
            resize: { width: 320, height: 240, quality: 90 },

            // Specify what files to browse for
            filters: [
                { title: "Image files", extensions: "jpg,gif,png" },
                { title: "Zip files", extensions: "zip" }
            ],

            // Flash settings
            flash_swf_url: '/plupload/js/plupload.flash.swf',

            // Silverlight settings
            silverlight_xap_url: '/plupload/js/plupload.silverlight.xap'
        });
    });
</script>
```

توضیحات و نکات :

- جهت آپلود فایل‌ها تگ "enctype = "multipart/form-data را فراموش نکنید.

- در قسمت مقداردهی به ویژگی‌های Plupload ، قسمت runtime به صورت ترتیبی کار میکند لذا اگر اولی پشتیبانی نشود سراغ دومی میرود و اگر دومی نشود سومی و ... در صفحه‌ی اول سایت PlUpload ، موارد پشتیبانی شده توسط تکنولوژی‌ها آورده شده است لذا این ترتیب را ترتیب مناسبی میبینم و اگر اولین مورد html5 باشد امکان Drag&Drop وجود خواهد داشت.

خود سایت PlUpload داکیومنت خیلی خوبی جهت توضیح موارد مختلف دارد لذا توضیح دوباره لازم نیست.

همان طور که در ویژگی url مشاهده میکنید به کنترلر Home و اکشن متود Upload اشاره شده است که طرز کار به این گونه است که هر بار که یک فایل آپلود میشود درخواستی به این آدرس و محتوای فایل در قسمت Request.Files ارسال میشود و همین طور نام فایل که unique ارسال میشود و chunk که تیکه‌های فایل است(پست میشود).

پس اکشنی با نام Upload در کنترلر HomeController بسازید :

```
[HttpPost]
        public ActionResult Upload(int? chunk, string name)
        {
            var fileUpload = Request.Files[0];
            var uploadPath = Server.MapPath("~/App_Data");
            chunk = chunk ?? 0;
            using (var fs = new FileStream(Path.Combine(uploadPath, name), chunk == 0 ? FileMode.Create
: FileMode.Append))
            {
                if (fileUpload != null)
                {
                    var buffer = new byte[fileUpload.InputStream.Length];
                    fileUpload.InputStream.Read(buffer, 0, buffer.Length);
                    fs.Write(buffer, 0, buffer.Length);
                }
            }
            return Content("chunk uploaded", "text/plain");
        }
```

توضیحات : ابتدا فایل مورد نظر از قسمت Request.Files واکشی میشود و سپس فایل را در پوشه App_Data ذخیره میکند. (یکی از چندین روش ذخیره سازی که مطالعه در این قسمت به خواننده واگذار میشود.)

حال برنامه را اجرا کنید و از این ابزار لذت ببرید:) پروژه‌ی ضمیمه شده

نکته : قسمت فارسی ساز اونو تغییر دادم چون که ترجمه‌ی فارسی خودش یه سری نقایصی داشت که گویا از کار با google translate به وجود اومده بود!

نظرات خوانندگان

نویسنده: امیرحسین جلوداری
تاریخ: ۱۳۹۱/۰۶/۰۶ ۱۵:۵۲

پروژه ضمیمه شد!

نویسنده: امیرحسین
تاریخ: ۱۳۹۱/۰۶/۰۶ ۱۶:۲۶

سلام
همه این ابزارها برای ارسال فایل به سرور از فلش استفاده می‌کنند.
و یه طورایی نمی‌شه گفت این کار با ای جکس انجام شده (ولی می‌شه گفت بدون پست بک)
موفق باشید.

نویسنده: امیرحسین جلوداری
تاریخ: ۱۳۹۱/۰۶/۰۶ ۱۷:۲۷

فلاش یکی از روش هاس ... ولی در کل حرف شما درسته و من به همین خاطر از واژه‌ی " آپلود فایل‌ها به صورت آنی " استفاده کردم

نویسنده: مهدی سعیدی فر
تاریخ: ۱۳۹۱/۰۶/۰۶ ۲۲:۹

ممنون از آموزش شما.

من خودم از این پلاگین استفاده می‌کنم.پلاگینی برای jquery هست و بدون استفاده از فلش و خیلی راحت با استفاده از ajax فایل را آپلود می‌کنه.

بازم ممنون

نویسنده: سعید یزدانی
تاریخ: ۱۳۹۱/۱۱/۲۷ ۱:۴۴

تشکر بابت مطلب کاربردیتون
فقط یک مشکل داره اونم وقتی هست که فایل‌ها upload شدن . دیگه اجازه‌ی ulpoad دوباره نمیده

نویسنده: asal Mansuri
تاریخ: ۱۳۹۲/۰۱/۰۲ ۲۱:۴۶

سلام
من یک فایل آپلودر خیلی جمع و جور میخوام که فقط یک فایل رو آپلود کنه ولی ایجکسی باشه / اگه لطف کنین راهنمایی کنین که چطوری میتونم فایل آپلود رو در MVC بتونم به صورت ajax همراه با loading داشته باشم ممنون میشم .

نویسنده: ابوالفضل رجب پور
تاریخ: ۱۳۹۲/۰۲/۱۴ ۱۳:۱۲

سلام
چطور میشه از لیست اسامی فایل‌های ارسال شده، در هنگام ثبت اطلاعات مدل در سرور استفاده کرد؟

در واقع اطلاعات فایل ها، قسمتی از اطلاعات مدل من محسوب میشه و در هنگام ثبت در دیتابیس باید این لیست رو هم ثبت کنم. ولی نمیدونم چطور در کنترلر ثبت مدل به اسامی فایلها دسترسی داشته باشم.
مثلا مدل زیر :

```
public class album
{
public string  name;
public string des;
public string[] images;
}
```

من اطلاعات مدل رو در حالت آجاکس به صورت ajax.beginform ارسال میکنم.

ممنون میشم اگر دوستان پاسخ بدن

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۲/۱۴ ۱۳:۵۳

در متد public ActionResult Upload بالا fileUpload.FileName اسم واقعی فایل هست؛ پارامتر name یک اسم موقتی.

نویسنده: ابوالفضل رجب پور
تاریخ: ۱۳۹۲/۰۲/۱۴ ۱۴:۱۶

خیلی ممنون از پاسختون
این قسمتی رو که گفتین درسته. من مرحله بعد از این رو مشکل دارم.
سوال اینه:
بعد از ارسال فایل‌ها به سرور، حالا نوبت ارسال اطلاعات مدل به سرور می‌رسه که این اسامی فایل‌ها هم قسمتی از اطلاعات مدل هست. چطور می‌شود در اکشن ثبت مدل، اسامی فایل‌ها اپلود شده را دریافت کرد و استفاده نمود.
یه راه حلی پیدا کردم که گفته بود بعد از اینکه کار آپلودر تموم شد اطلاعات فایل‌ها را به صورت فیلد مخفی در فرم ذخیره کن تا پست بک شود. نمیدونم آیا راه حل دیگه ای هست؟ این روش درست و اصولیه؟

```
uploader.bind('FileUploaded', function(up, file, info) {
  var obj = JSON.parse(info.response);
    $('form#quoteRequest').append('<input type="hidden" name="file_name" value="' +
obj.result.cleanFileName + '" />');
    //note obj.result.cleanFileName instead obj.cleanFileName
 });
```

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۲/۱۴ ۱۴:۱۶

اینکار توصیه نمیشه. چون پس از اطمینان کامل از آپلود یک فایل باید نسبت به ذخیره اطلاعات آن اقدام کنید. الزامی نداره که ارسال فرم حتما به آپلود موفقیت آمیز فایل‌ها در این حالت خاص منجر بشه (چون مثلا از فلش یا سیلورلایت مجزای از فرم شما هم ممکن است برای ارسال داده‌های فایل‌ها استفاده کنه). یعنی آپلود فایل در اینجا بحث مجزایی از فرم شما است. ابتدا فایل‌ها را آپلود کنید؛ بعد اجازه بدید تا توضیحات مرتبط را ثبت کنند.
یا اینکه می‌شود در حین آپلود توسط افزونه هم یک سری توضیحات اضافی رو ارسال کرد:

```
// JS
up.settings.multipart_params = {
                    description: $("#imageDescription").val()
                 };
// C#
string description = context.Request.Form["description"];
```

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۲/۱۴ ۱۴:۲۱

پیاده سازی دیگری از بحث PlUpload

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۲/۱۴ ۱۴:۳۰

```
// JS
up.settings.multipart_params = {
                    description: $("#imageDescription").val(),
                    // other fields ....
                 };
// C#
string description = context.Request.Form["description"];
```

باید از multipart_params تنظیمات این افزونه برای ارسال اطلاعات اضافی استفاده کنید.

نویسنده: ابوالفضل رجب پور
تاریخ: ۱۳۹۲/۰۲/۱۴ ۱۴:۴۳

با تشکر از دوستان .
یک مثال عملی؛ یه نرم افزار **ارسال ایمیل تحت وب مثل جیمیل** رو در نظر بگیرید. یه **فرم ارسال ایمیل** داره که می‌خواد اطلاعات رو بفرسته به سمت سرور. یه سری فایل هم قراره بصورت آجاکسی پیوست بشه. مراحل کار بدین صورته که ابتدا فایل‌ها آجاکسی به سرور فرستاده می‌شوند و یه جایی ذخیره می‌شوند. بعد فرم اطلاعات شامل گیرنده ، متن نامه و غیره بصورت آجاکس ارسال میشه. وقتی در سمت سرور می‌خوایم ایمیل رو بفرستیم، نیاز داریم که نام فایل‌های پیوست شده در مرحله قبل رو هم داشته باشیم.
همونطور که قبلا خدمت دوستان عرض کردم؛ راه حلی که من بذهنم می‌رسه اینه که بعد از اینکه فایل‌ها در مرحله اول آپلود شد، اسم فایل‌ها رو توی فرم در فیلد مخفی تزریق کنیم. بعد که فرم پست شد، طبیعتا دسترسی به نام فایل‌ها داریم. کد ذکر شده هم کار تزریق در هنگام تمام شدن ارسال فایل‌ها انجام میداد.
خواستم بدونم آیا این روش درست هست؟ و آیا روش تمیزتری وجود داره؟
باز هم تشکر

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۲/۱۴ ۱۴:۵۶

همان روشی که برای دریافت نام فایل‌های آپلود شده از سرور انتخاب کردید، مناسب است:

```
$("#uploader").pluploadQueue({
      // Post init events, bound after the internal events
      init: {
               FileUploaded: function(up, file, info) {
               // Called when a file has finished uploading
               }
      }
});
```

نویسنده: فرشته محسنیان
تاریخ: ۱۳۹۲/۰۶/۱۵ ۹:۳۶

ممنون از مطالب که گفتین. فقط سایتش دیگه اجازه دانلود نمیده، آیا از جایی دیگه میتونم بگیرم؟ و دوم اینکه mvc 4 رو هم ساپورت میکنه یا نه؟ مرسی

نویسنده: محسن خان

تاریخ: ۱۳۹۲/۰۶/۱۵ ۱۰:۲۰

plupload سورس باز هست . البته با مجوز GPL. یعنی کار خودتون رو باید سورس باز کنید (یا مجوزش رو بخرید). ضمنا MVC4 با MVC3 سازگار هست. فقط یک سری افزونه بیشتر داره.

نویسنده: جواد
تاریخ: ۱۳۹۳/۰۳/۲۱ ۹:۱

باسلام. من این کارهایی که گفتید کردم. روی vs مشکلی نداشتم. ولی وقتی نرم افزارمو پابلیش کردم و روی سرور گذاشتم متأسفانه هرکار می‌کنم خطا عدم پشتیبانی مرورگر میده. با فایرباگ چک کردم دیدم به فایلهای کتابخانه اش عدم دسترسی می‌ده. تمام تنظیمات iis رو هم چک کردم. نتونستم درستش کنم.
خطایی که میده اینطوریه:

"NetworkError: 404 Not Found - http://~/Scripts/plupload/js/jquery.ui.plupload/jquery.ui.plupload.js

نویسنده: ناهید
تاریخ: ۱۳۹۳/۰۵/۰۶ ۹:۴

سلام
یه سوال؟ وقتی بخوایم اطلاعات ذخیره شده (فایلی که قبلا آپلود کرده) رو نمایش بدیم چجوری فایل رو توی plUpload نمایش بدیم؟ خود سایت FilesAdded داره که کار نمیکنه.

عنوان: Highlight کردن لینک صفحه جاری در ASP.NET MVC
نویسنده: علیرضا اسم‌رام
تاریخ: ۱۳۹۱/۰۶/۱۹ ۲۲:۵۷
آدرس: www.dotnettips.info
برچسب‌ها: C#, ASP.Net MVC, Helper

راه حل‌های مختلفی جهت Highlight کردن لینک صفحه جاری وجود دارد و مهم‌ترین کاربرد آن در منوی اصلی سایت است.

در این مطلب سعی داریم با ارائه یک Helper راه حل مناسبی را برای این موضوع ارائه کنیم. مسئولیت این Helper ایجاد لینک است با در نظر گرفتن یک شرط: آیا لینک ایجاد شده به Action جاری اشاره دارد؟ اگر بله یک CSS Class با عنوان currentMenuItem به آن اضافه کن.

```
public static MvcHtmlString MenuLink(this HtmlHelper helper, string text, string action, string
controller)
{
    var routeData = helper.ViewContext.RouteData.Values;
    var currentController = routeData["controller"];
    var currentAction = routeData["action"];

    if(String.Equals(action, currentAction as string,
              StringComparison.OrdinalIgnoreCase)
        &&
       String.Equals(controller, currentController as string,
               StringComparison.OrdinalIgnoreCase))

    {
       return helper.ActionLink(
           text,action, controller, null,
           new { @class="currentMenuItem"}
           );
    }
    return helper.ActionLink(text, action, controller);
}
```

نحوه استفاده:

```
<li>@Html.MenuLink("Contact", "Contact", "Home")</li>
```

و البته کمی تغییرات در فایل CSS خود را فراموش نکنید:

```
ul#menu li a {
    background: none;
    color: #999;
    padding: 5px;
    border-radius: 15px;
    text-decoration: none;
}

ul#menu li a.currentMenuItem {
    background-color: black;
    color: white;
}
```

همچنین دوستانی که از Bootstrap و البته Navbar آن استفاده می‌کنند می‌توانند با کمی تغییرات در این Helper استفاده بهینه ای از آن داشته باشند:

```
public static MvcHtmlString MenuLinkBootstrap(this HtmlHelper helper, string text, string action,
string controller)
{

    var routeData = helper.ViewContext.RouteData.Values;
    var currentController = routeData["controller"];
    var currentAction = routeData["action"];

   if (String.Equals(action, currentAction as string, StringComparison.OrdinalIgnoreCase) &&
String.Equals(controller, currentController as string, StringComparison.OrdinalIgnoreCase))
   {
        return new MvcHtmlString("<li class=\"active\">" + helper.ActionLink(text, action, controller)
+ "</li>");
   }

    return new MvcHtmlString("<li>" + helper.ActionLink(text, action, controller) + "</li>");

}
```

نظرات خوانندگان

نویسنده: امین
تاریخ: ۱۳۹۱/۰۶/۲۰ ۹:۲۶

مطلب عالی بود مخصوصا بخش آخر که برای bootstrap رو هم گفتین.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۷/۰۶ ۱۲:۳۷

ممنون. مطلب جالبی است. یک راه حل عمومی دیگر مبتنی بر jQuery :

```
//--------------انتخاب خودکار لینک‌های بالای صفحه به ازای صفحه جاری
$(document).ready(function () {
    $("#headermenu a").each(function () {
        var $a = $(this);
        var href = $a.attr("href");
        if (href && (location.pathname.toLowerCase() == href.toLowerCase())) {
            //صفحه جاری را یافتیم
            $a.css({
                "color": "Yellow",
                "border-bottom": "1px solid"
             });
        }
    });
});
```

این روش بر اساس آدرس صفحه جاری و یافتن آن در ناحیه headermenu و سپس رنگی کردن آن کار می‌کند.

نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۱/۰۷/۲۷ ۱۶:۲۳

ممنونم
نکته خوبی گفتید ... چندوقتی بود دنبال مطلب شما می‌گشتم که الان به چشمم خورد.

نویسنده: صابر فتح الهی
تاریخ: ۱۳۹۲/۰۱/۰۷ ۱۲:۴۸

سلام مهندس (تبریک سال نو با تاخیر)

روشی که گفتین به نظر من یه ایراد کوچک داره و اونم اینه که در صورتی لینک رنگی می‌شه که حتما آدرس مورد نظر با آدرس لینک برابر باشه، در صورتی که لینک موردنظر یک صفحه فرعی باشه و ان لینک در لیست منو وجود نداشته باشه هیچ لینکی رنگی نمی‌شه، مثلا در سایت دات نت تیپس در صورتی که کاربر روی لینک جزئیات یک پروژه کلیک کنه لینک پروژه‌ها رنگی میشه در صورتی که ادرسش توی منوها موجود نیست، من وقتی کدهای سایت مذکور بررسی کردم فهمیدم به جای اون شما از تابع startsWith (تعریف شده توسط شما) استفاده کردین، در این صورت لینکهایی که با قسمتی از آدرس مذکور شروع بشه تغییرات رنگ (استایل مورد نظر) رو اون اعمال میشه.
اگر اشتباه می‌کنم لطفا تصحیح بفرمایید.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۱/۰۷ ۱۲:۵۸

بله. هدف این بوده تا حد ممکن شخص بدونه در چه قسمتی از سایت است. مثلا اگر به جزئیات یک پروژه مراجعه کرده، باید بدونه در قسمت پروژه‌ها هست.
این کد جدید و اصلاح شده است:

```
if (typeof String.prototype.startsWith != 'function') {
    // see below for better implementation!
    String.prototype.startsWith = function (str) {
        return this.indexOf(str) == 0;
    };
}

$(document).ready(function () {
    $("#headermenu a").each(function () {
        var $a = $(this);
        var href = $a.attr("href");

        var path = $.trim(location.pathname.toLowerCase());
        var menuHref = $.trim(href.toLowerCase());

        if (href && (path == menuHref || (menuHref != '/' && path.startsWith(menuHref)))) {
            //صفحه جاری را یافتیم
            $a.css({
                "color": "Yellow",
                "border-bottom": "2px solid"
            });
        }
    });
});
```

نویسنده: مهدی موسوی
تاریخ: ۱۳۹۲/۰۱/۰۸ ۱۵:۱۱

سلام.
به گمانم می‌تونید با یه تغییر کوچک، کدتون رو بهینه کنید. شما می‌تونید پس از یافتن صفحه جاری، return false کنید تا each به سراغ مابقی anchorهای موجود نره (چون لینک مورد نظر پیدا شده و CSS مربوطه هم اعمال شده، بنابراین دیگه نیازی نیست این حلقه به کارش ادامه بده؛ مگر اینکه بگیم بیش از یک لینک در صفحه وجود داره که در اون صورت باید location.pathname رو خارج each یکبار cache کنیم و ...).

البته، می‌تونستیم ابتدا anchorهای مورد نظر رو filter کنیم، بعد با اضافه کردن کلاس (فرضا) highlight همون کارو انجام بدیم:

```
$(function(){
    var loc = location.pathname.toLowerCase();
    $('#headermenu a').filter(function(){
        var href = this.getAttribute('href');
        return href && href.toLowerCase() == loc;
    }).addClass('highlight');
});
```

طبیعتا اگر فرض کنیم که کلیه URLها از سمت سرور بصورت Lowercase رندر میشن، اونوقت می‌تونیم از کد فوق نیز پرهیز کنیم:

```
$('#headermenu a[href="' + location.pathname + '"]').addClass('highlight');
```

موفق باشید.

پ.ن.: البته میتونیم از is هم استفاده کنیم که مقایسه رشته‌ها بطور خودکار case-insensitive نیز باشه.

نویسنده: Mehdi
تاریخ: ۱۳۹۲/۰۳/۱۸ ۰:۳۳

منظور از ناحیه headermenu چیست؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۳/۱۹ ۱۷:۱۴

یک مثال فرضی بود. مثلا یک div با id مساوی headermenu.

عنوان: **تهیه خروجی RSS در برنامه‌های ASP.NET MVC**
نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۶/۲۰ ۱۷:۵۱
آدرس: www.dotnettips.info
برچسب‌ها: MVC, RSS, ASP.Net MVC

در این مطلب با کتابخانه تهیه شده جهت تولید فیدهای RSS سایت جاری آشنا خواهید شد. در این کتابخانه مسایل زیر لحاظ شده است:

1) تهیه یک ActionResult جدید به نام FeedResult برای سازگاری و یکپارچگی بهتر با ASP.NET MVC
2) اعمال زبان فارسی به خروجی نهایی (این مورد حداقل در IE محترم شمرده می‌شود و فید را، راست به چپ نمایش می‌دهد)
3) اعمال جهت‌های rtl و ltr به متون فارسی یا انگلیسی به صورت خودکار؛ به نحوی که خروجی نهایی در تمام فیدخوان‌ها یکسان به نظر می‌رسد.
4) اعمال کاراکتر یونیکد RLE به صورت خودکار به عناوین فارسی (این مساله سبب می‌شود تا عنوان‌های ترکیبی متشکل از حروف و کلمات فارسی و انگلیسی، در فیدخوان‌هایی که متون را، راست به چپ نمایش نمی‌دهند، صحیح و بدون مشکل نمایش داده شود.)
5) نیازی به کتابخانه اضافی خاصی ندارد و پایه آن فضای نام System.ServiceModel.Syndication دات نت است.
6) تنظیم صحیح ContentEncoding و ContentType جهت نمایش بدون مشکل متون فارسی

سورس کامل این کتابخانه به همراه یک مثال استفاده از آن را از اینجا می‌توانید دریافت کنید:
MvcRssApplication.zip

توضیحاتی در مورد نحوه استفاده از آن

کتابخانه کمکی MvcRssHelper به صورت یک پروژه Class library جدا تهیه شده است. بنابراین تنها کافی است ارجاعی را به اسمبلی آن به پروژه خود اضافه کنید. البته این پروژه برای MVC4 کامپایل شده است؛ اما با MVC3 هم قابل کامپایل است. فقط باید ارجاع به اسمبلی System.Web.Mvc.dll را حذف و نمونه MVC3 آن‌را جایگزین کنید.
پس از اضافه کردن ارجاعی به اسمبلی آن، اکشن متد فید شما یک چنین امضایی را باید بازگشت دهد:

```
FeedResult(string feedTitle, IList<FeedItem> rssItems, string language = "fa-IR")
```

آیتم اول، نام فید است. مورد دوم، لیست عناوینی است که قرار است در فید ظاهر شوند. برای مثال، هر بار 20 آیتم آخر مطالب سایت را گزارش‌گیری کرده و به فرمت لیستی از FeedItemها باید ارائه دهید. FeedItem هم یک چنین ساختاری دارد و در اسمبلی MvcRssHelper قرار گرفته:

```
using System;

namespace MvcRssHelper
{
    public class FeedItem
    {
        public string Title { set; get; }
        public string AuthorName { set; get; }
        public string Content { set; get; }
        public string Url { set; get; }
        public DateTime LastUpdatedTime { set; get; }
    }
}
```

دو نکته در اینجا حائز اهمیت است:
الف) تاریخ استاندارد یک فید میلادی است نه شمسی. به همین جهت DateTime در اینجا ظاهر شده است.
ب) Url آدرسی است به مطلب متناظر در سایت و باید یک آدرس مطلق مثلا شروع شده با http باشد.

یک مثال از استفاده آن

فرض کنید مدل مطالب سایت ما به نحو زیر است:

```
using System;

namespace MvcRssApplication.Models
{
    public class Post
    {
        public int Id { set; get; }
        public string Title { set; get; }
        public string AuthorName { set; get; }
        public string Body { set; get; }
        public DateTime Date { set; get; }
    }
}
```

و یک منبع داده فرضی (کوئری از بانک اطلاعاتی یا استفاده از یک ORM یا ... موارد دیگر) نهایتا تعدادی رکورد را در اختیار ما خواهد گذاشت:

```
using System;
using System.Collections.Generic;
using MvcRssApplication.Models;

namespace MvcRssApplication.DataSource
{
    public static class BlogItems
    {
        public static IList<Post> GetLastBlogPostsList()
        {
            var results = new List<Post>();
            for (int i = 1; i < 21; i++)
            {
                results.Add(new Post
                {
                     AuthorName = "شخص " + i,
                     Body = "مطلب " + i,
                     Date = DateTime.Now.AddDays(-i),
                     Id = i,
                     Title = "عنوان "+i
                });
            }
            return results;
        }
    }
}
```

اکنون برای نمایش این اطلاعات به صورت یک فید، تنها کافی است به صورت زیر عمل کنیم:

```
using System.Collections.Generic;
using System.Web.Mvc;
using MvcRssApplication.DataSource;
using MvcRssApplication.Models;
using MvcRssHelper;

namespace MvcRssApplication.Controllers
{
    public class HomeController : Controller
    {
        const int Min15 = 900;

        [OutputCache(Duration = Min15)]
        public ActionResult Index()
        {
            var list = BlogItems.GetLastBlogPostsList();
            var feedItemsList = mapPostsToFeedItems(list);
            return new FeedResult("فید مطالب سایت ما", feedItemsList);
        }

        private List<FeedItem> mapPostsToFeedItems(IList<Post> list)
        {
            var feedItemsList = new List<FeedItem>();
            foreach (var item in list)
```

```
            {
                feedItemsList.Add(new FeedItem
                {
                    AuthorName = item.AuthorName,
                    Content = item.Body,
                    LastUpdatedTime = item.Date,
                    Title = item.Title,
                    //این آدرس باید مطلق باشد به نحو زیر
                    Url = this.Url.Action(actionName: "Details", controllerName: "Post", routeValues:
new { id = item.Id }, protocol: "http")
                });
            }
            return feedItemsList;
        }
    }
}
```

توضیحات

BlogItems.GetLastBlogPostsList منبع داده فرضی ما است. در ادامه باید این اطلاعات را به صورت لیستی از FeedItemها در آوریم. می‌توانید از AutoMapper استفاده کنید یا در این مثال ساده، نحوه انجام کار را در متد mapPostsToFeedItems ملاحظه می‌کنید.

نکته مهم بکارگرفته شده در متد mapPostsToFeedItems، استفاده از Url.Action برای تولید آدرس‌هایی مطلق متناظر با کنترلر نمایش مطالب سایت است.

پس از اینکه feedItemsList نهایی به صورت پویا تهیه شد، تنها کافی است return new FeedResult را به نحوی که ملاحظه می‌کنید فراخوانی کنیم تا خروجی حاصل به صورت یک فید RSS نمایش داده شود و قابل استفاده باشد. ضمنا جهت کاهش بار سرور می‌توان از OutputCache نیز به مدتی مشخص استفاده کرد.

نظرات خوانندگان

نویسنده: علیرضا اسم‌رام
تاریخ: ۱۳۹۱/۰۶/۲۰ ۱۹:۸

سلام. بسیار کاربردی و عالی. سپاس.

نویسنده: احمدعلی شفیعی
تاریخ: ۱۳۹۱/۰۶/۲۱ ۰:۳۳

یک‌سری از وبگاه‌ها عادت دارن از description به‌جای content استفاده کنند.آیا این کتابخانه این قابلیت رو داره؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۶/۲۱ ۰:۴۳

خروجی کتابخانه فوق، RSS استاندارد است و description دارد. مقادیر خواص کلاس FeedItem نهایتا به این خروجی نگاشت می‌شود.
برای نمونه این خروجی رو می‌تونید در سورس نهایی فید سایت مشاهده کنید.

نویسنده: محمد صافدل
تاریخ: ۱۳۹۱/۰۶/۲۱ ۱۰:۳۰

ممنون آقای نصیری. بسیار عالی بود و ابهامات زیادی در مورد Rss برای من برطرف شد.

نویسنده: محمد صافدل
تاریخ: ۱۳۹۱/۰۶/۲۳ ۱۵:۲۴

در صورت امکان در مورد نحوه استفاده از کتابخانه MvcRssHelper در پروژه‌های ASP.NET و تغییراتی که باید در این کتابخانه انجام شود، راهنمایی کنید. با تشکر

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۶/۲۳ ۱۶:۲۲

آنچنان تفاوتی نداره. فقط اینبار باید از فایل‌های ashx استفاده کنید. روال ProcessRequest آن معادل روال ExecuteResult موجود در فایل FeedResult.cs است.

نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۱/۰۷/۲۲ ۱۹:۹

سلام
ممنوم بابت مطلب خوبتون
ولی فایل که برای دانلود گذاشتید مشکل داره
پروژه ام وی سی توش نیست ... ممنونم می‌شم اگر اصلاح کنید

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۷/۲۲ ۱۹:۴۱

داخل فایل MvcRssApplication.zip فوق این موارد هست:
پوشه MvcRssApplication : یک پروژه مثال

پوشه MvcRssHelper : کتابخانه اصلی

نویسنده: مهدی پایروند
تاریخ: ۱۳۹۱/۰۷/۲۳ ۱۵:۲۲

سلام، من هم توی لود پروژه مشکل دارم، البته همه نسخه‌های mvc رو نصب شده داشتم ولی:

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۷/۲۳ ۱۶:۲۶

- این یک پروژه MVC4 است. همچنین فرض هم بر این است که IIS Express بر روی سیستم شما نصب است (^).

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۷/۳۰ ۰:۱۱

نسخه به روز شده این پروژه:

MvcRssApplication.zip

تغییرات:

پیشتر فقط تاریخ به روز رسانی را داشت:

```
<a10:updated>2012-10-20T16:09:13+03:30</a10:updated>
```

الان تاریخ انتشار هم به آن اضافه شده:

```
<pubDate>Sat, 30 Jan 2010 02:26:32 -0800</pubDate>
```

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۸/۰۲ ۲۳:۷

به روز رسانی سوم:

MvcRssApplication3.zip

تغییرات (بر اساس نظرات W3C Feed Validation Service):

- channel link به آن اضافه شد.
- فضای نام a10 ایی که تولید می‌شد با atom جایگزین شد.

نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۱/۰۸/۱۳ ۱۷:۳

سلام

از این روش می‌شه برای ساخت sitemap هم استفاده کرد؟
اگر نه چطور؟

عنوان: **ایجاد Helper سفارشی جهت نمایش ویدئو در ASP.NET MVC**
نویسنده: علیرضا اسم‌رام
تاریخ: ۱۳۹۱/۰۶/۲۲ ۰:۵۳
آدرس: www.dotnettips.info
برچسب‌ها: MVC, ASP.Net MVC, Helper

معرفی HTML Helpers

در صورتی که در مورد Helperها در ASP.NET MVC اطلاعات بیشتری نیاز دارید پیشنهاد می‌کنم ابتدا این مطلب + را مطالعه کنید.

ایجاد یک HTML Helper سفارشی برای نمایش ویدئوهای سایت آپارات

قبل از ایجاد هر Helper ی باید با خروجی نهایی آن آشنا بود. پس از بررسی خروجی نهایی کافیست بخش هایی از آن را Optional کنیم تا در زمان استفاده از آن، نسبت به مقادیر این دسته از پارامترها تصمیم گیری کنیم. سایت آپارات

نمونه‌ی فارسی سایت YouTube است و امکان اشتراک گذاری فایل‌های ویدئویی را برای کاربران مهیا کرده است. در این سایت در صفحه نمایش هر ویدئو، بخشی با عنوان "دریافت کد ویدئو" وجود دارد. با کمی بررسی در کدهای نهایی ایجاد شده توسط دستورات ذکر شده در این بخش، کد نهایی برای نمایش یک ویدئو به صورت زیر خواهد بود:

```
<embed height="400" width="500"
flashvars="config=http://www.aparat.com//video/video/config/videohash/BA9Md/watchtype/embed"
allowfullscreen="true"
quality="high" name="aparattv_BA9Md" id="aparattv_BA9Md"
src="http://host10.aparat.com/public/player/aparattv" type="application/x-shockwave-flash">
```

برای تولید پویای این کد Helper زیر می‌تواند مفید باشد:

```
using System.Web.Mvc;

namespace MvcApplication1
{
    public static class AparatPlayerHelper
    {
        public static MvcHtmlString AparatPlayer(this HtmlHelper helper, string mediafile, int height,
int width)
        {
            var player = @"<embed height=""{0}"" width=""{1}""
flashvars=""config=http://www.aparat.com//video/video/config/videohash/{2}/watchtype/embed""
                                allowfullscreen=""true""
                                quality=""high""
                                name=""aparattv_{2}"" id=""aparattv_{2}""""
src=""http://host10.aparat.com/public/player/aparattv""
                                type=""application/x-shockwave-flash"">";

            player = string.Format(player, height, width, mediafile);
            return new MvcHtmlString(player);
        }
    }
}
```

نحوه استفاده از این Helper:

```
@Html.AparatPlayer("BA9Md", 400, 500)
```

اگر به آدرس صفحات در سایت آپارات دقت کنید URL جاری به یک عبارت چند حرقی ختم می‌شود که از این عبارت به عنوان مقدار پارامتر mediafile و شناسه منحصر بفرد فایل ویدئو استفاده شده است.

ایجاد Helper سفارشی جهت نمایش ویدئوهای YouTube:

با توجه به توضیحات فوق یک Helper سفارشی برای نمایش ویدئوهای YouTube بصورت زیر است:

```
using System;
using System.Drawing;
using System.Web.Mvc;
```

```
namespace MvcApplication1
{

    public static class YouTubePlayerHelper
    {
        public static MvcHtmlString YouTubePlayer(this HtmlHelper helper, string playerId, string
mediaFile, YouTubePlayerOption youtubePlayerOption)
        {

            const string baseURL = "http://www.youtube.com/v/";

            // YouTube Embedded Code
            var player = @"<div id=""YouTubePlayer_{7}""width:{1}px; height:{2}px;"">
                                <object width=""{1}"" height=""{2}"">
                                <param name=""movie""
value=""{6}{0}&fs=1&border={3}&color1={4}&color2={5}""></param>
                                <param name=""allowFullScreen"" value=""true""></param>
                                <embed src=""{6}{0}&fs=1&border={3}&color1={4}&color2={5}""
                                type = ""application/x-shockwave-flash""
                                width=""{1}"" height=""{2}"" allowfullscreen=""true""></embed>
                                </object>
                            </div>";

            // Replace All The Value
            player = String.Format(player, mediaFile, youtubePlayerOption.Width,
youtubePlayerOption.Height, (youtubePlayerOption.Border ? "1" : "0"),
ConvertColorToHexa.ConvertColorToHexaString(youtubePlayerOption.PrimaryColor),
ConvertColorToHexa.ConvertColorToHexaString(youtubePlayerOption.SecondaryColor), baseURL, playerId);

            //Retrun Embedded Code
            return new MvcHtmlString(player);
        }
    }

    public class YouTubePlayerOption
    {
        int _width = 425;
        int _height = 355;
        Color _color1 = Color.Black;
        Color _color2 = Color.Aqua;

        public YouTubePlayerOption()
        {
            Border = false;
        }

        public int Width { get { return _width; } set { _width = value; } }
        public int Height { get { return _height; } set { _height = value; } }
        public Color PrimaryColor { get { return _color1; } set { _color1 = value; } }
        public Color SecondaryColor { get { return _color2; } set { _color2 = value; } }
        public bool Border { get; set; }
    }

    public class ConvertColorToHexa
    {
        private static readonly char[] HexDigits =
            {
                '0', '1', '2', '3', '4', '5', '6', '7',
                '8', '9', 'A', 'B', 'C', 'D', 'E', 'F'
            };

        public static string ConvertColorToHexaString(Color color)
        {
            var bytes = new byte[3];
            bytes[0] = color.R;
            bytes[1] = color.G;
            bytes[2] = color.B;
            var chars = new char[bytes.Length * 2];
            for (int i = 0; i < bytes.Length; i++)
            {
                int b = bytes[i];
                chars[i * 2] = HexDigits[b >> 4];
                chars[i * 2 + 1] = HexDigits[b & 0xF];
            }
            return new string(chars);
        }


    }

}
```

نحوه استفاده:

```
@Html.YouTubePlayer("Casablanca", "iLdqKUkkM6w", new YouTubePlayerOption()
                                {
                                    Border = true
                                })
```

در اینجا جهت مقداردهی پارمترهای پخش کننده ویدئو، از کلاس YouTubePlayerOption استفاده شده است.

نظرات خوانندگان

نویسنده: شاهین کیاست
تاریخ: ۱۳۹۱/۰۶/۲۲ ۱:۳۹

سلام ،
ممنون.

درباره پخش فایل ویدیویی از دیتابیس راه حل چیست ؟ من روش‌های زیادی رو امتحان کردم که یک فایل ویدیو با پسوند FLV یا MP4 را به کمک JWPlayer پخش کنم ، اما موفق نشدم ، در اینترنت هم مثالی پیدا نکردم.
پخش ویدیو از فولدر مشکلی ندارد اما خواند از دیتابیس و ارسال به صورت FileStream کار نمی‌کند.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۶/۲۲ ۹:۳۵

یک مثال « SQL2K8 FILESTREAM-WPF-HTTP Sample »
(معادل MediaElement مخصوص WPF در سیلورلایت هم هست)

عنوان: **معرفی کتاب: مرجع کامل ASP.NET MVC 4**
نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۶ ۱۶:۲۵
آدرس: www.dotnettips.info
گروه‌ها: ASP.Net MVC, کتاب

کتاب "مرجع کامل ASP.NET MVC 4"، ترجمه و تالیف اینجانب منتشر شد.

در این کتاب می‌خوانید:
دلایل معرفی و استفاده از ASP.NET MVC
اکشن ها، کنترلرها، Viewها، فیلترها، مسیریابی
آشنایی با آزمایش‌های واحد و استفاده از آنها
سفارشی سازی اجزای ASP.NET MVC
تزریق وابستگی
Model Templates و Model Binding
تعیین اعتبار داده ها
استفاده از AJAX و jQuery
امنیت
ASP.NET Web API
انتشار پروژه
و بسیاری مطالب دیگر...
مطالعه‌ی قسمتی از کتاب...
جزئیات بیشتر...

نظرات خوانندگان

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۷/۰۶ ۱۶:۳۷

تبریگ می‌گم. با آرزوی موفقیت‌های بیشتر.

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۶ ۱۷:۲

ممنونم آقای نصیری.

نویسنده: سیدمجتبی حسینی
تاریخ: ۱۳۹۱/۰۷/۰۶ ۱۷:۵۶

تشکر و سپاس صمیمانه من رو پذیرا باشید
دیشب تقریبا در اولین روز ارائه کتاب، یک دور مختصر آن را تورّق کردم.
اینکه سعی کردید بسیار کاربردی و تا حد امکان آن را حرفه‌ای بنویسید، من را به وجد آوردید!
واقعا خسته نباشید.
به امید سایر فعالیت‌های موفقیت آمیز شما

نویسنده: حسین مرادی نیا
تاریخ: ۱۳۹۱/۰۷/۰۶ ۱۸:۴

تبریک میگم مهندس.

نویسنده: مهدی پایروند
تاریخ: ۱۳۹۱/۰۷/۰۶ ۲۰:۵

تبریک میگم، که با حوصله و صبرتون این فعالیت‌های با ارزش رو تو کشور رونق میدید
به امید موفقیت بیشتر شما در همه زمینه‌های زندگیتون

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۶ ۲۰:۱۶

مرسی دوست من. لطف دارید.

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۶ ۲۰:۱۶

ممنونم.

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۶ ۲۰:۱۶

متشکرم. شما هم موفق باشید.

نویسنده: سیروان عفیفی
تاریخ: ۱۳۹۱/۰۷/۰۶ ۲۰:۱۷

تبریک میگیم مهندس.

نویسنده: مهدی سعیدی فر
تاریخ: ۱۳۹۱/۰۷/۰۶ ۲۰:۲۵

تبریک بنده را نیز پذیرا باشید

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۶ ۲۱:۱۲

سپاسگزارم مهندس :)

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۶ ۲۱:۱۲

ممنون از لطف شما

نویسنده: سالار خلیل زاده
تاریخ: ۱۳۹۱/۰۷/۰۶ ۲۱:۱۸

با تبریک و تشکر و البته خسته نباشید.

نویسنده: علیرضا اسم‌رام
تاریخ: ۱۳۹۱/۰۷/۰۶ ۲۲:۳۵

سلام،
تبریک عرض می‌کنم جناب راد.
امیدوارم این کتاب را در هفته آینده شروع کنم.

با آرزوی موفقیت روز افزون برای شما

نویسنده: علی
تاریخ: ۱۳۹۱/۰۷/۰۶ ۲۲:۳۹

سلام
عالی بود.
میشه فونتش رو درشتر بکنین.مثل فونتی که انتشارات appress بکار میبره
با تشکر

نویسنده: صابر فتح اللهی
تاریخ: ۱۳۹۱/۰۷/۰۷ ۲:۹

مثل همیشه عالیه مهندس
فقط کمی کیفیت چاپ پندار پارس پایینه
در هر صورت از دور دست مریزاد میگم و خسته نباشید

نویسنده: رضا
تاریخ: ۱۳۹۱/۰۷/۰۷ ۷:۲۳

ممنونم مهندس. بیصبرانه منتظر دریافت این کتاب هستم!

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۷ ۹:۱۱

قربان شما

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۷ ۹:۱۲

مرسی علیرضا جان.

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۷ ۹:۱۳

زمانی که تعداد صفحات زیاد میشه، ناشرها از همین فونت استفاده می‌کنن تا در فضا صرفه جویی بشه. کتاب C# انتشارات دیباگران هم با همین فونت هست.

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۷ ۹:۱۴

متشکرم. سلامت باشید.

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۷ ۹:۱۵

خواهش می‌کنم. انشاا... مفید واقع بشه.

نویسنده: مجتبی صحرائی
تاریخ: ۱۳۹۱/۰۷/۰۷ ۱۰:۵۱

سلام بهروز جان

مثل همیشه عالی

موفق باشی

نویسنده: ایمان اسلامی
تاریخ: ۱۳۹۱/۰۷/۰۷ ۱۱:۳

مهندس خسته نباشی.
امیدوارم همیشه سلامت و پر انرژی باشی.

نویسنده: میثم جوادی
تاریخ: ۱۳۹۱/۰۷/۰۷ ۱۱:۲۷

کتاب بسیار غنی و ارزشمندی است و به جامعه فارسی زبانان و همینطور نویسنده گرانقدر اش تبریک عرض میکنم.

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۷ ۱۲:۲۷

سلام مجتبی جان.
لطف داری.

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۷ ۱۲:۲۷

مرسی. شما هم همین طور.

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۷ ۱۲:۲۸

ممنونم.

نویسنده: سعدی همتی
تاریخ: ۱۳۹۱/۰۷/۰۷ ۱۲:۴۲

سلام آقای راد
تبریک میگم. حقیقتا از خواندن کتاب‌های شما لذت می‌برم.
راستش فکر میکنم اگر در مورد EF Code First هم یک کتاب بنویسید ، شاهکار خواهید کرد.
من تمام کتاب‌های شما را خوانده ام. بی صبرانه منتظر کتاب‌های آینده شما هستم.
با تشکر
موفق و سربلند باشید.

نویسنده: مریم
تاریخ: ۱۳۹۱/۰۷/۰۷ ۱۳:۱۳

سلام تبریک برای کتاب فوق العادتون
تو سایت برنامه نویس نوشتید 30 جلد از کتابو دارید و هر کی کتاب می‌خواد فیش پرداختی و با ادرس برا من به جیمیل بفرسته
1) اگه تقاضاها از 30 تا بیشتر بود چی میشه؟
2) شما به جیمیل که دسترسی دارید ایشالا :دی

نویسنده: شاهین کیاست
تاریخ: ۱۳۹۱/۰۷/۰۷ ۱۳:۲۰

قربان صمیمانه تبریک عرض می‌کنم خدمتون >:D<

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۷ ۱۶:۲۴

سلام.
1) فکر می‌کنم باید تا چند ساعت دیگه تاپیک رو آپدیت کنم و بگم که کتاب‌ها تمام شد (در این لحظه هنوز چند تا باقی مونده)! بقیه‌ی دوستان می‌تونن از طریق سایت انتشارات یا از کتاب فروشی‌های شهرشون تهیه کنن.
2) بله.

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۷ ۱۶:۲۵

صمیمانه متشکرم >:D<

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۷ ۱۶:۲۶

ممنون از لطف شما. انشاا... به زودی با کمک یکی از برجسته‌ترین افراد این سایت، کتاب EF قبلی رو آپدیت می‌کنم.

نویسنده: شهروز جعفری
تاریخ: ۱۳۹۱/۰۷/۰۷ ۱۷:۳۸

تبریک میگم مهندس.
آیا امکان خرید آنلاین وجود داره؟

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۷ ۱۷:۵۴

به سایت انتشارات برید. در اونجا توضیح داده شده.

نویسنده: Naser Tahery
تاریخ: ۱۳۹۱/۰۷/۰۷ ۲۱:۴۹

باید کتاب محشری باشه.
ممنون مهندس راد. سربلند و موفق باشی. :*

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۷ ۲۲:۱۴

مرسی. شما هم موفق باشید.

نویسنده: محسن درم بخت
تاریخ: ۱۳۹۱/۰۷/۰۸ ۸:۴۷

با عرض سلام و خسته نباشید
تبریک عرض می‌کنم به خاطر کتاب بسیار خوبتون
همیشه مشکلی که تو این زمینه داشتیم این بود که بسیار دیر آماده ارائه به مخاطب می‌شدند که خوشبختانه با وجود دوستانی مثل شما این مشکل نیز در حال برطرف شدن است
با آرزوی موفقیت‌های بیشتر برای شما

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۸ ۱۰:۳۷

ممنون از لطف شما.

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۸ ۱۰:۳۹

با تشکر از استقبال بی نظیر دوستان. کتاب هایی که دست بنده بودند به اتمام رسیدند. دوستان می‌توانند از طریق کتابفروشی‌های معتبر شهرشان یا از طریق سایت پندار پارس **نسبت به تهیه‌ی این کتاب اقدام بفرمایند.**

نویسنده: افشین
تاریخ: ۱۳۹۱/۰۷/۰۸ ۱۴:۴۹

من کتاب EF شما رو خوندم .... با اینکه بی نقص نیست ولی بی شک بهترین کتاب فارسی چاپ شده و چاپ نشده است و خیلی کاربردی توضیح داده شده ...
من حتما این کتاب شما رو هم تهیه میکنم (با وجود قیمت‌های بسیار زیاد کتابها)
از انقلاب که نتونستم گیرش بیارم ....

کاش امکان خرید آنلاین بصورت پست بانک هم بود توی سایت !
یا اگه کتاب فروشی مشخصی هست بد نیست معرفی کنید !

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۸ ۱۹:۰

برای خرید حضوری در تهران می‌تونید به آدرس زیر مراجعه کنید:
انقلاب، ابتدای کارگر جنوبی، کوی رشتچی، پلاک 14، واحد 16

نویسنده: آرمان فرقانی
تاریخ: ۱۳۹۱/۰۷/۰۹ ۱۲:۲۰

سلام بر مهندس راد عزیز و عرض تبریک برای انتشار این کتاب. همچنین تشکر فراوان از این همه زحمت و فداکاری شما. در شرایط حال حاضر انجام فعالیت‌های آموزشی و به ویژه نشر کتاب هایی از این دست واقعا نیاز به عشق و فداکاری دارد.

به تازگی مرور سریعی بر کتاب EF 4.1 داشتم تا اگر مناسب بود برای آموزش معرفی کنم. باید بگم هر چند برخی تصاویر کمی بیش از حد کوچک شده اند و چند اشکال کوچک دیگر وجود دارد ولی در مجموع کتاب خوبی بود.

موفق باشید.

نویسنده: شهروز جعفری
تاریخ: ۱۳۹۱/۰۷/۰۹ ۱۲:۳۵

کتابو تهیه کردم واقعاً تشکر میکنم.

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۰۹ ۱۲:۴۳

ممنون از لطف شما. برنامه‌ی بعدی من، آپدیت کتاب EF است.

نویسنده: علیرضا1
تاریخ: ۱۳۹۱/۰۷/۱۲ ۹:۲۰

با عرض سلام و خسته نباشید.
نمی دونم چطوری از زحمات بی دریغ شما تشکر کنم .
به امید موفقیت روز افزون شما.

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۷/۱۲ ۹:۳۶

مرسی. شما هم موفق باشید.

نویسنده: بهمن خلفی
تاریخ: ۱۳۹۱/۰۸/۳۰ ۱۴:۵۱

با عرض سلام و تبریک خدمت دوست گرامی جناب راد
با توجه به اینکه بنده این کتاب را دارم و برخی مواقع جهت بررسی مسائل و کسب راهنمایی به آن مراجعه میکنم سوالی داشتم :
این کتاب تقریبا براساس فصل بندی‌های کتاب wrox تالیف شده
آیا شما به نظر نویسنده این کتاب کدام کتاب را بیشتر توصیه مینمائید ...
آیا تمامی مطالب مفید و کاربردی MVC4 پوشش داده شده است یا باید به سراغ کتاب wrox بریم؟

در هر صورت خوشحالم که افراد با سوادی همچون شما در کشورمان ایران فعالیت میکنید . با آرزوی سعادت و کامیابی شما

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۰۸/۳۰ ۲۱:۵۹

بر اساس فصل بندی‌های کتاب Apress نوشته شده. به نظر من کتاب Apress جامع‌تر هست.
تمامی قابلیت‌های برجسته‌ی ASP.NET MVC 4 در این کتاب هست. اما هر چه بیشتر مطالعه کنید، ضرر نمی‌کنید.

نویسنده: مسعود زیانی
تاریخ: ۱۳۹۱/۰۹/۱۶ ۲۲:۴۷

آقای راد بابت کتاب بسیار مفیدتان تشکر میکنم

نویسنده: Dr.Bronx
تاریخ: ۱۳۹۱/۱۰/۲۹ ۱۷:۱۳

خیلی دنبال یک کتاب خوب بودم .
خوشحالم که قراره کتاب آقای راد رو بخونم .
مطمئنم که مثل کتاب CSS ایشون این هم کتاب فوق العاده هست .
از اینجا : http://www.pendarepars.com/book/index/137/?filter=MVC
خریداری کردم . امیدوارم که زودتر به دستم برسه ...

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۱۰/۲۹ ۱۸:۴۴

مرسی دوست من.
امیدوارم مفید واقع بشه.

نویسنده: عبداللهی راد
تاریخ: ۱۳۹۲/۰۴/۲۸ ۲۲:۵۱

باسلام
ضمن تشکر از بابت کتاب پرمحتوای asp.net mvc4 و همچنین entity framework
سوالی ازتون داشتم جناب بهروز راد
من آشنایی دارم با asp.net و sql server ولی زیاد حرفه ای نیستم البته مبتدی مبتدی هم نیستم
میخواستم ازتون بپرسم مطالعه این دوتا کتاب شما میتونه برام بصورت عملی نه تئوری مفید باشه؟البته تو انجمن آقای کرامتی فر هم توضیحاتی راجع به همین کتابهای شما رو خوندم میخوام تهیه کنم بنظرتون در سطحی هستم که بتونم از این کتابهای شما استفاده کنم؟
بسیار ممنون
موفق باشید

نویسنده: بهروز راد
تاریخ: ۱۳۹۳/۰۳/۱۵ ۱۹:۲۲

کار آپدیت این کتاب به نسخه‌ی 5.2 به اتمام رسیده. امیدوارم در تیرماه چاپ بشه :)

نویسنده: شهریار
تاریخ: ۱۳۹۳/۰۳/۱۵ ۲۱:۵۴

آقای راد در این کتاب موارد به روز شده از قبیل asp.net identity 2.0 نیز گنجانده شده ؟

نویسنده: بهروز راد
تاریخ: ۱۳۹۳/۰۳/۱۶ ۷:۱۸

قابلیت‌های تغییر پیدا کرده، بازنویسی شدند و تمامی قابلیت‌های جدید ASP.NET MVC آورده شده.

نویسنده: آتوسا فتوحی
تاریخ: ۱۳۹۳/۰۳/۱۶ ۱۲:۱۰

سلام و عرض خسته نباشین خدمت آقای راد

ویرایش جدید کتاب Entity Framework چه زمانی منتشر می‌شود ؟

نویسنده: بهروز راد
تاریخ: ۱۳۹۳/۰۳/۱۶ ۱۴:۱۳

سلام.

متاسفانه کمبود وقت و گستردگی مباحث، مانع از تکمیل کار برای کتاب Entity Framework شد.

عنوان: **ساخت DropDownList های مرتبط به کمک jQuery Ajax در MVC**
نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۱/۰۷/۲۷ ۲۱:۵۸
آدرس: www.dotnettips.info
گروه‌ها: ASP.Net MVC, jQuery, jQuery-Tips, Ajax, MVC

در پروژه‌های زیادی با مواردی مواجه می‌شویم که دو انتخاب داریم ، و یک انتخاب زیر مجموعه انتخاب دیگر است ، مثلا رابطه بین استان‌ها و شهرها ، در انتخاب اول استان انتخاب می‌شود و مقادیر DropDownList شهرها حاوی ، شهرهای آن استان می‌شود.
در پروژه‌های WebForm این کار به کمک Update Panel انجام می‌شد ، ولی در پروژه‌های MVC ما این امکان را نداریم و خودمان باید به کمک موارد دیگر این درخواست را پیاده سازی کنیم.
در پروژه ای که فعلا مشغول آن هستم ، با این مورد مواجه شدم ، و دیدم شاید بد نباشد راهکار خودم را با شما در میان بگذارم.
صورت مسئله :
کاربری نیاز به وارد کردن اطلاعات یک شیرالات بهداشتی (Tap) را دارد ، از DropDownList اول نوع آن شیر (Type) را انتخاب می‌کند ، و از DropDownList دوم ، که شامل گروه‌های آن نوع است ، گروه(Category) مورد نظر را انتخاب می‌کند. (در انتهای همین مطلب ببینید)
ساختار Tableهای دیتابیس به این صورت است:

رابطه ای از جدول Type به جدول Category.

به کمک Scaffolding یک کنترلر برای کلاس Tap (شیر آب) می‌سازیم ، به طور عادی در فایل Create.chtml مقدار گروه را به صورت DropDown نمایش می‌دهد، حال ما نیاز داریم که خودمان DropDown را برای Type ایجاد کنیم و بعد ارتباط این‌ها را بر قرار کنیم.

تابع اولی Create را این طوری ویرایش می‌کنیم :

```
public ActionResult Create()
        {
            ViewBag.Type = new SelectList(db.Types, "Id", "Title");
            ViewBag.Category = new SelectList(db.Categories, "Id", "Title");
            return View();
        }
```

همان طور که مشخص است ، علاوه بر مقادیر Category که خودش ارسال می‌کند ، ما نیز مقادیر نوع‌ها را به View مورد نظر ارسال می‌کنیم.

برای نمایش دادن هر دو DropDownList ویو مورد نظر را به این صورت ویرایش می‌کنیم :

```
<div>
            نوع
        </div>
        <div>
            @Html.DropDownList("Type", (SelectList)ViewBag.Type, "-- انتخاب ---", new { id = "rdbTyoe"
})
            @Html.ValidationMessageFor(model => model.Category)
        </div>

        <div>
            دسته بندی
        </div>
        <div>
            @Html.DropDownList("Category", (SelectList)ViewBag.Category, "-- انتخاب ---", new { id =
"rdbCategory"})
            @Html.ValidationMessageFor(model => model.Category)
        </div>
```

همان طور که مشاهده می‌کنید ، در اینجا DropDownList مربوط به Type که خودمان سمت سرور ،مقادیر آن را پر کرده بودیم نمایش می‌دهیم.

خب شاید تا اینجای کار ، ساده بود ولی میرسیم به اصل مطلب و ارتباط بین این دو DropDownList. (~~قبل از این قسمت حتما نگاهی به ساختار DropDownList یا همان تگ select بیندازید ، اطلاعات جی کوئری شما در این قسمت خیلی کمک حال شما است~~)

برای این کار ما از jQuery استفادی می‌کنیم ، **کار به این صورت است که** هنگامی که مقدار DropDownList اول تغییر کرد :
ما Id آن را به سرور ارسال می‌کنیم.
در آنجا Category هایی که دارای Type با Id مورد نظر هستند را جدا می‌کنیم
فیلدهای مورد نیاز یعنی Id و Title را می‌گیریم
و بعد به کمک Json مقادیر را بر می‌گردانیم

و مقادیر ارسالی از سرور را در optionهای DropDownList دوم (گروه‌ها) قرار می‌دهم

در ابتدا متدی در سرور می‌نویسیم ، کار این متد (همان طور که گفته شد) گرفتن Id از نوع Type استو برگرداندن Categoryهای مورد نظر. به این صورت :

```
public ActionResult SelectCategory(int id)
        {
            var categoris = db.Categories.Where(m => m.Type1.Id == id).Select(c => new { c.Id, c.Title
});
            return Json(categoris, JsonRequestBehavior.AllowGet);
        }
```

ابتدا به کمک Where مقادیر مورد نظر را پیدا می‌کنیم و بعد به کمک Select فیلد هایی مورد استفاده را جدا می‌کنیم و به کمک Json اطلاعات را بر می‌گردانیم. توجه داشته باشید که مقادیری که در categoris قرار می‌گیرد ، شبیه به مقادیر json هستند و ما می‌تانیم مستیما از این مقادیر در javascript استفاده کنیم.
در کلایت (همان طور که گفته شد) به این نیاز داریم که هنگاهی که مقدار در لیست اول تغییر کرد ، ای دی آن را به تابع بالا بفرستیم و مقادیر بازگشتی را در DropDownList دوم قرار دهیم : به این صورت :

```
$('#rdbTyoe').change(function () {
                jQuery.getJSON('@Url.Action("SelectCategory")', { id: $(this).attr('value') }, function
(data) {
```

```
                    $('#rdbCategory').empty();
                    jQuery.each(data, function (i) {
                        var option = $('<option></option>').attr("value",
data[i].Id).text(data[i].Title);
                        $("#rdbCategory").append(option);
                    });
                });
            });
```

توضیح کد :

با کمک تابع .change() از تغییر مقدار آگاه می‌شویم ، و حالا می‌توانیم مراحل کار را انجام دهیم. حالا وقت ارسال اطلاعات (Id مورد گزینه انتخاب شده در تغییر مقدار) به سرور است که ما این کار را به کمک jQuery.getJSON انجام می‌دهیم ، در تابع callback باید کار مقدار دهی انجام شود.

به ازای رکورد هایی که برگشت شده است که حالا در data قرار دارد به کمک تابع each لوپ می‌زنیم و هربار این کار را تکرار می‌کنیم.

ابتدا یک تگ option می‌سازیم

مقادیر مربوطه شامل Id که باید در attribute مورد نظر value قرار گیرد و متن آن که باید به عنوان text باشد را مقدار دهی می‌کنیم

option آماده شده را به DropDownList دومی (Category) اضافه می‌کنیم.

توجه داشته باشید که هنگامی که کار دریافت اطلاعات با موفقیت انجام شد ، مقادیر داخل DropDownList دومی ، مربوط به Category را به منظور بازیابی دوباره در

```
$('#rdbCategory').empty();
```

خالی می‌کنیم.

نتیجه کار برای من می‌شود این :

قبل از انتخاب :

بعد از انتخاب :

موفق و پیروز باشید

نظرات خوانندگان

نویسنده: فرشاد کریمی
تاریخ: ۱۳۹۱/۰۷/۲۷ ۲۳:۲۹

خیلی خوب و جامع بود

نویسنده: Hamid NCH
تاریخ: ۱۳۹۱/۰۷/۲۹ ۱۵:۲۶

مرسی مفید بود.

نویسنده: امیرسهیل
تاریخ: ۱۳۹۱/۰۹/۲۱ ۱۳:۲

سلام
ممنونم ، خیلی مطلب خوب مفیدی بود
یکی بزرگترین مشکلات ما رو حل کردید ، ما از روش شما در چند پروژه استفاده کردیم.

نویسنده: ابوالفضل
تاریخ: ۱۳۹۱/۱۰/۰۶ ۱۳:۱۸

سلام
برای من کار نکرد. به عبارتی متد getjson اجرا نمی‌شد.
تا قبل از متد jquery.getjson اجرا میشه. alert رو اجرا کرد. بعد دیگه اجرا نمیشه. تو firebug هم اساسا درخواستی json دیده نمیشه.
jquery-1.8.3

نویسنده: ابوالفضل
تاریخ: ۱۳۹۱/۱۰/۰۶ ۱۴:۱۰

مشکلم حل شد.
متد اجرا می‌شد. ولی در قسمت کنترلر حالت چرخشی پیش می‌اومد که نمی‌تونه به json تبدیل بشه. با new حل شد.

```
var Cities =from c in Cities
Select new {Id=c.Id, Name=c.Name};
return Json(Cities  , JsonRequestBehavior.AllowGet);
```

نویسنده: سعید
تاریخ: ۱۳۹۱/۱۰/۰۷ ۹:۴۴

بدون new هم می‌شه خروجی linq گرفت؟

نویسنده: ابوالفضل
تاریخ: ۱۳۹۱/۱۰/۰۷ ۹:۵۶

بله، میشه خروجی گرفت. ولی همونطور که گفتم اگر آبجکت 1 به آبجکت 2 ارتباط داشته باشه و آبجکت 2 هم متقابلا از یک مسیر دیگه به آبجکت 1 ارتباط داشته باشه، در اینجا چرخه پیش میاد و json نمی‌تونه اون رو تبدیل کنه.
بنابراین با new یک آبجکت مختصر برای خروجی تولید کردم که شامل خصوصیات دیگر که باعث چرخش می‌شوند نباشد.

البته یه جایی خوندم در نگارش‌های جدید json.net حل شده این مشکل.

موفق باشید

نویسنده: سعید
تاریخ: ۱۳۹۱/۱۰/۰۷ ۱۰:۱۴

در مثال شما که نشانی از اشیاء تو در تو نیست. حتی در سؤال شما هم نبود. خروجی return json با اشیاء تو در تو مشکلی ندارد. فقط خروجی آن‌را باید بتوانید سمت کلاینت پردازش کنید.

نویسنده: ابوالفضل
تاریخ: ۱۳۹۱/۱۰/۰۷ ۱۰:۲۵

اشیاء تو در تو با اشیاء چرخشی فرق میکنه دوست عزیز. من گفتم خروجی جیسون با اشیاء چرخشی مشکل دارد. اینجا

نویسنده: سعید
تاریخ: ۱۳۹۱/۱۰/۰۷ ۱۰:۴۳

اون شیءایی که لینک دادی تو در تو هست و ارجاع حلقوی هم دارد. این نوع سؤال پرسیدن در کل صحیح نیست. افراد از راه دور که نمی‌تونن صفحه مونتیور شما رو مشاهده کنند. نه خطایی رو نوشتی و نه کلاس‌های مدلت مشخص بود.

نویسنده: سعید یزدانی
تاریخ: ۱۳۹۱/۱۱/۰۶ ۲۳:۷

با تشکر
میشه درباره‌ی selector این خط بیشتر توضیح بدبد

```
var option = $('<option></option>').attr("value", data[i].Id).text(data[i].Title);
```

نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۱/۱۱/۰۷ ۱:۷

ببین دوست من ، ابتدا نگاهی به ساختار dropdown لیست‌ها داشته باش.
مقداری که باید توی dropdown نشان داده بشند چیزی به این صورته :

```
<option value="value">Name</option>
```

مقداری که به سرور ارسال می‌شه مقدار داخل value هست و چیزی که نمایش داده می‌شه جای Name قرار می‌گیره
حالا ما ابن جا می‌خواهیم این قسمت‌ها رو خودمان درست کنیم.
در یک حلقه each به اضای مقادیری که از سرور گرفتیم loop می زنیم option را می‌سازیم و به dropdown اضافه می‌کنیم.
در مورد این قسمت هم باید بگم

```
.attr("value", data[i].Id).text(data[i].Title)
```

attr برای ایجاد کردن یک اتریبیوت به یک المان هست و از text هم برای مقداری دهی به خود المان به کار می‌رود.
مثلا برای یک رکورد که داری ای دی 100 و عنوان AmirHossein هست این option اینطوری ساخته می‌شه:

```
<option value="100">AmirHossein</option>
```

نویسنده: سعید یزدانی
تاریخ: ۱۳۹۱/۱۱/۰۷ ۱۹:۱۹

مرسی از جواب کاملتون

نویسنده: ziya
تاریخ: ۱۳۹۲/۰۲/۰۶ ۱۷:۴۵

1- من dropdownlist رو درست کردم ولی لازمه که توی کنترلرم مقدارش رو بدونم تا بتونم توی دیتابیسم ذخیره کنم ... چه جوری بدونم مقدارش چیه؟
2- با تگ input دوتا radio button درست کردم به مقدارهای این دو هم چه جوری از کنترلر دسترسی پیدا کنم (یعنی چطور متوجه شم که کاربر کودوم رو انتخاب کرده که براساس انتخابش باید توی جدولهای جداگونه ذخیره بشه)

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۲/۰۶ ۲۱:۴۵

- در مورد radio button list مطلب داریم در سایت.
- برای استفاده مقدماتی از DropDownList یک مثال کامل در سایت ASP.NET هست.

نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۲/۰۲/۰۶ ۲۱:۵۶

سلام
1 - به کامنت بالایی مراجعه کنید و ساختار DropDownList رو مطالعه کنید. همان طور که می‌بینید مقداری که به سرور ارسال می‌شود همان مقدار داخل Value است.
این شما هستید که DropDown رو می‌سازید، همان مقداری که در value قرار می‌دهید ارسال می‌شود.
2 - برای این هم شما رو ارجاع می‌دم به ساختار Radio button .
نمونه:

```
<form>
<input type="radio" name="sex" value="male">Male<br>
<input type="radio" name="sex" value="female">Female
</form>
```

شما در این جا دو radio دارید که دارای نام یکسال هستند و مقادیر Value آن‌ها تفاوت دارد.
در اینجا هم شما می‌توانید در کنترلر خود با name ان radio مقدار آن را در یافت کنید.
به عنوان مثال شما در کنترلر یکی از مقدارهای male و female را دریافت می‌کنید.

نویسنده: nasim
تاریخ: ۱۳۹۲/۰۲/۱۴ ۱۸:۴۷

سلام
من از linq, SelectListItem استفاده کردم.در view هم ازSelectListItem. ولی بقیه دقیقا مثل شما ولی جواب نمیده.مشکل چیه؟
لطفا راهنمایی کنید

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۲/۱۴ ۱۹:۳۱

یه مطلب خوب در مورد دیباگ کدهای جی‌کوئری چند وقت پیش در سایت مطرح شده. شاید بتونه به شما در پیدا کردن مشکل کمک کنه.

نویسنده: nasim

تاریخ: ۱۳۹۲/۰۲/۱۷ ۱۵:۲۹

من این کارو امتحان کردم مشکلی نداره.میدونید کلا کد جی کوئری من رو اجرا نمیکنه.با یک alert امتحان کردم.اگه کنترلر از نو scaffolding نباشه کار نمیکنه؟من زیاد بلد نیستم

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۲/۱۷ ۱۶:۴۷

این قسمت jQuery.getJSON رو با Url درستی مقدار دهی کردید؟ این آدرس رو باید بر اساس برنامه خودتون تغییر بدید.

نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۲/۰۲/۱۷ ۱۶:۵۳

سلام
اگر امکان داره دقیق‌تر سوال رو بیان کنید و یا کدی که نوشتید اینجا بگذارید.

نویسنده: nasim
تاریخ: ۱۳۹۲/۰۲/۱۸ ۱:۳۲

سلام
من از دو فیلد از 2 دیتابیس رو با دو dropdown نمایش دادم به صورت زیر:

```
public ActionResult Create()
        {
           Models.category cat = new Models.category();
            List<SelectListItem> list = cat.GetList().Select(p => new SelectListItem { Text =
p.category1, Value = p.ID.ToString() }).ToList();

            list[0].Selected = true;
            ViewData["Category1"] = list;

            Models.subcategory subcat = new Models.subcategory();


            List<SelectListItem> list1 = subcat.GetList().Select(p => new SelectListItem { Text =
p.subcategory1, Value = p.ID.ToString() }).ToList();

            list1[0].Selected = true;
            ViewData["Category2"] = list1;

            return View();
        }
-------------------------razor view-----------------------------------------------
  <div>
         @{ List<SelectListItem> lstCategories = (List<SelectListItem>)ViewData["category1"]; }
            @Html.DropDownList("dbcat", lstCategories, new { id="dbcat" })
        </div>
        <div>
            @{ List<SelectListItem> lstsubCategories = (List<SelectListItem>)ViewData["Category2"]; }
            @Html.DropDownList("dbsubcat", lstsubCategories, new { id="dbsubcat"})
```

قسمت کنترلر برای ارتباط دو dropdownlist

```
public ActionResult SelectCategory(int id)
        {


           studentDataContext  db= new studentDataContext();
            var subcat = db.subcategories.Where(m => m.ID == id).Select(c => new
            {
                c.ID,
                c.subcategory1
            });
            return Json(subcat, JsonRequestBehavior.AllowGet);
        }
```

وقسمت کد جی کوئری:

```
<script src="~/Scripts/jquery-1.7.1.min.js"></script>
```

```
<script type="text/javascript">

    $('#dbcat').change(function () {

    jQuery.getJSON('@Url.Action("SelectCategory")', { id: $(this).attr('value') }, function (data) {
            $('#dbsubcat').empty(); jQuery.each(data, function (i) {
                var option = $('<option></option>')
             .attr("value", data[i].Id).text(data[i].Title); $("#dbcat").append(option);
            });
        });
   });

</script>
```

قسمت کد جی کوئری من اجرا نمیشه.حتی alert داخل

```
$('#dbcat').change(function ()
```

گذاشتم اجرا نشد.لطفا راهنمایی کنید
با تشکر

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۲/۱۸ ۹:۱۳

```
$(function () {
  // اینجا اسکریپت‌های جی‌کوئری باید تعریف شوند
});
```

قسمت کد جی‌کوئری رو باید در document ready قرار بدید

نویسنده: زرین
تاریخ: ۱۳۹۲/۰۲/۲۹ ۱۴:۲۸

سلام ،با تشکر از آموزشهای فوق العاده تون از درس1 تا این قسمت‌ها
من 2 ماهی هست که شروع کردم به یادگیری mvc و امروز رسیدم به این قسمت ، برنامه رو مثل شما نوشتم و یه کم مشکل دارم: قسمتی که باید dropdownList دوم که گروه است رو پر کنه کار نمی‌کنه! وارد کد jquery میشه و حتی متد SelectCategory رو هم فراخوانی میکنه اما dropdownList دومم رو خالی بر میگردونه!

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۲/۲۹ ۱۴:۴۸

```
Select(c => new { c.Id, c.Title })
```

خروجی شما باید همین‌ها باشه در متد انتخاب گروه، البته مطابق کدهای جی‌کوئری نوشته شده

نویسنده: زرین
تاریخ: ۱۳۹۲/۰۲/۲۹ ۱۵:۱۰

ممنونم از پاسختون؛ جواب داد (-:

نویسنده: بهادر
تاریخ: ۱۳۹۲/۰۸/۱۹ ۲:۳۶

با سلام
من از کد زیر استفاده کردم تا موارد تکراری حذف بشه اما dropdown بدون value هست در هر آیتمش.

```
ViewBag.city = new SelectList(_db.regions.Select(r => r.region_main).Distinct(), "region_main");
```

لطفا راهنمایی بفرمایید
با تشکر

نویسنده: داود
تاریخ: ۱۳۹۳/۰۲/۲۵ ۱۱:۲۸

با سلام؛ من شبیه همچین کاری می‌خوام انجام بدم با این تفاوت که وقتی با استفاده از dropdownlistFor یکی رو انتخاب می‌کنیم property مخصوص به اون سمت viewmodel نیز تغییر کنه.
مکان dropdownlist خارج از beginform هستش و با binding ام وی سی مقدار set نمی‌شه.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۲/۲۵ ۱۲:۴۱

```
$('#dbcat').change(function () {
    var selectedItem = $(this).val();
    $.post( "-----url----", { item: selectedItem } );
});
```

+ عنصر خارج از فرم را نمی‌شود توسط submit معمولی به سرور ارسال کرد؛ مگر اینکه از Ajax استفاده کنید و عناصر مورد نیاز را از قسمت‌های مختلف صفحه جمع‌آوری و به سرور ارسال کنید؛ مانند کدهای فوق. یا اینکه در رخداد change، یک فیلد مخفی داخل فرم را با مقدار value انتخابی مقدار دهی کنید. به این ترتیب چون این فیلد مخفی، داخل فرم هست، قابلیت ارسال به سرور را از طریق دکمه‌ی استاندارد و غیر Ajax ایی submit، خواهد یافت.

نویسنده: رضا منصوری
تاریخ: ۱۳۹۳/۰۴/۱۹ ۱۴:۱۴

یک تجربه :
این سلکتور $('this).attr('value) برای کار با value در Option نال برمیگردوند و من من اینجوری استفاده کردم

```
$('#Cities').change(function () {
                jQuery.getJSON('@Url.Action("SelectTown")', { id: $(this).val() }, function (data) {
                    $('#Towns').empty();
                    jQuery.each(data, function (i) {
                        var option = $('<option></option>').val(data[i].ID).text(data[i].Name);
                        $("#Towns").append(option);
                    });
                });
            });
```

منظورم استفاده از .val به جای .attr('value') هستش. نمیدونم شاید بخاطر نسخه جدیدتر Jquery هست.

نویسنده: مهیاس هانی
تاریخ: ۱۳۹۳/۱۰/۰۳ ۹:۱۱

با سلام؛ من راجب dropDown در پیج Edit.cshtml سوال دارم. چطور می‌تونم SelectedItem رو تنظیم کنم، منظورم اینکه وقتی صفحه ویرایش لود میشه، منوی کشویی روی هموم آیتمی باشه که کاربر قبلا وارد کرده و در داخل DataBase ذخیره شده.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۱۰/۰۳ ۹:۲۰

از مقدار دهی چهارمین پارامتر SelectList برای تعیین selectedValue استفاده کنید:

```
public SelectList(IEnumerable items, string dataValueField, string dataTextField, Object selectedValue)
```

```
// in controller
ViewBag.Countries = new SelectList(countries.GetCountries(), "id", "countryName", "82" /* selected id
*/);

// in view
@Html.DropDownListFor(model => model.CountryId, (IEnumerable<SelectListItem>)ViewBag.Countries, new {
@class = "form-control" })
```

عنوان: **صفحه کلید مجازی برای ورود اطلاعات**
نویسنده: صابر فتح الهی
تاریخ: ۱۳۹۱/۰۹/۰۶ ۹:۱۵
آدرس: www.dotnettips.info
گروه‌ها: ASP.Net, JavaScript, ASP.Net MVC, ASP.Net 4.5, Security

همانگونه که اطلاع دارید یکی از روش‌های سرقت اطلاعات استفاده از نرم افزارهای جاسوس صفحه کلید (Key Logger) است، البته ثبت کلیدهای فشرده شده می‌تواند توسط سخت افزارهایی که سر راه سوکت صفحه کلید و کیس قرار می‌گیره، انجام بشه. در صورتی که چنین سخت افزاری (مخصوصا در کافی نت‌ها) روی کامپیوتر کاربر نصب باشه، یا توسط ویروس و بدافزارها اینگونه نرم‌افزارهایی روی سیستم کاربر قرار بگیره هر کلیدی که توسط کاربر روی صفحه کلید فشرده میشه توسط اینها ثبت شده و در موقع مناسب برای فرد سازنده به طریقی (ایمیل یا ارتباط از طریق برنامه‌های مبتنی بر سوکت) حتی بسیاری از این برنامه‌ها پا را فراتر گذاشته و عنوان پنجره ای که کلیدها در آن فشرده شده نیز ثبت می‌شود (توسط توابع API ویندوز- البته اگر دوستان مایل باشن و از نظر مدیریت سایت ایرادی نداشته باشه، نحوه طراحی این نوع برنامه‌های جاسوس سخت افزار، صفحه کلید، یا ماوس آموزش می‌دم توی همین سایت)، حال برای امنیت برنامه‌های تحت وب یا ویندوز چگونه می‌توان در زمان ورود اطلاعات حساس مانند کلمه عبور یا شماره کارت اعتباری این امنیت را برای کاربر ایجاد کرد که داده‌هایش توسط این سخت افزارها یا بدافزارها جایی ثبت نشود؟

بله، حدس شما درست است استفاده از صفحه کلیدهای مجازی میشه گفت یکی از بهترین راه‌های ممکن هست، چون در این روش‌ها کلید به صورت سخت افزاری فشرده نمی‌شود (کلید فشرده شده به صف پیام‌های ویندوز نمی‌رود) در نتیجه نرم افزارها یا سخت افزارهای جاسوس نمی‌توانند این اطلاعات را ثبت کنند. و کاربر با خیال راحت می‌تواند داده‌های خود را وارد نمایند (تاکید می‌کنم این روش فقط جلو این نرم افزارها یا سخت افزارها را می‌گیرد و تضمینی برای اینکه در زمان ارسال داده‌های شما لو نرود ندارد).

خوب حال چه باید کرد؟

یک راه می‌تواند پیاده‌سازی صفحه کلید مجازی با کدهای طرف کلاینت مانند جاوا اسکریپت و وی‌بی اسکریپت است، اما گروهی پلاگینی را توسعه داده‌اند که با چند خط کدنویسی ساده به راحتی می‌توانید یک صفحه کلید مجازی چندزبانه (با هر زبانی که دلتون می‌خواد) داشته باشید و از اون در برنامه‌های خودتون استفاده کنید.

نحوه‌ی نصب:

ایتدا فایل‌های مورد نیاز را از سایت سازنده که شامل فایل جاوا اسکریپت ، فایل استایل و یک تصویر (آخرین نسخه) یا از این آدرس به صورت کامل (در حال حاضر نسخه 1.49) دریافت کرده، پس از دریافت فایل‌ها آنها را در هاست خود بارگزاری (آپلود) نمائید. سپس کدهای زیر را در صفحه‌ای که می‌خواهید صفحه کلید نمایش یابد در بین تگ <head> ... و </head> قرار دهید.

```
<script type="text/javascript" src="keyboard.js" charset="UTF-8"></script>
<link rel="stylesheet" type="text/css" href="keyboard.css">
```

حالا فقط کافی است به inputboxها و یا هر ورودی دیگر خود class=keyboardInput بدهید.

مثال:

```
<input type="text" value="" class="keyboardInput">
```

در نهایت کد صفحه شما باید اینگونه باشد:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <title></title>
    <script type="text/javascript" src="keyboard.js" charset="UTF-8"></script>
    <link rel="stylesheet" type="text/css" href="keyboard.css"/>
</head>
<body>
    <input type="text" value="" class="keyboardInput"/>
</body>
```

```
</html>
```

با این کار پس از اجرای صفحه مورد نظر خروجی شما مانند تصویر زیر خواهد بود، جهت محدود کردن کلیدها و عملیات دلخواه و سفارشی سازی با پارامترهای دلخواه می‌تواند از دموهای موجود در سایت سازنده بهره بگیرید.

موفق وموید باشید.

نظرات خوانندگان

نویسنده: مجتهدی
تاریخ: ۱۳۹۱/۰۹/۰۶ ۱۸:۰

سلام
تا جایی که یادم میاد، برنامه هایی هم بودن که نقاطی که موس روشون کلیک کرده بود رو ذخیره می‌کردن (چطوریش رو نمی‌دونم!)
اگه همچین برنامه هایی وجود داشته باشن حتی اگه مثل صفحات پرداخت آنلاین بانک‌ها که جای ۱۰ کاراکتر عدد، تغییر می‌کنه، با آنالیز شماره کارت‌ها و مسائلی از قبیل اینکه هر بانک با شماره خاصی شروع می‌شه و... ، فکر کنم بدست آوردن اطلاعات حساس کار سختی نباشن...
درست می‌گم؟

نویسنده: صابر فتح الهی
تاریخ: ۱۳۹۱/۰۹/۰۷ ۱۱:۵۸

دوست من کار نشد نداره، امنیت نسبی هست
بله میشه این کار انجام داد به راحتی امکان پذیر هست، اما ذخیره نقاط چه کاربردی می‌تونی داشته باشه؟

نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۱/۰۹/۰۷ ۱۶:۴۶

مطلب مفیدی بود.
ممنونم

نویسنده: nojirom
تاریخ: ۱۳۹۱/۱۰/۱۹ ۱۵:۵۸

با تشکر از آقای فتح اللهی عزیز بابت این مطلب فقط زبان پیش فرض کیبورد راچگونه فارسی تنظیم کنیم

نویسنده: صابر فتح الهی
تاریخ: ۱۳۹۱/۱۰/۲۰ ۷:۳۷

سلام
دوست من در فایل اسکریپت مزبوطه اوایل فایل این خط پیدا کرده

```
this.VKI_kts = this.VKI_kt = "US International";  // Default keyboard layout
```

و آن را به هر زبانی که دلخواه شماست تغییر دهید مثلا برای فارسی اینگونه است:

```
this.VKI_kts = this.VKI_kt = "\u0641\u0627\u0631\u0633\u06cc";  // Default keyboard layout
```

البته من توی همون فایل "fa" جستجو کردم و عنوان اون در این قسمت قرار دادم
موفق وموید باشید

نویسنده: حیدری
تاریخ: ۱۳۹۱/۱۲/۱۸ ۲۰:۵۶

سلام

مرسی ، دستتون درد نکنه
ایشالا همیشه موفق باشید

نویسنده: مرتضی محمودی
تاریخ: ۱۳۹۲/۰۸/۲۴ ۲۰:۱۵

درود
منو راهنمایی کنین که چطور از این کلاس برای RadTextBox یا دیگر Inputهای تلریک استفاده کنم
باتشکر....

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۸/۲۴ ۲۰:۵۷

چه فرقی می‌کنه؟ به خاصیت یا کلاس CSS اش مقدار keyboardInput رو نسبت بده. تمام این کنترل‌های مصنوعی، نهایتا به inputهای استاندارد HTML تبدیل و رندر می‌شن. وگرنه مرورگر نمی‌تونه اون‌ها رو تشخیص بده یا پردازش کنه.

نویسنده: مرتضی محمودی
تاریخ: ۱۳۹۲/۰۸/۲۴ ۲۲:۲

آخه سمت سرور که میره دیگه کلاس میپره. من تکست هام داخل AjaxPanel هست.

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۸/۲۴ ۲۳:۱۲

دوباره باید بایندش کنی.

نویسنده: مهیار نظری پور
تاریخ: ۱۳۹۳/۰۴/۲۵ ۱۹:۱۳

بی نهایت درود؛ بنده، به یک کیبورد، به شکل کِی پَد (تنها اعداد، بصورت ماشین حسابی) نیازمندم.
برای درج در فیلد "شماره تماس" فُرم رزروشِن سایتم.
در نت خیلی گشتم، امّا چیزی نیافتم.
دقیقاً چیزی مثل کِی پَد درگاه آنلاین بانک، که برای درج شماره حساب و این‌ها بکار میره روی سایت.

نویسنده: صابر فتح الهی
تاریخ: ۱۳۹۳/۰۴/۲۷ ۰:۳۹

سلام
خوب میتوانید از همان کدهای درگاه بانک الگو برداری کنید
تلاشتون بکنید اگر نتوانستید من کدش براتون بگیرم
موفق و موید باشید

عنوان: **نمایش رکوردها به ترتیب اولویت به کمک jQuery UI sortable در ASP.NET MVC**
نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۱/۰۹/۰۸ ۲۰:۴۷
آدرس: www.dotnettips.info
برچسب‌ها: ASP.Net MVC, jQuery, jQuery Ajax, UI, MVC

همان طور که می‌دانید کاربرد پذیری در خیلی از پروژه‌ها حرف اول رو می‌زند و کاربر دوست دارد کارهایی که انجام می‌دهد خیلی راحت و با استفاده از موس باشد.یکی از کار هایی که در اکثر پروژه‌ها نیاز است ، چیدمان ترتیب رکوردها است. ما می‌خواهیم در این پست ترتیبی اتخاذ کنیم که کاربر بتواند رکوردها را به هر ترتیبی که دوست دارد نمایش دهد.

از توضیحاتی که قبلا دادم مشخص است که این کار احتمالا در ASP.NET WebForm کار سختی نیست ولی این کار باید در MVC از ابتدا طراحی شود.

طرح سوال : یک سری رکورد از یک Table داریم که می‌خواهیم به ترتیب وارد شدن رکوردها نباشد و ترتیبی که ما می‌خواهیم نمایش داده شود.

پاسخ کوتاه : خب باید ابتدا یک فیلد (برای اولویت بندی) به Table اضافه کنیم بعد اون فیلد رو بنا به ترتیبی که دوست داریم رکوردها نمایش داده شود پر کنیم (Sort می کنیم) و در آخر هم هنگام نمایش در View رکوردها را بر اساس این فیلد نمایش می‌دهیم.

(این پست هم در ادامه پست قبلی در همان پروژه است و از همان Table ها استفاده شده است)

اضافه کردن فیلد :

ابتدا یک فیلد به Table مورد نظر اضافه می‌کنیم. من اسم این فیلد رو Priority گذاشتم. Table من چنین وضعیتی دارد.

AMIRHOSSEIN-PC.Khazar - dbo.Types

Column Name	Data Type	Allow Nulls
Id	int	☐
Title	nvarchar(200)	☑
Description	text	☑
ImagePath	varchar(500)	☑
Priority	int	☑
IsActive	bit	☐
CreatedOn	datetime	☐
CreatedBy	uniqueidentifier	☐
ModifiedOn	datetime	☑
ModifiedBy	uniqueidentifier	☑
IsDeleted	bit	☐
		☐

افزودن فایل‌های jQuery UI :

در این مرحله شما نیاز دارید فایل‌های مورد نیاز برای Sort کردن رکوردها را اضافه کنید. شما می‌توانید فقط فایل‌های مربوط به Sortable را به صفحه خودتان اضافه کنید و یا مثل من فایل هایی که حاوی تمام قسمت‌های jQuery UI هست را اضافه کنید.

من برای این کار از Section استفاده کردم ، ابتدا در Head فایل Layout دو Section تعریف کردم برای CSS و JavaScript . و فایل‌های مربوط به Sort کردن را در صفحه ای که باید عمل Sort انجام بشود در این Section ها قرار دادم.

فایل Layout

```
<head>
    <meta charset="utf-8" />
    @RenderSection("meta", false)
    <title>@ViewBag.Title</title>
    <link href="@Url.Content("~/Content/~Site.css")" rel="stylesheet" type="text/css" />
    <link href="@Url.Content("~/Content/redactor/css/redactor.css")" rel="stylesheet" type="text/css"
/>
    <link href="@Url.Content("~/Content/css/bootstrap-rtl.min.css")" rel="stylesheet" type="text/css"
/>
    @RenderSection("css", false)
    <script src="@Url.Content("~/Scripts/jquery-1.8.2.min.js")" type="text/javascript"></script>
    <script src="@Url.Content("~/Scripts/modernizr-1.7.min.js")" type="text/javascript"></script>
    <script src="@Url.Content("~/Content/js/bootstrap-rtl.js")" type="text/javascript"></script>
    <script src="@Url.Content("~/Content/redactor/redactor.min.js")" type="text/javascript"></script>
    <script src="@Url.Content("~/Scripts/jquery.unobtrusive-ajax.min.js")"
type="text/javascript"></script>
    @RenderSection("js", false)

</head>
```

فایل Index.chtml در پوشه کنترلر Type

```
@model IEnumerable<KhazarCo.Models.Type>
@{
    ViewBag.Title = "Index";
    Layout = "~/Areas/Administrator/Views/Shared/_Layout.cshtml";
}
@section css
{<link href="@Url.Content("~/Content/themes/base/jquery-ui.css")" rel="stylesheet" type="text/css" />
}
@section js
{
    <script src="@Url.Content("~/Scripts/jquery-ui-1.9.0.min.js")" type="text/javascript"></script>
}
```

در آخر فایل Index.chtml به اینصورت شده است:

```
<h2>
    نوع ها</h2>
<p>
    @Html.ActionLink("ایجاد یک مورد جدید", "Create", null, new { @class = "btn btn-info" })
</p>
<table>
    <thead>
        <tr>
            <th>
                عنوان
            </th>
            <th>
                توضیحات
            </th>
            <th>
                فعال
            </th>
            <th>
            </th>
        </tr>
    </thead>
    <tbody>
        @foreach (var item in Model.OrderBy(m => m.Priority))
        {
            <tr id="@item.Id">
                <td>
                    @Html.DisplayFor(modelItem => item.Title)
                </td>
                <td>
                    @(new HtmlString(item.Description))
                </td>
                <td>
                    @Html.DisplayFor(modelItem => item.IsActive)
                </td>
                <td>
                    @Html.ActionLink("ویرایش", "Edit", new { id = item.Id }, new { @class = "btnEdit
label label-warning" })
                    |
                    @Html.ActionLink("مشاهده", "Details", new { id = item.Id }, new { @class =
"btnDetails label label-info" })
                    |
                    @Html.ActionLink("حذف", "Delete", new { id = item.Id }, new { @class = "btnDelete
label label-important" })
                </td>
            </tr>
        }
    </tbody>
</table>
```

** توجه داشته باشید که من به هر tr یک id اختصاص داده ام که این مقدار id همان مقدار فیلد Id همان رکورد هست ، ما برای مرتب کردن به این Id نیاز داریم (خط 25).

افزودن کدهای کلاینت :

حالا باید کدی بنویسم که دو کار را برای ما انجام دهد : اول حالت Sort پذیری را به سطرهای Table بدهد و دوم اینکه هنگامی که ترتیب سطرهای تغییر کرد ما را با خبر کند:

```
<script type="text/javascript">
    $(function () {
        $("table tbody").sortable({
            helper: fixHelper,
            update: function (event, ui) {
                jQuery.ajax('@Url.Action("Sort", "Type", new { area = "Administrator" })', {
                    data: { s: $(this).sortable('toArray').toString() }
                });
            }
        }).disableSelection();
    });
    var fixHelper = function (e, ui) {
        ui.children().each(function () {
            $(this).width($(this).width());
        });
        return ui;
    };
</script>
```

توضیح کد :

در این کد ما حالت ترتیب پذیری را به Table می دهیم و هنگامی که عمل Update در Table انجام شد تابع مربوطه اجرا می‌شود. ما در این تابع، ترتیب جدید سطرها را می‌گیریم (** به کمک مقدار Id که به هر سطر دادیم ، این مقدار Id برابر بود با Id خود رکورد در Database) و به کمک jQuery.ajax به تابع Sort از کنترلر Type در منطقه (Administrator) area ارسال می‌کنیم و در آنجا ادامه کار را انجام میدهیم.

تابع fixHelper هم به ما کمک می‌کند که هنگامی که سطرها از جای خود جدا می‌شوند ، دارای عرض یکسانی باشند و عرض آن‌ها تغییری نکند.

افزودن کد Server :

حالا باید تابع Sort که مقادیر را به آن ارسال کردیم بنویسم. من این تابع را بر اساس مقداری که از کلاینت ارسال می‌شود اینگونه طراحی کردم .

```
public EmptyResult Sort(string s)
        {
            if (s != null)
            {
                var ids = new List<int>();
                foreach (var item in s.Split(','))
```

```
                {
                    ids.Add(int.Parse(item));
                }
                int intpriority = 0;

                foreach (var item in ids)
                {
                    intpriority++;
                    db.Types.Single(m => m.Id == item).Priority = intpriority;
                }
                db.SaveChanges();
            }
            return new EmptyResult();
        }
```

در ابتدا مقادیر Id که از کلاینت به صورت String ارسال شده است را می‌گیریم و بعد به همان ترتیب ارسال در لیستی از int قرار می‌دهیم ids.

سپس به اضای هر رکورد Type مقدار اولویت را به فیلدی که برای همین مورد اضافه کردیم Priority اختصاص می‌دهیم . و در آخر هم تغییرات را ذخیره می‌کنیم. (خود کد کاملا واضح است و نیاری به توضیح بیشتر نیست)

حالا باید هنگامی که لیست Type ها نمایش داده می‌شود به ترتیب (OrderBy) فیلد Priority نمایش داده شود پس تابع Index را اینطور تغییر می‌دهیم.

```
public ViewResult Index()
        {
            return View(db.Types.Where(m => m.IsDeleted == false).OrderBy(m => m.Priority));
        }
```

این هم خروجی کار من:

نوع ها

ایجاد یک مورد جدید

عنوان	توضیحات	فعال	
مدل اهرمی	این مدل از طرح هاي جديد و نوين بهره مي برد.	☑	ویرایش | مشاهده | حذف
مدل کلاسیک	طرح هاي قديمي شيرآلات	☑	ویرایش | مشاهده | حذف
سینک شاوری	توضيحاتي در مورد سينک هاي شاوري	☑	ویرایش | مشاهده | حذف
دسته ها و پوسته ها	توضيحاتي در مورد دسته ها و پوسته ها	☑	ویرایش | مشاهده | حذف
علم های دوش	توضيحاتي در مورد علم هاي دوش	☑	ویرایش | مشاهده | حذف
سایر محصولات		☑	ویرایش | مشاهده | حذف

این عکس مربوط به است به قسمت مدیریت پروژه شیرآلات مرجان خزر .

نظرات خوانندگان

نویسنده: سعید
تاریخ: ۱۳۹۱/۰۹/۰۸ ۲۱:۱۲

« این کار احتمالا در ASP.NET WebForm کار سختی نیست »

اتفاقا کار ساده‌ای نیست و همین مراحل باید طی شود. ضمن اینکه کار با jquery ajax در آنجا به این یک دستی نیست. نیاز است در code behind فرم، متد وب سرویس مانندی به صورت استاتیک تعریف شود (که خودش سبب می‌شود تا دسترسی به اعتبار سنجی توکار مبتنی بر فرم‌ها محدود شود) یا اینکه از یک هندلر مجزا بجای یک اکشن متد کمک گرفته شود ... خلاصه خیلی داستان دارد.

نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۱/۰۹/۰۸ ۲۱:۱۶

من خودم این کار رو در webform انجام ندادم و خیلی هم با سختی هاش آشنایی ندارم.
ولی فکر کردم شاید با update panel بشه این کار رو راحت انجام داد.
خب نظرتون راجب به راه حل من چیه؟

راه حل خوبی ارائه دادم؟ متن قابل فهم بود؟

نویسنده: سعید
تاریخ: ۱۳۹۱/۰۹/۰۸ ۲۱:۲۰

بسیار عالی.

فقط در مورد update panel ... تنها کاری که انجام می‌دهد کپسوله کردن ارسال مقادیر به سرور است به صورت ajax سازگار با کنترل‌های دارای view state . بیشتر از این کاری انجام نمی‌دهد. مابقی آن اگر یک سری کنترل در toolkit آن موجود باشد برای این کارها یا خیر. این toolkit هم محدود است و آنچنان به روز نمی‌شود.

نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۱/۰۹/۰۸ ۲۱:۲۳

ممنونم از پاسختون
از اینکه سوییچ کردم روی MVC خوشحالم :)
البته یکی از دلایلی هم که این کار رو کردم همین راحتی استفاده Ajax بود.

نویسنده: امیر آشنا
تاریخ: ۱۳۹۱/۰۹/۰۸ ۲۲:۴

سلام
مطلب مفیدی بود ولی می‌تونستید بیشتر توضیح بدید.
برای ما افراد مبتدی درک بعضی قسمت‌ها کمی مشکل هست.
ولی باز هم ممنونم

نویسنده: امیرحسین مرجانی

تاریخ: ۱۳۹۱/۰۹/۰۸ ۲۲:۷

سلام
بفرمایید کدوم قسمت‌ها تا بیشتر توضیح بدم.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۹/۰۸ ۲۲:۲۱

پیشنیازهای مطلب جاری:
- در مورد RenderSection : (^)
- مقدمه‌ای بر jQuery Ajax در MVC : (^)
- db.SaveChanges : کل مباحث Entity framework سایت (^)
- EmptyResult و کلا خروجی‌های اکشن متدها: (^)
- علت استفاده از @Sort Url.Action در حین آدرس دهی: (^)

نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۱/۰۹/۰۸ ۲۲:۲۳

واقعا تشکر می‌کنم بابت این لیستی که ارائه کردید.
بحث تکمیل شد
به نظر شما باید بیشتر به جزییات اهمیت داده می‌شد؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۹/۰۸ ۲۲:۴۱

ضرورتی نداره. چون واقعا به اندازه لیستی که عنوان شد نیاز به پیشنیاز درک این مطالب هست و فرصت تکرار آن‌ها نیست. این مطالب جدید، یک سری مطالب تکمیلی هستند نه مطالب پایه و از صفر.

نویسنده: محسن موسوی
تاریخ: ۱۳۹۱/۰۹/۰۹ ۰:۲۴

با استفاده از Web API Controller تقریبا به همین راحتی امکانپذیره. تغییر زیادی هم نیاز نداره.

نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۱/۰۹/۰۹ ۰:۳۷

ممنونم آقای موسوی
قابل توجه آقا سعید (احتمالا این مطلب برای شما مفیده)

نویسنده: سعید
تاریخ: ۱۳۹۱/۰۹/۰۹ ۰:۵۲

- نیاز به vs2012 داره.
- نمیشه یک فرم رو strongly typed تعریف کرد مثل مثال بالا:

```
@model IEnumerable<KhazarCo.Models.Type>
```

نویسنده: محسن موسوی
تاریخ: ۱۳۹۱/۰۹/۰۹ ۱۰:۲۱

بحث در مورد UI نبود.بحث در مورد استفاده کردن بود.
در ضمن در Web Formها نیز از امکاناتی شبیه به MVC می‌توان بهره برد.(^)

نویسنده: سعید
تاریخ: ۱۳۹۱/۰۹/۰۹ ۱۰:۴۶

فقط mvc هست که امکان استفاده از چند view engine را با هم دارد.

صفحه‌ای که لینک دادید مربوط است به web pages asp.net و نه web forms. این web pages برای کار با webmatrix طراحی شده.

نویسنده: محسن موسوی
تاریخ: ۱۳۹۱/۰۹/۰۹ ۱۱:۴

- استفاده از این قابلیت در Web Page ها توسط VS
- استفاده از این قابلیت در Web Form ها توسط VS
هر دو مورد توسط VS استفاده شده است.

نویسنده: سعید
تاریخ: ۱۳۹۱/۰۹/۰۹ ۱۱:۴۵

```
<iframe src="/twitter" frameborder="0" height="400" width="270" scrolling="no"></iframe>
```

فلسفه و کاربرد web pages متفاوت است. در مورد وب فرم‌ها هم نمی‌تونید از razor استفاده کنید چون در یک فایل نمی‌شود از دو موتور view استفاده کرد. موتور view وب فرم‌ها، نامش همین web forms view engine است و قابل تعویض هم نیست (برخلاف MVC). در مقاله‌ای که لینک دادید، داره از یک *iframe* استفاده می‌کنه برای الحاق razor.

به علاوه در مورد Web API که کلا مطلب جدیدی نیست. نسخه ساده شده WCF است.

نویسنده: محسن موسوی
تاریخ: ۱۳۹۱/۰۹/۰۹ ۱۲:۲۷

دوست عزیز
بحث ما در مورد توانایی انجام موارد فوق الذکر بود. نه اینکه حالا چون ...
به توانایی‌های MVC شکی نیست.مسئله اصلی اینه که آیا پست جاری را میتوان به راحتی با WebPage و یا WebForm انجام داد؟!
چه از طریق قابلیت‌های ASP.NET Web Form و ASP.NET Web Page راحته.
استفاده از Razor چه از طریق Jquery و یا Iframe (نظر قبلی)و یا به طور مستقل و یا روش‌های دیگر در ASP.NET Web Form و ASP.NET Web Page
و استفاده معمولی با توانایی‌های DataBind
یا بطور کامل از Razor View Engine در Web Page
در نهایت کار سختی نیست.

نویسنده: سعید
تاریخ: ۱۳۹۱/۰۹/۰۹ ۱۲:۴۳

razor بی‌جهت اینجا طرح شد. من در مورد strongly typed view سطری رو نوشتم. این نوع viewها مستقل از نوع view

engine هستند. در mvc حتی با موتور web forms هم میشه یک چنین viewهای تحت نظر کامپایلری رو با پشتیبانی کامل از intelisense داشت.

عنوان: ارسال انواع بی نام (Anonymous) بازگشتی توسط Entity framework به توابع خارجی
نویسنده: سید مهران موسوی
تاریخ: ۱۳۹۱/۰۹/۲۱ ۱۶:۵۱
آدرس: www.dotnettips.info
برچسب‌ها: C#, Entity framework, ASP.Net MVC, Anonymous Type

فرض کنید ساختار زیر را در مدل ساخته شده به وسیله‌ی Entity framework در پروژه‌ی خود داریم .

جدول Post با جدول GroupsDetail ارتباط یک به چند و در مقابل ان جدول GroupsDetail با جدول PostGroup ارتباط چند به یک دارد . به زبان ساده ما تعدادی گروه بندی برای مطالب داریم (در جدول PostGroup) و میتوانیم برای هر مطلب تعدادی از گروه‌ها را در جدول GroupsDetail مشخص کنیم ...

فکر میکنم همین مقدار توضیح به اندازه‌ی کافی برای درک روابط مشخص شده در مدل رابطه ای مفروض کفایت کند.

حالا فرض کنید ما میخواهیم لیستی از عنوان مطالب موجود به همراه [نام] گروه‌های هر مطلب را داشته باشیم .

توجه شما رو به قطعه کد ساده‌ی زیر جلب میکنم:

```
var context = new Models.EntitiesConnection();
var query = context.Posts.Select(pst => new {
            id = pst.id,
            Title = pst.Title,
            GNames = pst.GroupsDetails.Select(grd => new { Name = grd.PostGroup.name })
            })
            .OrderByDescending(c => c.id)
            .ToList();
```

همانطور که شاهد هستید در قطعه کد بالا توسط خواص راهبری (Navigation Properties) به صورت مستقیم نام گروه‌های ثبت شده برای هر مطلب در جدول GroupsDetail را از جدول PostGroup استخراج کردیم . خب تا اینجا مسئله ای وجود نداشت (البته با ORMها این کار بسیار سهل و اسان شده است تا جایی که در حالت عادی برای همین کار باید کلی join نویسی کرد و .... در کل خدارو شکر که از دست کارهای تکراری و خسته کننده توسط این موجودات مهربان (ORM's) نجات یافتیم)...

خب میرسیم به ادامه‌ی مطلبمون . نتیجه‌ی این query چه خواهد بود ؟ کاملا واضح است که ما تعداد دلخواهی از فیلدها رو برای واکشی مشخص کردیم پس در نتیجه نوع داده ای که توسط این query بازگشت داده خواهد شد یک لیست از یک نوع بی نام میباشد ... چگونه از نتیجه‌ی بازگشتی این query در صورت ارسال اون به عنوان یک پارامتر به یک تابع استفاده کنیم ؟ اگر ما تمامی فیلدهای جدول Post را واکشی میکریم مقدار بازگشتی یک لیست از نوع Post بود که به راحتی قابل استفاده بود مثل مثال زیر

```
public bool myfunc(List<Post> query)
        {
            foreach (var item in query)
            {
            .....  string title =  item.Title;
            }
...return true;
        }
.
.
.

var context = new Models.EntitiesConnection();
var queryx = context.Posts.ToList();
.... myfunc(queryx );
```

اما حالا با یک لیست از یک نوع بی نام رو به رو هستیم ... چند راه مختلف برای دسترسی به این گونه از مقادیر بازگشتی داریم .

1 : استفاده از Reflection برای دسترسی به فیلدهای مشخص شده .

2 : تعریف یک مدل کامل بر اساس فیلدهای مشخص شده‌ی بازگشتی و ارسال یک لیست از نوع تعریف شده به تابع . (به نظرم این روش خسته کنندست و زیاد شکیل نیست چون برای هر query خاص مجبوریم یک مدل کلی تعریف کنیم و این باعث میشه تعداد زیادی مدل در نهایت داشته باشیم که استفاده‌ی زیادی از اونها نمیشه عملا)

3 : استفاده از یکی از روش‌های خلاقانه‌ی تبدیل نوع Anonymousها .

روش سوم روش مورد علاقه‌ی من هست و انعطاف بالایی داره و خیلی هم شیکو مجلسیه ! در ضمن این روش که قراره ذکر بشه کاربردهای فراوانی داره که این مورد فقط یکی از اونهاست

خب بریم سر اصل مطلب . به کد زیر توجه کنید :

```
public static List<T> CreateGenericListFromAnonymous<T>(object obj, T example)
        {
            return (List<T>)obj;
        }
        public static IEnumerable<T> CreateEmptyAnonymousIEnumerable<T>(T example)
        {
            return new List<T>(); ;
        }

////

        public bool myfunc(object query)
        {
            var cquery = CreateGenericListFromAnonymous(query,
                                new {
                                        id = 0,
                                        Title = string.Empty,
                                        GNames = CreateEmptyAnonymousIEnumerable(new { Name =
string.Empty })
                                });
            foreach (var item in cquery)
            {
```

```
                string title = item.Title;
                foreach (var gname in item.GNames)
                {
                    string gn = gname.Name;
                }
.
.
.
            }
            return false;
        }

.
.
.
var context = new Models.EntitiesConnection();
var query = context.Posts.Select(pst => new
                                {
                                     id = pst.id,
                                     Title = pst.Title,
                                     GNames = pst.GroupsDetails.Select(grd => new { Name =
grd.PostGroup.name })
                                }).OrderByDescending(c => c.id);
            myfunc(query.ToList());
```

در کد بالا به صورت تو در تو از انواع بی نام استفاده شده تا مطلب براتون خوب جا بیوفته . یعنی یک نوع بی نام که یکی از فیلدهای اون یک لیست از یک نوع بی نام دیگست ... خب میبینید که خیلی راحت با دو تابع ما تبدیل انواع بی نام رو به صورت inline انجام دادیم و ازش استفاده کردیم . بدون اینکه نیاز باشه ما یک مدل مجزا ایجاد کنیم ...

تابع CreateGenericListFromAnonymous : یک object رو میگیره و اون رو به یک لیست تبدیل میکنه بر اساس نوعی که به صورت inline براش مشخص میکنیم .

تابع CreateEmptyAnonymousIEnumerable یک لیست از نوع IEnumerable رو بر اساس نوعی که به صورت inline براش مشخص کردیم بر میگردونه . دلیل اینکه من در اینجا این تابع رو نوشتم این بود که ما در query یک فیلد با نام GNames داشتیم که مجموعه ای از نام گروه‌های هر مطلب بود که از نوع IEnumerable هستش . در واقع ما در اینجا نوع بی نامی داریم که یکی از فیلدهای اون یک لیست از یک نوع بی نام دیگست . امیدوارم مطلب براتون جا افتاده باشه.

دوستان عزیز سوالی بود در قسمت نظرات مطرح کنید.

نظرات خوانندگان

نویسنده: سعید
تاریخ: ۱۳۹۱/۰۹/۲۱ ۱۷:۱۱

با تشکر. روش‌های دیگری هم برای بازگشت انواع بی‌نام و نشان وجود دارند:

- خروجی متد را object تعریف کنیم

- خروجی متد را یک لیست از نوع dynamic (سی شارپ 4) تعریف کنیم

- خروجی متد را فقط ienumerable تعریف کنیم (نیازی به ذکر t ندارد الزاما)

نویسنده: سید مهران موسوی
تاریخ: ۱۳۹۱/۰۹/۲۱ ۲۰:۴۷

ممنون از نکاتی که ذکر کردید . از نکات ذکر شده برجسته ترینش استفاده از انواع dynamic هستش که کار رو ساده میکنه ولی خب مشکلاتی رو هم به وجود میاره مثلا اشکال زدایی رو سخت میکنه. هر چند که این نوع کار خودش رو با Reflection در زمان اجرا انجام میده و استفاده از اون رو ساده میکنه ولی خب استفاده مستقیم از روش‌های سطح پایین‌تر مزایایی رو هم داره مثلا مثال زیر رو در نظر بگیرید

```
public bool sample(object query)
        {
            System.Reflection.BindingFlags flags = System.Reflection.BindingFlags.Public |
                System.Reflection.BindingFlags.NonPublic |
                System.Reflection.BindingFlags.Static |
                System.Reflection.BindingFlags.Instance |
                System.Reflection.BindingFlags.DeclaredOnly;
            foreach (var item in query.GetType().GetProperties(flags))
            {

                string test = item.GetValue(query,null).ToString();
                // Do Something ...
            }
            return false;
        }
```

خب حلا فرض کنید ما میخوایم یک Table Generator بسازیم که بر اساس لیست نتایج بازگشتی توسط query بازگشت داده شده توسط Entity framework یک جدول رو برامون ایجاد میکنه . طبیعی هست که هر بار ما یک نوع داده ای بی نام رو براش ارسال میکنیم اگه ما بخوایم نوع dynamic رو به عنوان پارامتر تابع تعریف کنیم نمیتونیم به فیلدهای نوع بی نام دسترسی داشته باشیم چرا که هر بار فیلدها فرق میکنه و این تابع قراره در زمان اجرا فیلدها رو تشخیص بده ولی در انواع dynamic ما نام فیلد رو در زمان طراحی مشخص میکنیم و در زمان اجرا توسط نوع dynamic بسط داده میشه که این موضوع واسه همچین مواردی کارایی نداره ... و باید در نظر داشته که انواع dynamic فقط میتونن به فیلدهای public دسترسی داشته باشن ولی ما با reflection میتونیم محدوده‌ی دسترسی رو مشخص کنیم...

و اما در رابطه با مطلب بالا : بر فرض ما مجموعه ای از داده‌ها رو توسط Entity framework واکشی کردیم و یک نوع بی نام داریم و حالا میخوایم مثلا با Table Generator فرضی دو جدول رو از همین یک بار واکشی ایجاد کنیم که هر کدوم شامل یک سری فیلدها هستن و یک سری فیلدها رو از نوع بی نام واکشی شده شامل نمیشن . ما میتونیم یک تابع داشته باشیم که لیست نوع بی نام مرجع رو براش ارسال کنیم و یک نوع بی نام هم به عنوان یک پارامتر به صورت inline براش بفرستیم که فیلدهای مورد نظرمون رو از نوع بی نام مرجع شامل میشه . حالا با کمی توسعه CreateGenericListFromAnonymous و مپ کردن نوع بی نام مرجع با نوع بی نام ارسال شده توسط پارامتر و استفاده از Reflection میتونیم فقط فیلدهایی رو که به صورت inline مشخص کردیم داشته باشیم و با یک بار واکشی اطلاعات چندین بار اون رو با شکل‌های مختلف پردازش کنیم و .... این فقط یک مثال بود و مطلب بالا صرفا ایده ای بود که دوستان بتونن کارهای خلاقانه ای رو از طریق اون انجام بدن ....

نویسنده: ایلیا
تاریخ: ۱۳۹۱/۰۹/۲۱ ۲۰:۵۹

عالی بود. من همیشه یک مدل کامل ایجاد می‌کردم. ولی حالا خیلی راحت شد. تشکر فراوان.

نویسنده: سجاد
تاریخ: ۱۳۹۱/۰۹/۲۲ ۱۷:۴۴

بسیار عالی ! اگه در مورد " **حالا با کمی توسعه CreateGenericListFromAnonymous و مپ کردن نوع بی نام مرجع با نوع بی نام ارسال شده توسط پارامتر و استفاده از Reflection میتونیم فقط فیلدهایی رو که به صورت inline مشخص کردیم داشته باشیم** " بیشتر توضیح بدین ممنون میشم

نویسنده: سید مهران موسوی
تاریخ: ۱۳۹۱/۰۹/۲۲ ۲۲:۱۹

خواهش میکنم قابل نداشت . دوست عزیز این مطلب واقعا پیچیده و تخصصی هستش من یک نمونه واستون نوشتم که کد رو براتون میزارم ولی برای فهم کاملش نیاز به اشنایی عمقی با ساختار دات نت و کار با رفلکشن داره که توضیحش در چند خط نمیگنجه بحثه یک کتاب کامله. این نمونه کد که نوشتم دقیقا همون چیزی هست که درخواست توضیحش رو دادید .

```
public static List<T> CreateGenericListFromAnonymous<T>(object obj, T example)
        {
            var newquery = new List<T>();
            var constructor = typeof(T).GetConstructors(
            System.Reflection.BindingFlags.Public | System.Reflection.BindingFlags.NonPublic |
System.Reflection.BindingFlags.Instance
            ).OrderBy(c => c.GetParameters().Length).First();
            foreach (var item in ((IEnumerable<object>)obj))
            {
                var mapobj = new object[example.GetType().GetProperties().Count()];
                int counter = 0;
                foreach (var itemmap in example.GetType().GetProperties())
                {

                    object value = item.GetType().GetProperty(itemmap.Name).GetValue(item, null);
                    Type t = itemmap.PropertyType;
                    mapobj[counter] = Convert.ChangeType(value, t);
                    counter++;
                }
                newquery.Add((T)constructor.Invoke(mapobj));
            }

            return newquery;
        }


var context = new Models.EntitiesConnection();
var query = context.Posts.Select(pst => new
                                {
                                     id = pst.id,
                                     Title = pst.Title,
                                      Likes = pst.Likes,
                                      Unlikes = pst.Unlikes
                                }).OrderByDescending(c => c.id);

object test_custom_casting = CreateGenericListFromAnonymous(query.ToList(), new { id = 0, Title =
string.Empty });
```

همینطور که میبینید نتیجه‌ی بازگشتی از یک query مرجع یک list شامل **فقط و فقط** فیلدهایی هست که به صورت inline توسط انواع بی نام مشخص شده ! امیدوارم مفید واقع شده باشه . یا حق

نویسنده: ناصر فرجی
تاریخ: ۱۳۹۲/۰۱/۱۸ ۱۲:۲۳

من همیشه از viewmodel استفاده میکنم و فک میکنم روش استانداردتری باشه. ممنون بابت اشتراک گذاری این مطلب.

نویسنده: pjimax
تاریخ: ۱۳۹۲/۰۲/۲۱ ۲۱:۹

با سلام و تشکر من مقاله شما رو در پروژه خودم پیاده سازی کردم.فقط جای دوتا چهارتا جدول join کردم اما return(List<T>)obj; خطا میده
لطفا راهنمایی کنید

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۲/۲۱ ۲۱:۲۱

خطا رو اگر نوشته بودی الان جواب گرفته بودی!

نویسنده: pjimax
تاریخ: ۱۳۹۲/۰۲/۲۲ ۱۰:۵۳

من میخام جداول زیر رو به روش بالا با هم join و استفاده کنم

```
var query = pro.types.Select(ty => new
       {
           typID = ty.type_id,
           typ_nam = ty.type_nam,
           cat_names = ty.categories.Select(cat => new
           {
               catgID = cat.cat_id,
               catg_name = cat.cat_nam,
               items = pro.items.Select(itm => new
               {
                   item_ID = itm.item_id,
                   item_nam = itm.item_nam,
                   item_path = itm.url,
                   coments = pro.comments.Select(cmm => new
                   {
                       comm_title = cmm.comment_nam,
                       comm_mail = cmm.mail,
                       comm_text = cmm.text,
                   })

               })


           })

       }).ToList();

 bool l = myfunc(query);

  protected bool myfunc(object q)
```

```
    {
        var cquery = CreateGenericListFromAnonymous(q,
            new
            {
                typID = 0,
                typ_nam = string.Empty,
                cat_names = CreateEmptyAnonymousIEnumerable(new
                {
                    catgID = 0,
                    catg_name = string.Empty,
                    items = CreateEmptyAnonymousIEnumerable(new
                    {
                        item_ID = 0,
                        item_name = string.Empty,
                        item_path = string.Empty,
                        coments = CreateEmptyAnonymousIEnumerable(new
                        {
                            comm_title = string.Empty,
                            comm_mail = string.Empty,
                            comm_text = string.Empty,

                        })
                    })
                })
            });
        foreach (var item in cquery)
        {
            int ID = item.typID;
            string type_title = item.typ_nam;
            foreach (var Cname in item.cat_names)
            {
                int categoryID = Cname.catgID;
                string category_name = string.Empty;

                foreach (var Iname in Cname.items)
                {
                    int ItmID = Iname.item_ID;
                    string ItmName = Iname.item_name;
                    string ItmPath = Iname.item_path;
                    foreach (var cm in Iname.coments)
                    {
                        string com_title = cm.comm_title;
                        string com_mail = cm.comm_mail;
                        string com_text = cm.comm_text;
                    }
                }
            }
        }
        return true;
    }
```

```
public class DBupload
{
    projeEntities pro = new projeEntities();
    uploadEntity up = new uploadEntity();
    public static List<T> CreateGenericListFromAnonymous<T>(object obj, T example)
    {
        return (List<T>)obj;
    }
    public static IEnumerable
    {
        return new List<T>();
    }
    public uploadEntity m(){
        uploadEntity tup = new
        //var query = from c
        //            join b
        //            join d
        //            join e
```

InvalidCastException was unhandled by user code

Unable to cast object of type 'System.Collections.Generic.List`1
[<>f__AnonymousType3`3
[System.Int32,System.String,System.Collections.Generic.IEnumerable`1
[<>f__AnonymousType2`3
[System.Int32,System.String,System.Linq.IQueryable`1
[<>f__AnonymousType1`4

Troubleshooting tips:

When casting from a number, the value must be a number less than infinity.
Make sure the source type is convertible to the destination type.
Get general help for this exception.

Search for more Help Online...

Actions:

View Detail...
Copy exception detail to the clipboard

Errors | 10 Warnings | 0 Messa

Description

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۲/۲۲ ۱۱:۳۷

```
IList<Types> typesList = context.Types.Include(x=>x.Categories)
                                      .Include(x=>x.Items).Include(x=>x.Comments).ToList();
```

اگر از EF استفاده می‌کنید، استفاده از متد Include کار جوین رو برای شما انجام می‌ده. بعدش نیازی به استفاده از متد CreateGenericListFromAnonymous که فقط یک سطح رو بررسی می‌کنه نیست. برای بررسی بیشتر از یک سطح باید متد بازگشتی نوشته بشه ولی در حالت شما واقعا نیازی نیست. ضمنا متد تصویر شما با متد نوشته شده یکی نیست.

عنوان: **چندین Submit در یک Html Form و انتساب Action های مجزا به هر یک از Submit ها در MVC**
نویسنده: سید مهران موسوی
تاریخ: ۱۳۹۱/۰۹/۲۲ ۱:۰
آدرس: www.dotnettips.info
برچسب‌ها: C#, ASP.Net MVC

تا به حال به این نکته برخورد کردید که برای یک فرم Html نیاز به چندین Submit داشته باشید که هر کدوم یک Action مجزا داشته باشن و یک کار متفاوت انجام بدن ؟
برای مثال فرمی داریم که داده‌های وارد شده در ان باید به دو صورت برای یک کاربر ارسال بشن یا از طریق پیامک یا از طریق ایمیل (این فقط یک مثال پیش فرض هست) و .... در حالت عادی ما در یک فرم نمیتونیم دو عدد Submit داشته باشیم که هر کدوم به یک Action جدا بسط داده بشه خب راه حل چیه ؟ شاید با خودتون بگید خب دو input از نوع radio قرار میدیم و در یک اکشن کنترل میکنیم که کدوم یکی انتخاب شده و عملیات رو با اون معیار انجام میدیم ... به نظرتون زیباتر نیست برای هر عملیات که ممکن باشه هر کدوم کاملا روال کاری متفاوتی داشته باشه یک Action وجود داشته باشه ؟ در این صورت خوانایی کد خیلی بالاتر میره و Unit Test هر Action کاملا مشخص هست که قراره چه فرایندی رو مورد تست قرار بده و مجبور نیستیم چندین حالت رو با عبارات شرطی از هم جدا کنیم و همه چی قاطی بشه با هم ... من در کل با امکاناتی که C# و MVC در اختیارم قرار میده حاظر نیستم تن به کد نویسی به صورت کلاسیک و قاطی پاتی بدم سعی میکنم با مطالعه‌ی سورس MVC بهترین حالت رو انتخاب کنم شما چطور ؟ معلومه که همه همینو میخوان پس بریم سر اصل مطلب .
قطعه کد Html و Razor ساده‌ی زیر رو در نظر بگیرید برای View :

```
@model Models.MyModel
@{
    Layout = null;
}
<!DOCTYPE html>
<html>
<head>
    <title>ViewPage1</title>
</head>
<body>
    <div>
        @using (Html.BeginForm("SendMessage", "Home", FormMethod.Post))
        {

            @Html.LabelFor(x => x.Name);
            @Html.TextBoxFor(x => x.Name);

            <input type="submit" value="ارسال توسط پیامک" name="Send_sms" />
            <input type="submit" value="ارسال توسط ایمیل" name="Send_email" />
        }
    </div>
</body>
</html>
```

خب ما دو تا Submit داریم . یکم اگه شیطنت کنید و مقادیر ارسال شده بعد از submit این فرم رو توسط ابزارهای مانیتورینگ بررسی کنید میبینید که روی هر کدوم از Submitها که کلیک میشه داده ای با نام اون که در خاصیت name اون و مقدار موجود در value اون همراه اون فرم به سرور ارسال میشه و اون یکی Submit از این اتفاق بی نصیب میمونه ... خب ما هم استفاده‌ی لازم رو از این موضوع شیرین میبریم و با یک تکنیک تهاجمی از این موضوع برای رسیدن به هدفمون استفاده میکنیم .

این هم کلاس Model ماست :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.ComponentModel;

namespace Models
{
    public class MyModel
    {
        [DisplayName("نام خود را وارد کنید :")]
```

```
        public string Name { get; set; }
    }
}
```

و اما یک نکته‌ی دیگه . توجه داشته باشید که ما در قسمت View نام Action رو در فرم, SendMessage مشخص کردیم . ولی ... اصلا در واقع همچین اکشنی وجود نداره ! پس چرا ما همچین نامی رو واسه اکشن فرم گذاشتیم !؟
دلیل اینه که ما قصد داریم با یک ActionNameSelectorAttribute درخواست کاربر رو شکار کنیم و اون رو به اکشن دلخواه ارجاع بدیم ... جالبه نه ؟ ولی چه جوری ... کلاس زیر رو بهش دقت مضاعف کنید و در پروژتون ایجادش کنید :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using System.Reflection;

namespace ActionHandlers
{
    public class SendMessageHandlerAttribute : ActionNameSelectorAttribute
    {
        public override bool IsValidName(ControllerContext controllerContext, string actionName,
MethodInfo methodInfo)
        {
            if (actionName.Equals(methodInfo.Name, StringComparison.InvariantCultureIgnoreCase))
                return true;

            if (!actionName.Equals("SendMessage", StringComparison.InvariantCultureIgnoreCase))
                return false;

            var request = controllerContext.RequestContext.HttpContext.Request;
            return request[methodInfo.Name] != null;
        }
    }
}
```

خب حالا بخش Controller رو بهش دقت کنید که ما در اون دو اکشن رو با نام هایی که برای هر Submit مشخص کردیم مینویسیم و ActionNameSelectorAttribute نوشته شده رو به اونها بسط میدیم.

```
[SendMessageHandler ]
        [HttpPost]
        public ContentResult Send_sms(MyModel mdl)
        {
            /// Do something ...
           return string.Empty ;
        }

        [SendMessageHandler ]
        [HttpPost]
        public ContentResult Send_email(MyModel mdl)
        {
           /// Do something ...
          return  string.Empty;
        }
```

خب حالا بعد از کلیک بر روی هر Submit اکشن متناظر با اون اجرا میشه . بعد از ارسال درخواست به سرور MVC در بین اکشن‌های موجود در Controller مشخص شده به دنبال اکشن معین شده میگرده و وقتی به اکشن‌های ما میرسه میبینه عجب ! اون دوتا ActionNameSelectorAttribute سفارشی دارن پس میره ببینه چه خبره اونجا که ما با یک حرکت تهاجمی بررسی میکنیم که اگه نام اکشن مشخص شده در فرم با نام اکشن در حال بررسی مساوی بود که همینو اجرا کن (یعنی ما میتونی اکشنی با نام SendMessage هم داشته باشیم) . اگه نام اکشن مشخص شده در فرم اون نامی نبود که ما میخوایم که کلا بیخیال هندل کردن اکشن میشیم میزاریم خود MVC تصمیم بگیره . و در اخر بررسی میکنیم که ایا در درخواست جاری مقداری با نام اکشن در حال بررسی وجود داره !؟ اگه داشت یعنی همون Submit که ما میخوایم وصل بشه به این اکشن کلیک شده پس اکشن در حال بررسی رو بسط میدیم به درخواست ارسال شده ... به همین سادگی ...

پیروز و موفق باشید .

نظرات خوانندگان

نویسنده: سعید
تاریخ: ۱۳۹۱/۰۹/۲۲ ۱۷:۳۱

راه جالبی است. نمی‌دونستم که ActionNameSelectorAttribute وجود خارجی دارد!
چند راه حل دیگر:
الف) استفاده از قابلیت‌های binding . مثلا اگر نام پارامترها را به نام همان دکمه‌های موجود تنظیم کنیم، این نام موقع دریافت از دکمه کلیک شده، نال نیست:

```
[HttpPost]
public ActionResult MyAction([other params here], string buttonName1, string buttonName2, etc)
{
     if(!string.IsNullOrEmtpy(buttonName1)) { //button1 was clicked}
}
```

ب) میشه از ویژگی‌های جدید HTML5 مثل data-form-action استفاده کرد:

```
<input type="submit" value="Standard action">
<input data-form-action="@Url.Action("mysecondaction")" type="submit" value="Second action">
```

بعد موقع ارسال کمی از jQuery استفاده کرد

```
$(document).on('click', '[type="submit"][data-form-action]', function(event) {
    var $this = $(this),
        formAction = $this.attr('data-form-action'),
        $form = $($this.closest('form'));
        $form.attr('action', formAction);
});
```

نویسنده: سید مهران موسوی
تاریخ: ۱۳۹۱/۰۹/۲۲ ۱۸:۱

ممنون دوست عزیز که توجه میکنی به مطالب بنده . به جز روشی که در مقاله بالا ذکر شد استفاده از data-form-action موجود در نگارش 5 از HTML اصولیترین روش یا بهتره بگیم تمیزترین روش در بین مواردی هست که ذکر کردید ولی خب برنامه نویسای وب همیشه با مرورگرهای مختلف در جنگ هستن و فعلا زیاد استفاده از این روش جایز نیست چون بعضی از مرورگرهای کمی قدیمیتر پشتیبانی نمیکنند پس بهتره از تکنیکهای مورد اطمینانتر استفاده کنیم که نمونش در مقاله‌ی بالا ذکر شد ...

این نمونه روش‌های خلاقانه و جالب از جمله ActionNameSelectorAttribute به وفور در نسخه‌های جدید mvc وجود داره خوشبختانه . در مقالات بعدیم سعی میکنم مطالب تخصصی و تکنیکهای جالب دیگرو هم با دوستان به اشتراک بزارم . انصافا mvc یکی از شیرین‌ترین تکنلوژی هایی هست که هر سری سورسش رو مطالعه میکنم راه کارهای جدیدی برای پیاده سازی نرم افزارها توسط اون تکنیک‌ها دستگیرم میشه ...
پاینده باشید و موفق

عنوان: **مدیریت سفارشی سطوح دسترسی کاربران در MVC**
نویسنده: سید مهران موسوی
تاریخ: ۱۳۹۱/۱۰/۰۸ ۲۳:۱۵
آدرس: www.dotnettips.info
گروه‌ها: MVC, ASP.Net MVC

فرض کنید در پروژه‌ی جاری خودتون قصد دارید یک سیستم مدیریت سطوح دسترسی کاربران رو با انعطاف بالا پیاده سازی کنید . مثلا سیستم شما دارای صفحات مختلفی هستش که هر گروه کاربری اجازه دسترسی به هر صفحه رو نداره ... هدف اینه که شما این گروه‌های کاربری و سطوح دسترسی برای هر گروه رو برای سیستم مشخص کنید .
مثلا فقط کاربرانی که دسترسی admin دارن بتونن به صفحات مدیریتی دسترسی داشته باشن و ....
برای این منظور در دات نت کلاسی با نام Role Provider وجود داره که در ادامه‌ی این مبحث به کار با اون میپردازیم.
مثلا فرض کنید قرار بر اینه که سطوح دسترسی رو از بانک اطلاعاتی استخراج کنیم . کلاس مذکور در فضای نام System.Web.Security قرار گرفته . برای شروع ما نیاز داریم یک نمونه از این کلاس رو پیاده سازی کرده و چند تابع از اون رو بازنویسی کنیم .

پیاده سازی کلاس به این صورته :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Security;

namespace Myproject.Security
{
    public class CustomRoleProvider : RoleProvider
    {

    }
}
```

خب در مرحله‌ی بعد دو تابع از این کلاس رو بازنویسی میکنیم . اول تابع GetRolesForUser که در این مقاله وظیفه‌ی استخراج لیست مجوزها برای هر کاربر رو از بانک اطلاعاتی داره که به شکل زیر پیاده سازی میشه .

```
public override string[] GetRolesForUser(string username)
        {
            using (DatabaseEntities db = new DatabaseEntities())
            {
                User user = db.Users.FirstOrDefault(u => u.UserName.Equals(username,
StringComparison.CurrentCultureIgnoreCase));

                var roles = from ur in user.UserRoles
                            from r in db.Roles
                            where ur.RoleId == r.Id
                            select r.Name;
                if (roles != null)
                    return roles.ToArray();
                else
                    return new string[] {}; ;
            }
        }
```

همونطور که میبینید در تابع بالا از کلاس CustomRoleProvider ما عملیات استخراج لیست مجوزهای دسترسی مربوط به هر کاربر رو از بانک اطلاعاتی انجام دادیم . توجه داشته باشین که شما میتونید این واکشی رو از هر محیط دیگه ای جز بانک اطلاعاتی هم بسته به نوع کارتون انجام بدید .

تابع بعد IsUserInRole نام داره که با بازنویسی اون مشخص میکنیم که ایا یک کاربر دارای مجوز لازم برای دسترسی هست یا نه . اون رو به شکل زیر بازنویسی میکنیم.

```
public override bool IsUserInRole(string username, string roleName)
        {
```

```
            using (DatabaseEntities db = new DatabaseEntities())
            {
                User user = db.Users.FirstOrDefault(u => u.UserName.Equals(username,
StringComparison.CurrentCultureIgnoreCase));

                var roles = from ur in user.UserRoles
                            from r in db.Roles
                            where ur.RoleId == r.Id
                            select r.Name;
                if (user != null)
                    return roles.Any(r => r.Equals(roleName,
StringComparison.CurrentCultureIgnoreCase));
                else
                    return false;
            }
        }
```

همونطور که شاهد هستید در تابع بالا بعد از واکشی لیست مجوزهای ثبت شده برای هر کاربر بررسی انجام میشه که ایا اولا کاربر یک کاربر ثبت شده هست و ثانیا اینکه ایا درخواستی که ارسال کرده برای دسترسی به یک بخش مجوز اون رو داره یا خیر ...

این **نکته** رو یاد آور بشم که این توابع رو میشه به شکل‌های مختلفی پیاده سازی کرد و اونچه که در اینجا نوشته شده **فقط جهت مثال** هستش . مثلا تابع IsUserInRole رو میشه به شکل زیر هم نوشت و این بسته به شرایط کاری داره که قصد انجام اون رو دارید ....

```
public override bool IsUserInRole(string username, string roleName)
        {
  return this.GetRolesForUser(username).Contains(roleName);
        }
```

خب میرسیم به بخش معرفی این Provider در web.config که به صورت زیر انجام میشه ...

```
<system.web>
    ...
    <rolemanager cacherolesincookie="true" defaultprovider="CustomRoleProvider" enabled="true">
        <providers>
            <clear />
            <add name="CustomRoleProvider" type="Myproject.Security.CustomRoleProvider" />
        </providers>
    </rolemanager>
    ...
</system.web>
```

توجه داشته باشید که مجوزهای هر کاربر با معرفی بالا بعد از یک بار واکشی در کوکی ذخیره میشه و دیگه هر بار، بار اضافه برای واکشی از بانک اطلاعاتی به برنامه تحمیل نمیشه ...

حالا به این صورت میتونیم مثلا یک Controller رو محافظت کنیم در برابر درخواست از جانب کاربرانی که سطح دسترسی به اون رو ندارند .

```
using System;
using System.Web.Mvc;

namespace MyProject.Areas.Admin.Controllers
{
    [Authorize(Roles = "Administrators")]
    public class HomeController : Controller
    {
        //
        // GET: /Admin/Home/

        public ActionResult Index()
```

```
        {
            return View();
        }

    }
}
```

توجه داشته باشید که کنترل مجوز برای بررسی وجود مجوز در بخش‌های کوچکتر هم مانند اکشن‌ها و ... میتونه در نظر گرفته بشه .

نظرات خوانندگان

نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۱/۱۰/۰۹ ۰:۱۵

فکر می‌کنم خلاصه مطلب شما می‌شه :
نمونه گیری از اینترفیس role provider و پیاده سازی شخصی این کلاس
نظر دوستان رو جلب می‌کنم به این سری مطلب که هرچند قدیمیه ولی واقعا مفیده.
به خود من که خیلی کمک کرد.

نویسنده: سعید
تاریخ: ۱۳۹۱/۱۰/۰۹ ۲:۳۱

ممنون. کلاس‌های users و roles به چه نحوی تعریف شدن؟ به نظر می‌رسه رابطه many-to-many است.

نویسنده: سید مهران موسوی
تاریخ: ۱۳۹۱/۱۰/۰۹ ۱۰:۲

سه جدول user و UserRoles و Roles در نظر گرفته شده که جدول UserRoles بین دو جدول دیگه واسط هست . هر کاربر میتونه چندین مجوز داشته باشه و هر مجوز میتونه برای چندین کاربر باشه . ارتباط n به n بین دو جدول Roles و user به واسطه‌ی جدول UserRoles برقراره.
توجه به این نکته حائز اهمیته که این مدل فقط یک مثال ساده صرفا مربوط به این مقالست و هیچ جنبه‌ی دیگه ای نداره و همونطور که تو مقاله هم به صورت ضمنی اشاره شد به هر روش دیگه ای بسته به کارتون میشه پیاده سازیش کرد.

نویسنده: سید مهران موسوی
تاریخ: ۱۳۹۱/۱۰/۰۹ ۱۰:۱۳

دوست عزیز RoleProvider یک اینترفیس نیست بلکه یک کلاس مشتق شده از ProviderBase هستش که اونم از کلاس system.object مشتق شده . برای اطلاعات بیشتر توجهتون رو به صفحه‌ی مربوطه در msdn جلب میکنم : RoleProvider

نویسنده: علیرضا
تاریخ: ۱۳۹۱/۱۰/۰۹ ۱۱:۱۵

دوست عزیز سلام،
ممنون بابت آموزش خوبتون امکانش هست یه کم در باره این بخش از صحبت تون و نحوه پیاده سازیش بیشتر توضیح بدید؟

[توجه داشته باشید که کنترل مجوز برای بررسی وجود مجوز در بخش‌های کوچکتر هم مانند اکشن‌ها و ... میتونه در نظر گرفته بشه .]

نویسنده: سید مهران موسوی
تاریخ: ۱۳۹۱/۱۰/۰۹ ۱۲:۱۹

شما فیلتر Authorize رو میتونید برای یک اکشن هم در نظر بگیرید مثل مثال زیر . با این کار دیگه کل کنترلر به مجوزی که ذکر میکنید محدود نمیشه و فقط یک اکشن از اون محدود میشه...

```
using System;
using System.Web.Mvc;

namespace MyProject.Areas.Admin.Controllers
{
    public class HomeController : Controller
    {
        //
```

```
        // GET: /Admin/Home/

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

        [Authorize(Roles = "Administrators")]
        public ActionResult ViewProfile()
        {
            return View();
        }
    }
}
```

نویسنده: رامین
تاریخ: ۱۳۹۱/۱۰/۰۹ ۱۲:۲۹

ممنون بابت مطلب کامل شما ،
سوالی داشتم آیا در مبحث ASP.NET هم میتوان از این روش استفاده نمود؟

نویسنده: سید مهران موسوی
تاریخ: ۱۳۹۱/۱۰/۰۹ ۱۲:۵۸

خواهش میکنم دوست عزیز . بله میشه استفاده کرد . دعوتید به مطالعه‌ی این مبحث در MSDN :

How To: Use Role Manager in ASP.NET

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۱۰/۱۰ ۹:۱۰

خوبه ولی برای انعطاف پذیری بیشتر، من و تیمم با استفاده از Reflection، اسامی متدهایی که خروجی ActionResult دارند رو بازیابی می‌کنیم و در سیستم امنیت و پایه برای مدیر امنیت و برنامه نویس نشان میدیم و اونها می‌تونن دسترسی رو بر این مبنا تنظیم کنن. وجود یک Contoller پایه و یک فیلتر برای اون با override کردن متد OnActionExecuting از الزامات کار هست.

نویسنده: سید مهران موسوی
تاریخ: ۱۳۹۱/۱۰/۱۰ ۱۲:۷

جالبه . من در پروژه ای که در حال کار کردن روی اون هستم که یک CMF هست تقریبا همین کار رو انجام میدم . به این شکل که SecurityEngine این سیستم اکشن‌های مشخص شده که نیاز به سطح دسترسی دارن رو بر اساس یک attribute سفارشی شده استخراج میکنند به واسطه‌ی reflection و مدیران سیستم میتونن برای هر اکش roleهای دلخواه رو انتصاب بدن و سطوح هر اکشن رو توسط یک AuthorizeAttribute و با بازنویسی AuthorizeCore از پایگاه داده استخراج میکنم . هر کاربری که سطح دسترسی معین رو داشته باشه میتونه به اون اکشن دسترسی پیدا کنه ... البته پشت این SecurityEngine یک CashManagment هم قرار داره که از بار اظافه‌ی هر بار رفتن به پایگاه داده برای استخراج سطوح جلوگیری میکنه و این پروسه رو مدیریت میکنه

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۱۰/۱۰ ۱۲:۵۵

وجود یک مدیر Cache برای سیستم سطح دسترسی برای من ناملموس هست. اینکه چطور میشه در یک سیستم با حدود 5000 کاربر همزمان، نقش‌ها رو در Cache نگه داشت؟ ساختار ذخیره سازی در حافظه به چه شکل هست؟ ترجیح من این هست که به ازای هر درخواست، پایگاه داده مورد اشاره قراره بگیره.

نویسنده: سید مهران موسوی
تاریخ: ۱۳۹۱/۱۰/۱۰ ۱۳:۲۶

نقش‌های کاربران در کش نگهداری نمیشه . اونها در کوکی‌ها ذخیره میشن برای بهینه کردن و کم کردن فشار از روی سیستم در

ازای دسترسی هر با به پایگاه داده . با مشخص کردن cacherolesincookie مربوط به rolemanager در web.config....

و اینکه یک سیستم مجوزهای 5000 کاربر یا حتی 5 میلیون کاربر رو در کش ذخیره کنه برای من هم نه حتی ناملموس بلکه یک اشتباه محض هست چون بعد از یک مدت با برخورد مدیران it سرور و suspend شدن مواجه میشیم ... کار CacheManager که ازش حرف زدم ذخیره کردن مجوزهای اکشن هاست در حافظه که با هر بار Request برای اکشن دیگه نریم و مجوزهای اون رو از پایگاه داده بخونیم . شما بر فرض 100 اکشن مدیریتی دارید که هر کدوم هم 10 مجوز براشون مشخص شده در کل حساب بشه چیز زیادی نمیشه ولی در عوض سرعت بالا رو در قبال نرفتن به پایگاه داده براتون به ارمغان میاره.

اینکه بیان کردید به ازای هر درخواست پایگاه داده مورد اشاره قرار بگیره خیلی رو کارایی سیستم تاثیر منفی میزاره . به استناد گفته‌ی خودتون 5000 کاربر داریم که بر فرض در ثانیه در بدبینانه‌ترین حالت ممکن 100 درخواست ارسال میکنند . یعنی ما باید در هر ثانیه برای استخراج سطوح هر اکشن 100 بار به بانک اطلاعاتی مراجعه کنیم ؟ حال فرض کنید این درخواست‌ها در ثانیه 5000 تا و کاربران ما 5 میلیون تا باشن ... به نظرم این سیستم قبل از پیاده سازی شکستش قطعیه ...

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۱۰/۱۰ ۱۳:۵۹

منظورتون Cache کوکی هست. اگر نقش‌های کاربر تغییر کنه، اِعمال بلافاصله‌ی تغییرات در حین گشت و گذار وی در سایت به چه شکل هست؟

نویسنده: سید مهران موسوی
تاریخ: ۱۳۹۱/۱۰/۱۰ ۱۴:۳۲

Cache کوکی برای نگهداری نقش‌های کاربر و Cache سمت سرور برای نقش‌های مشخص شده هر اکشن ...

برای موردی هم که اشاره کردید راه‌های مختلفی میتونه وجود داشته باشه . مثلا شما یک نقش جدید رو برای یک کاربر مشخص کردید . یک کلاس تعریف میکنیم که نگهدارنده‌ی لیست شناسه کاربرانی هست که نقششون در **زمانی که انلاین هستن** تغییر کرده . بعد از اعمال نقش به این کاربران این لیست بروز رسانی و در کش سمت سرور ذخیره سازی میشه . توجه داشته باشید که این لیست فقط زمانی بروز میشه که نقش کاربری تغییر کرده که **انلاین** هست. بعد در زمان درخواست یک اکشن توسط یک کاربر، اون لیست که در کش هست و واکشیش با سرعت بسیار بالا انجام میشه بررسی میشه و اگه نام اون کاربر در اون لیست بود مجددا میتونیم لیست نقش‌های اون کاربر رو از بانک استخراج و کوکی اون رو به روز رسانی کنیم . و در اخر هم نام اون از لیست حذف و کش نگهدارنده‌ی لیست به روز میشه ....

این روش میتونه روش خوبی باشه به این دلیل که مگه ممکنه نقش چند کاربر در زمانی که انلاین هستن تغییر کنه ؟ به طور حتم این تعداد خیلی کم هستن پس کلاس نگهدارنده‌ی شناسه‌ی کاربران مربوطه دارای حجم بسیار کمی هست و فظای خیلی کمی رو از حافظه سرور میگیره ولی در عوض سیستم یک سیستم جاندار و منعطف میشه ... در آخر ذکر این نکته که این روش صرفا یک ایده بود که در زمان نوشتن این متن تحلیل شد و با زمان گذاشتن روی اون و ایده‌های مشابه میشه به نتایج عالی رسید . سیستم من از همچین مکانیزمی به صورت خیلی کاملتر و با جزئیات بالا استفاده میکنه

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۱۰/۱۰ ۱۶:۵۳

مرسی از توضیحاتت.

نویسنده: بهروز راد
تاریخ: ۱۳۹۱/۱۰/۱۰ ۱۹:۵۹

گاهی اوقات مشکل از اینجا است که ما به صرف شنیده‌ها و پس زمینه‌های فکری، پیچیده‌ترین روش‌ها رو پیاده سازی می‌کنیم تا

اون پس زمینه‌ی فکری رو سرکوب کنیم. پایگاه‌های داده‌ی امروزی بسیار قدرتمند و پخته هستند و اگر قرار هست با یک Connection اضافه کل سیستم زیر سوال بره، بهتره بساط اون پایگاه داده جمع بشه.
من فکر می‌کنم باید حالت‌های دیگه ای رو هم بررسی کنی. مثلاً اینکه تغییر در نقش کاربر بوده یا مجوزهای نقش. اگر کاربر در حین پر کردن یک فرم AJAXیی مجوزش گرفته شد چه واکنشی باید رخ بده؟ آیا تمامی این بررسی‌ها و پیاده سازی اونها، به ایجاد یک ارتباط ساده و واکشی مجوز جاری کاربر برای درخواست برتری دارند؟
اگر امکانش رو داری، یک بررسی زمانی بین اون سیستم مدیریت Cache (با در نظر گرفتن تمامی حالت ها) و ایجاد ارتباط ساده با پایگاه داده برای بررسی داشتن مجوز کاربر انجام بده.

نویسنده: سید مهران موسوی
تاریخ: ۱۳۹۱/۱۰/۱۰ ۲۱:۴۲

اقای راد من هیچوقت قصد شعار دادن بر مبنای تحلیلات ذهنیم رو نداشتم و نخواهم داشت .
تمام حالت هایی که تا الان اشاره کردید قابل پوشش هست و در سیستم من در نظر گرفته شده و همچنان مورد تست قرار میگیره و ضعف هاش برطرف میشه . پیاده سازی یک بار انجام میشه ولی استفاده از اون بعد از پیاده سازی در هر کدوم از پروژه هایی که لازم باشه به سادگی ممکن میشه (به نظرم برنامه نویسی شیء گرا زیباست) . اگر توسعه دهندگان همون پایگاه‌های داده‌ی قدرتمند که ذکر کردید همچین طرز فکری داشتن که : (آیا تمامی این بررسی‌ها و پیاده سازی اونها، به ایجاد یک ارتباط ساده و واکشی مجوز جاری کاربر برای درخواست برتری دارند؟) ، هیچوقت قدرتمند و پخته نمیشدند ... این موضوع بر هیچکس پوشیده نیست که دسترسی مکرر به پایگاه داده برای برنامه هایی که ارزشمند هستند و سرعت و انعطاف در اونها مهمه یک نقطه‌ی ضعف هست . شاید همینجور طرز فکر باشه که باعث شده سیستم هایی که در بعضی از اماکن و سازمان هایی دولتی ما استفاده میشه بعد از جا افتادن و زیاد شدن داده‌ها واقعا کار کردن باهاشون عذاب آور و کسل کننده باشه...

انشاالله در چندین ماه اینده بعد از کامل شدن سیستم و انتشار اون میتونید بررسیش کنید اگر مایل بودید. کیت‌های توسعه‌ی اون هم به صورت سورس باز منتشر خواهد شد و در همین سایت هم معرفی میشن. به نظر من ارزشش رو داره که روی جزئیات این سیستم کار کنم و امیدوارم بتونه جای تامل داشته باشه . در اخر من رو ببخشید برادر من هیچوقت قصد جسارت نداشتم به شما و دیگر دوستان . این سایت محیط مقدسی هست چون واقعا مطالب خوبی در اون قرار میگیره امیدوارم بتونم از اطلاعات شما و دیگر دوستان استفاده کنم . یا حق

نویسنده: ابوالفضل
تاریخ: ۱۳۹۱/۱۰/۱۲ ۱۳:۴

سلام
من از سیستم معمولی وب فرم برای تشخیص هویت در mvc 4 استفاده کردم. اما به یه مشکل جالب برخوردم.
اپلیکیشین وب من روی شبکه داخلی کار میکنه و هر کس با نام کاربری خودش وارد میشه و کار میکنه.
بعد از لوگین ریدایرکت میکنم و در صفحه مقصد در هنگام User.Identity.IsAuthenticated همیشه false برمیگردونه و نمی‌تونم وارد شم. با هیچکدوم از مرورگرها وارد نمیشه ولی با **فایرفاکس** میتونم وارد شم!
من بدین صورت کد لوگین رو نوشتم:

```
if (Membership.ValidateUser(UserName, Password))
            {
                FormsAuthentication.RedirectFromLoginPage(UserName, true);
                return true;
            }
```

دوستان نظرشون چیه؟ مشکل از چی می‌تونه باشه؟ فایرفاکس چه خاصیتی داره که میتونه لوگین کنه؟ اما با IE و Chrome و safari امتحان کردم نشد.
در وب کانفیگ هم اینجوری تنظیم کردم

```
<authentication mode="Forms" >
      <forms loginUrl="~/Account/Login"
```

```
                protection="All"
                timeout="30"
                name="myAppCookie"
                path="/"
                requireSSL="false"
                slidingExpiration="true"
                cookieless="UseCookies"
                enableCrossAppRedirects="false" />
    </authentication>

    <membership >
      <providers>
        <clear />
        <add name="AspNetSqlMembershipProvider" type="System.Web.Security.SqlMembershipProvider"
            connectionStringName="ApplicationServices"  enablePasswordRetrieval="false"
            enablePasswordReset="true" requiresQuestionAndAnswer="false"
            requiresUniqueEmail="false" maxInvalidPasswordAttempts="5" minRequiredPasswordLength="6"
            minRequiredNonalphanumericCharacters="0" passwordAttemptWindow="10" applicationName="myApp
" />
      </providers>
    </membership>

    <profile>
      <providers>
        <clear />
        <add name="AspNetSqlProfileProvider" type="System.Web.Profile.SqlProfileProvider"
                  connectionStringName="ApplicationServices" applicationName="myApp " />
      </providers>
    </profile>

    <roleManager enabled="true">
      <providers>
        <clear />
        <add name="AspNetSqlRoleProvider" type="System.Web.Security.SqlRoleProvider"
                connectionStringName="ApplicationServices" applicationName="myApp "/>

        <add name="AspNetWindowsTokenRoleProvider"
             type="System.Web.Security.WindowsTokenRoleProvider" applicationName="myApp "  />
      </providers>
    </roleManager>
```

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۰/۱۲ ۱۳:۲۶

- اولین بررسی: مراجعه کنید به قسمت 18 سری MVC به متد public ActionResult LogOn ، یک FormsAuthentication.SetAuthCookie هم باید پیش از Redirect اضافه شود.

نویسنده: ابوالفضل
تاریخ: ۱۳۹۱/۱۰/۱۲ ۱۴:۴۶

با تشکر آقای نصیری. حل شد. مشکل همون بود. کلی سرچ کرده بودم و اون رو هم امتحان کرده بودم، منتها در هنگام کپی روی سرور نهایی برای تست، فایل خروجی تغییر یافته رو اشتباهی کپی می‌کردم. الان فهمیدم قضیه این بوده! :)
ولی باز هم برام سوال هست که چرا فایرفاکس لوگین میکرد ولی بقیه نه؟!

نویسنده: مهتدی حسن‌پور
تاریخ: ۱۳۹۱/۱۱/۱۴ ۱۲:۲۵

با سلام جناب راد
امکانش هست کمی بیشتر درمورد روش‌تون توضیح بدید؟ اگه لینکی هم در این رابطه دارید ممنون می‌شم در اختیارم بذارید.

نویسنده: سعید یزدانی
تاریخ: ۱۳۹۱/۱۱/۲۴ ۱۰:۱۹

با سلام
این دستور رو من متوجه نشدم میشه یه توضیح مختصری بدید

```
StringComparison.CurrentCultureIgnoreCase
```

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۱/۲۴ ۱۱:۱۸

- در مورد تاثیر Culture به مطلب « از متد DateTime.ToString بدون پارامتر استفاده نکنید! » مراجعه کنید.
- IgnoreCase سبب میشه مقایسه انجام شده، حساس به حروف بزرگ و کوچک نباشد.

نویسنده: پژمان پارسائی
تاریخ: ۱۳۹۱/۱۲/۱۸ ۲۰:۲

ممنون بابت مطلب مفیدی که در اختیار ما قرار دادید.
گفتید که مجوزها بعد از یک بار واکشی در کوکی ذخیره میشه، این به خاطر وجود مقدار true برای خصوصیت cacheRolesInCookie هست درسته؟ اما من یک RoleProvider سفارشی تهیه کردم که به صورت زیر اونو به وب کانفیگ اضافه کردم:

```
<roleManager cacheRolesInCookie="true" enabled="true" defaultProvider="CustomRoleProvider">
      <providers>
        <clear />
        <add name="CustomRoleProvider"
             type="MvcApp.UserInterface.Models.CustomRoleProvider"
             connectionStringName="MvcAppDb"
             applicationName="/" />
      </providers>

</roleManager>
```

اما با مراجعه به کوکی‌های مروگر چیزی نمی‌بینم، در ضمن یک Break Point هم سر متد GetRolesForUserگذاشتم که هر وقت قصد دیدن یک اکشن برای یک Role رو مشخص دارم این متد اجرا میشه یعنی از کوکی نقش رو نمی‌خونه.
سوال بعدیم اینه که این نقش چطوری توی مرورگر کاربر ذخیره میشه؟ آیا رمز گزاری شده هست یا نه به صورت Plain Text ذخیره میشه؟ اگه رمزگزاری میشه با چه الگوریتمی این کار انجام میشه و آیا میشه اونو Decrypt کرد؟
با تشکر

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۲/۱۸ ۲۱:۴۷

- این کوکی رو اگر خواستید مشاهده کنید از افزونه Cookies manager استفاده کنید. چنین نام و محتوای رمزنگاری شده‌ای داره:

البته این نام پیش‌فرض است. اگر نیاز به تعیین نام دیگری بود به این صورت می‌شود عمل کرد:

```
<roleManager
    enabled="true"
    cacheRolesInCookie="true"
    defaultProvider="..."
    cookieName=".ASPXROLES"
    cookiePath="/"
    cookieTimeout="30"
    cookieRequireSSL="false"
    cookieSlidingExpiration="true"
    createPersistentCookie="false"
    cookieProtection="All">
    <providers>
      <!-- .... -->
    </providers>
</roleManager>
```

- این کوکی فقط پس از اولین فراخوانی متدهای IsInRole یا GetRoles تولید می‌شود و نه پیش از آن.
- اگر از دات نت 4 و نیم استفاده می‌کنید، برای حالت کش نشدن این نقش‌ها اخیرا یک patch ارائه شده : (^). مورد چهارم آن.

```
Assume that you set the value of the cachedRolesInCookie property to true in your web application.
Your application serializes the RolePrincipal object into the cookie, and then sends it in response.
In this situation, the role cookie value is empty in the application's following request.
```

نویسنده: سعید
تاریخ: ۱۳۹۲/۰۱/۰۶ ۱:۴

سلام

اگر کاربر بتونه متن Role رو ببینه و همچنین تغییرش بده... آیا این مشکل پیش نمیاد که شخصی بیاد و متن کوکی مدیریت رو

برداره و زمانی که لوگین کرد متن کوکی خودش رو تغییر بده. اینجوری چون همه رول‌ها از کوکی خونده میشه پس می‌تونه به این شیوه به رول‌ها کلک بزنه؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۱/۰۶ ۱:۶

این اطلاعات رمزنگاری شده هستند. کلید آن‌ها هم در سرور قرار دارد و به کلاینت ارسال نمی‌شود.

نویسنده: ali
تاریخ: ۱۳۹۲/۰۲/۲۹ ۱۰:۵۹

دوست عزیز چند ماه گذشته امکانش هست پروژه ای که برای سطح دسترسی نوشتید به صورت سورس باز بگذارید؟
اگر هم به صورت مطلب جدید بگذارید خیلی ممنون میشم.تا بتونیم بازخوردهای خوبی داشته باشیم.

ممنون

نویسنده: سید مهران موسوی
تاریخ: ۱۳۹۲/۰۲/۲۹ ۲۰:۶

این پروژه صرفا واسه سطح دسترسی نبود یک بخش از CMF شخصی بنده بود با نام نئوکس، فعلا قصد انتشار عمومی اون رو ندارم چون کارای شخصی خودم رو در حال حاضر دارم با اون انجام میدم و دلیل دیگه اینکه وقت پشتیبانی ازش رو در یک محیط عمومی بعد از انتشار مثلا در Gtihub فعلا متاسفانه ندارم.
فعلا بازده عالی ای داشته این CMF از همه جهات ، اگر تصمیم به انتشار عمومیش گرفتم حتما داخل همین سایت اطلاع رسانی میشه .
برای مثال یکی از کارایی که باهاش انجام شده سایت شرکت خود بنده هست ، اینجا
یا سایت شرکت رایان صنعت (سهامی خاص) با رتبه‌ی یک در چند کلمه‌ی کلیدی به لطف SEO Engine این CMF در اینجا

نویسنده: شاهین
تاریخ: ۱۳۹۲/۰۳/۱۲ ۲۳:۵

سلام
من وقتی CustomRoleProvider رو از RoleProvider ارث بری میکنم، به جز دو تابع GetRolesForUser و IsUserInRole که override کردیم واسه بقیه توابع کلاس RoleProvider خطای زیر رو میده:

```
Error2'student_mvc.Security.CustomRoleProvider' does not implement inherited abstract member
'System.Web.Security.RoleProvider.GetAllRoles()
```

یا مثلا

```
Error3'student_mvc.Security.CustomRoleProvider' does not implement inherited abstract member
'System.Web.Security.RoleProvider.GetUsersInRole(string)
```

مشکل و راه حل چیست.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۳/۱۲ ۲۳:۱۳

اون‌ها رو باید دستی اضافه کنی. یک نمونه‌اش در قسمت 18 سری MVC مطرح شده.

نویسنده: وحید

تاریخ: ۱۳۹۲/۱۰/۱۶ ۱۳:۵۴

بنظر شما اگر بجای احراز هویت پیش فرض در MVC یا همان FormAuthentication پس از لاگین کاربر اطلاعات موجودیت آن را در session ای بریزیم و بعد در جاهای مختلف برنامه از آن استفاده کنیم مثلا هر جا که لازم بود type کاربر را بدست آورد که مثلا مشتری یا نماینده یا مدیر است دیگر لازم نیست متد GetUserById را صدازد و بار بر روی سیستم را افزایش داد آیا روش بنده صحیح است از لحاظ performance و امنیت؟

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۱۰/۱۶ ۱۴:۳۴

اتفاقا سربار سشن بیشتر است از Forms Authentication. اطلاعات سشن به صورت پیش فرض در حافظه سرور ذخیره می‌شود اما اطلاعات تیکت Forms Authentication در یک کوکی رمزنگاری شده در مرورگر کاربر ذخیره خواهد شد. در این حالت مصرف حافظه کمتری را در سمت سرور خواهید داشت و ضمنا با ری‌سایکل شدن برنامه در IIS، تمام لاگین‌های کاربران از دست نخواهد رفت (اصطلاحا تمام سشن‌های لاگین نمی‌پرند) و مجبور به لاگین مجدد نخواهند شد؛ چون کوکی رمزنگاری شده جهت اعتبارسنجی بعدی، در مرورگر کاربر ذخیره شده و نه در حافظه سرور.

نویسنده: وحید
تاریخ: ۱۳۹۲/۱۰/۱۶ ۱۷:۰

باتشکر از شما .
پس شما میگویید هر زمان که ما خواستیم ببینیم کاربری نمایندست یا مدیر در اکشنی که تعداد آن اکشن‌ها هم کم نیست می‌بایست GetUserById را استفاده نمود؟

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۱۰/۱۶ ۱۷:۲۵

دو مطلب هست. یکی اینکه اگر نیاز به ID کاربر داشتید، این مورد همیشه در User.Identity.Name موجود هست و زمانیکه در حین لاگین انتساب داده شد، در کوکی رمزنگاری شده آن قرار می‌گیره. یعنی واکشی از دیتابیس نداره. ضمنا RoleProvider استفاده شده هم زمانیکه در وب کانفیگ تنظیم میشه، مقدار cacherolesincookie=true داره (در مقاله هست). یعنی این هم کش میشه و هربار از دیتابیس واکشی نمیشه.

نویسنده: وحید
تاریخ: ۱۳۹۲/۱۰/۱۶ ۲۳:۲۷

ممنون از شما آیا با cacherolesincookie=true فقط roleها کش میشوند ؟ مثلا زمانی که کاربری لاگین کرده و در هدر پیغام خوش آمدید کاربر گرامی آقای...می شود این نام و نام خانوادگی را باید هر دفعه بازای هر request از دیتابیس بیاریم یا میتونیم از حالت شبیه این cacherolesincookie=true استفاده کرد با تشکر

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۱۰/۱۶ ۲۳:۳۸

برای این حالت‌های خاص می‌تونید از CacheManager قسمت 19 سری MVC استفاده کنید. برای بررسی حالت Forms Authentication هم قسمت 18 سری MVC به مباحث مقدماتی اون پرداخته.

نویسنده: رضا گرمارودی
تاریخ: ۱۳۹۲/۱۱/۲۰ ۹:۲۲

سلام
من کل مطالب Mvc 18 و بخش کوکی‌ها را مطالعه کردم اما به دلیل نظرات مختلف متوجه نشدم راه اصولی چیه؟
برای هر کاربر یکسری اطلاعات وجود دارد که میشه در کوکی‌ها ذخیره کرد مثل نام کاربری یا هر چیز دیگه ای که حتی اگر کاربر

آنها را تغییر بده مهم نیستند و صرفا جنبه نمایش در صفحات را دارند اما یکسری اطلاعات هست که خیلی مهمه هستند مثل این که این کاربر مدیر هست یا خیر. این اطلاعات یا باید هر بار که نیاز هست از دیتابیس یا هر منبع دیگه ای واکشی بشه ویا در جایی ذخیره بشه که هر وقت خواستیم به اون دسترسی داشته باشیم.
پیش‌تر این نوع اطلاعات و در Session ذخیره می‌کردیم که از دید کاربر به دور بود ، اما برای پردازش موازی و انجام چک لیست تهیه شده بهتره که از Sessionها استفاده نشه،
خب حالا ذخیره این اطلاعات در کوکی‌ها درسته؟ حتی اگر کد بشن باز خطرناکه ، چرا که یک کاربر اگر بتونه کوکی که مدیر بودن یا نبودن کاربر را مشخص می‌کند را تغییر بده می‌تونه به همه بخش‌ها دسترسی پیدا کنه!
از طرفی صرفا مدیر بودن یا نبودن یک کاربر مطرح نیست ، اطلاعات زیادی هستند که مهم هستند و ذخیره اون‌ها در Session میتونه منابع سرور و به خودش مشغول کنه!
روش درست چیه؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۱/۲۰ ۹:۲۸

- راه حل‌های مبتنی بر سشن، از Classics ASP دهه نود به ارث رسیده‌اند. عملا با پیشرفت‌هایی که حاصل شده نیازی به بسیاری از آن‌ها نیست. مصرف حافظه بالایی دارند و همچنین با ری‌استارت شدن برنامه در سرور، تمام سشن‌ها از بین خواهند رفت. این مشکلات در Forms Authentication وجود ندارند.
- قدمت Forms Authentication به ASP.NET *1.x* بر می‌گردد. می‌توانید در این مورد در سایت‌های دیگر نیز بیشتر تحقیق کنید که آیا مشکل حادی از سال 2001 تا الان گزارش شده یا خیر.
- کلید رمزنگاری این کوکی‌ها در سمت سرور قرار دارد و تنها یک راه برای دسترسی به آن‌ها هست؛ دسترسی به سرور. در این حالت عملا کل سیستم مورد حمله قرار گرفته و یک کوکی شاید اهمیت خاصی نداشته باشد.
- ضمنا طول مدت زمان معتبر بودن اطلاعات Forms Authentication و دائمی بودن و نبودن کوکی‌های آن قابل تنظیم است (بحث شده در مطلب فوق).

نویسنده: رضا گرمارودی
تاریخ: ۱۳۹۲/۱۱/۲۰ ۱۰:۴۵

ممنون از پاسختون اما Authenticate در مند SetAuthCookie تنها نام یوزر و ذخیره می‌کند، برای ذخیره سایر اطلاعات چه کار باید کرد؟

اگر باید از متدهای HttpContext استفاده بشود جهت رمزنگاری کوکی‌ها چه الگوریتمی به اندازه Authentication مطمئن است؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۱/۲۰ ۱۱:۲۸

- سایر اطلاعات مانند نقش‌های یک کاربر (مدیر است یا نویسنده مثلا) در مطلب جاری به صورت یک role provider مکمل پیاده سازی شده‌است.
+ بیشتر از این نیازی نیست اطلاعاتی را در کوکی‌ها یا جای دیگری ذخیره کنید. اطلاعاتی فراتر از این (مانند صفحه پروفایل یک شخص در سایت)، بسیار شخصی بوده و هر زمانیکه نیاز باشد، باید مستقلا از دیتابیس واکشی شوند.

نویسنده: مهدی سعیدی فر
تاریخ: ۱۳۹۲/۱۱/۲۰ ۱۳:۴۳

بنده در حال توسعه‌ی یه سیستم ساده Membership متن باز هستم که توانایی تعریف گروه و تعیین سطوح دسترسی به صورت پویا را دارد. کار خیلی زیادی ازش باقی نمانده و در صورت اتمام حتما به اشتراک می‌گذارم.

نویسنده: رضا گرمارودی
تاریخ: ۱۳۹۲/۱۱/۲۳ ۱۲:۱۸

سلام؛ من هم مشکل ایشون دارم و رول‌ها کش نمیشه و هر بار متد GetRolesForUser اجرا میشود. اون هم نه یکبار بلکه به تعداد زیاد و بار زیادی به دیتابیس وارد می‌کنه. لینک شما هم پیغام زیر و میدهد:

An update is available for the .NET Framework 4.5 in Windows 7 SP1, Windows Server 2008 R2 SP1, Windows Server 2008 SP2, and Windows Vista SP2: January 2013

اما ویندوز من 8 هست و پیغام سایت برای ویندوزهای 2008و7وvista هستش. با گشتن هم دیدم بعضی‌ها خودشون متد را دستی کش کردن

```
public override string[] GetRolesForUser(string username)
        {
            var cacheKey = string.Format("{0}:{1}", username, ApplicationName);
            var cache = HttpContext.Current.Cache;
            var roles = cache[cacheKey] as string[];
            if (null == roles)
            {
                using (var db = new Uas3Context())
                {
                    var u = new EfStudentService(db);
                    roles=u.GetUserRoles(username);
                    cache.Insert(cacheKey, roles, null, Cache.NoAbsoluteExpiration,
Cache.NoSlidingExpiration);
                }
            }
            return roles;
        }
```

اما این متد و من کش کردم باقی متدها چی؟ کل متدها را همین طوری کش کنم؟ این کد هم به نظرم ناقصه . چون با تغییر دیتابیس به روز نمیشه و صرفا درصورت وجود کش اطلاعات می‌خونه. اصلا چرا در دات نت 4.5 این مشکل هست و چرا بر روی ویندوز 8 این پیغام داده میشود؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۱/۲۳ ۱۲:۴۴

- مشکل طولانی بودن حاصل BinaryFormatter serialization برطرف شده (نزدیک به *یکسال قبل* در January 2013). این مشکل سبب می‌شده تا حاصل RolePrincipal.ToEncryptedTicket بسیار طولانی شده و بیشتر از حد مجاز اندازه قابل ذخیره سازی در یک کوکی شود.
- نسخه‌ی ویندوز 8 و ویندوز سرور 2012 آن از اینجا قابل دریافت است؛ نسخه‌ی ویندوز 7 و ویندوز سرور 2008 از اینجا .
+ آپدیت ویندوز را روشن کنید تا آخرین به روز رسانی‌ها و نگارش‌های دات نت نصب شده را به صورت خودکار دریافت کنید.
+ بررسی کنید آیا ویژگی rolemanager cacheRolesInCookie=true در وب کانفیگ تنظیم شده یا خیر. اگر خیر، چیزی کش نخواهد شد.

نویسنده: ارکا
تاریخ: ۱۳۹۲/۱۲/۱۵ ۲۱:۳۷

سلام،
الان دو، سه روزه که میخوام شخصی سازی Membership رو یاد بگیرم، اما واقعا گیج شدم،
میدونم که باید از کلاس‌های MembershipUser,Membership Provider و RoleProvider ارث بری بکنم و توابع مورد نظرم رو Override، اما اینکه دیتابیس چجوری باید باشه، توابعی که نوع خروجیش (تعداد و نوع پارامترهاش) اونی نیس که لازم دارم، چجوری باید پیاده سازی کنم.

امکان راهنمایی وجود داره؟
- من Web Form کار میکنم.
ممنون

نویسنده: محمد شهریاری
تاریخ: ۱۳۹۲/۱۲/۱۶ ۸:۳۰

مبحث مربوط به مدیریت کاربران و گروه کاربران و پروفایل از طریق سه کلاس پایه System.Web.Security.MembershipProvider
System.Web.Security.RoleProvider
System.Web.Profile.ProfileProvider
همونطور که از کلاس‌ها مشخص هست به صورت Provider مدیریت می‌شود. این به این معنی است که اگر شما بخواید Provider خودتون با هر Datasource جدید یا منطق جدید داشته باشید مجاز هستید نسخه جدید از این Providerها رو پیاده سازی و با تنظیمات مربوطه در Web.Config استفاده کنید . در این قسمت شما مجاز به تغییر ورودی یا خروجی متدهای abstract classهای گفته شده نیستید .

پیاده سازی‌های مختلفی از این سه کلاس به صورت ODBC , EF , .. وجود داره که لینک زیر مربوط به پیاده سازی odbc است .
پیاده سازی MembershipProvider
پیاده سازی RoleProvider
پیاده سازی ProfileProvider

موفق باشید

نویسنده: ارکا
تاریخ: ۱۳۹۲/۱۲/۱۶ ۹:۲۸

ممنون بابت راهنمایی،
تقریبا تو پیاده سازی مجدد مشکلی ندارم، بیشتر مشکلم با امضای توابع هستش (نوع خروجی و پارامترهاش).

یعنی راهی نیست که بتونم توابعی با نوع خروجی و پارامترهای دلخواه داشته باشم و در عین حال از سیستم ممبرشیب هم استفاده کنم؟

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۱۲/۱۶ ۱۰:۵۳

وجود اینترفیس و قرارداد در کدها برای این است که هم طراح و هم استفاده کننده تکلیف خودشان را بدانند. اگر قرار باشد این اینترفیس‌ها به میل شما هر روز تغییر کنند که دیگر به آن قرارداد گفته نمی‌شود. ضمنا برای پاسخگویی به این نوع سؤالات تمامی ناپذیر، سیستم جدیدی رو طراحی کردند به نام ASP.NET Identity . این سیستم از بنیان سورس باز هست. در اینجا شما هر طور که دوست داشتید، تمام اینترفیس‌ها و کدها رو تغییر بدید. سورس رو که دارید. وابسته هم نیست به بانک اطلاعاتی خاصی.

نویسنده: امیر ارسلان
تاریخ: ۱۳۹۳/۰۵/۱۰ ۱۲:۴۲

ممنون از تلاش و سایت خوبتون. اما کاش یه نمونه میزاشتین من خیلی تلاش کردم اما نتونستم جواب درستی بگیرم. اگه امکان داره یه نمونه بزازین ممنون میشم.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۵/۱۰ ۱۲:۴۶

- از پروژه سورس باز « Iris Membership برای احراز هویت کاربران در ASP.NET MVC به صورت پویا » ایده بگیرید.
- همچنین در پروژه‌ی سورس باز « سیستم مدیریت محتوای IRIS » نیز پیاده سازی کاملی از این مبحث وجود دارد.

نویسنده: امیر ارسلان
تاریخ: ۱۳۹۳/۰۵/۱۰ ۲۰:۵۷

ممنون. لینک دوم من نتونستم روی لوکال نصب کنم اما Iris membership دیدم کمی درک کردم اما نمیدونم database چطور به پروژه Add کنم.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۵/۱۰ ۲۳:۱۶

پیشنیاز درک این پروژه‌ها EF Code first است.

نویسنده: فایز مهربانی
تاریخ: ۱۳۹۳/۰۷/۲۶ ۱۳:۴۲

سلام وقت بخیر...
ببخشید در مورد Claim Authorization چه مطالبی وجود داره ؟ رفرنس فارسی وجود داره ؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۷/۲۶ ۱۴:۴۴

« مروری بر Claim »

عنوان: Globalization در ASP.NET MVC
نویسنده: یوسف نژاد
تاریخ: ۱۳۹۱/۱۰/۲۱ ۱۶:۲۵
آدرس: www.dotnettips.info
گروه‌ها: ASP.Net MVC, .NET, Localization, Globalization, Internationalization, Resource, .resx, MVC

اگر بازار هدف یک محصول شامل چندین کشور، منطقه یا زبان مختلف باشد، طراحی و پیاده سازی آن برای پشتیبانی از ویژگی‌های چندزبانه یک فاکتور مهم به حساب می‌آید. یکی از بهترین روشهای پیاده سازی این ویژگی در دات نت استفاده از فایلهای Resource است. درواقع هدف اصلی استفاده از فایلهای Resource نیز Globalization است. Globalization برابر است با Internationalization + Localization که به اختصار به آن g11n میگویند. در تعریف، Internationalization (یا به اختصار i18n) به فرایند طراحی یک محصول برای پشتیبانی از فرهنگ(culture)ها و زبانهای مختلف و Localization (یا L10n) یا بومی‌سازی به شخصی‌سازی یک برنامه برای یک فرهنگ یا زبان خاص گفته میشود. (اطلاعات بیشتر در اینجا).
استفاده از این فایلها محدود به پیاده سازی ویژگی چندزبانه نیست. شما میتوانید از این فایلها برای نگهداری تمام رشته‌های موردنیاز خود استفاده کنید. نکته دیگری که باید بدان اشاره کرد این است که تقریبا تمامی منابع مورد استفاده در یک محصول را میتوان درون این فایلها ذخیره کرد. این منابع در حالت کلی شامل موارد زیر است:
- انواع رشته‌های مورد استفاده در برنامه چون لیبل‌ها و پیغام‌ها و یا مسیرها (مثلا نشانی تصاویر یا نام کنترلرها و اکشنها) و یا حتی برخی تنظیمات ویژه برنامه (که نمیخواهیم براحتی قابل نمایش یا تغییر باشد و یا اینکه بخواهیم با تغییر زبان تغییر کنند مثل direction و امثال آن)
- تصاویر و آیکونها و یا فایلهای صوتی و انواع دیگر فایل ها
- و ...
نحوه بهره برداری از فایلهای Resource در دات نت، پیاده سازی نسبتا آسانی را در اختیار برنامه نویس قرار میدهد. برای استفاده از این فایلها نیز روشهای متنوعی وجود دارد که در مطلب جاری به چگونگی استفاده از آنها در پروژه‌های ASP.NET MVC پرداخته میشود.

Globalization در دات نت

فرمت نام یک culture دات نت (که در کلاس CultureInfo پیاده شده است) بر اساس استاندارد RFC 4646 (^ و ^) است. (در اینجا اطلاعاتی راجع به RFC یا Request for Comments آورده شده است). در این استاندارد نام یک فرهنگ (کالچر) ترکیبی از نام زبان به همراه نام کشور یا منطقه مربوطه است. نام زبان برپایه استاندارد ISO 639 که یک عبارت دوحرفی با حروف کوچک برای معرفی زبان است مثل fa برای فارسی و en برای انگلیسی و نام کشور یا منطقه نیز برپایه استاندارد ISO 3166 که به عبارت دوحرفی با حروف بزرگ برای معرفی یک کشور یا یک منطقه است مثل IR برای ایران یا US برای آمریکاست. برای نمونه میتوان به fa-IR برای زبان فارسی کشور ایران و یا en-US برای زبان انگلیسی آمریکایی اشاره کرد. البته در این روش نامگذاری یکی دو مورد استثنا هم وجود دارد (اطلاعات کامل کلیه زبانها: National Language Support (NLS) API Reference). یک فرهنگ خنثی (Neutral Culture) نیز تنها با استفاده از دو حرف نام زبان و بدون نام کشور یا منطقه معرفی میشود. مثل fa برای فارسی یا de برای آلمانی. در این بخش نیز دو استثنا وجود دارد (^).
در دات نت دو نوع culture وجود دارد: **Culture** و **UICulture** . هر دوی این مقادیر در هر Thread مقداری منحصربه فرد دارند. مقدار Culture بر روی توابع وابسته به فرهنگ (مثل فرمت رشته‌های تاریخ و اعداد و پول) تاثیر میگذارد. اما مقدار UICulture تعیین میکند که سیستم مدیریت منابع دات نت (Resource Manager) از کدام فایل Resource برای بارگذاری داده‌ها استفاده کند. درواقع در دات نت با استفاده از پراپرتی‌های موجود در کلاس استاتیک Thread برای ثرد جاری (که عبارتند از CurrentCulture و CurrentUICulture) برای فرمت کردن و یا انتخاب Resource مناسب تصمیم گیری میشود. برای تعیین کالچر جاری به صورت دستی میتوان بصورت زیر عمل کرد:

```
Thread.CurrentThread.CurrentUICulture = new CultureInfo("fa-IR");
Thread.CurrentThread.CurrentUICulture = CultureInfo.CreateSpecificCulture("fa-IR");
```

دراینجا باید اشاره کنم که کار انتخاب Resource مناسب با توجه به کالچر ثرد جاری توسط ResourceProviderFactory پیشفرض دات نت انجام میشود. در مطالب بعدی به نحوه تعریف یک پرووایدر شخصی سازی شده هم خواهم پرداخت.

پشتیبانی از زبانهای مختلف در MVC

برای استفاده از ویژگی چندزبانه در MVC دو روش کلی وجود دارد.

1. استفاده از فایلهای Resource برای تمامی رشته‌های موجود
2. استفاده از Viewهای مختلف برای هر زبان

البته روش سومی هم که از ترکیب این دو روش استفاده میکند نیز وجود دارد. انتخاب روش مناسب کمی به سلیقه‌ها و عادات برنامه نویسی بستگی دارد. اگر فکر میکنید که استفاده از ویوهای مختلف به دلیل جداسازی مفاهیم درگیر در کالچرها (مثل جانمایی اجزای مختلف ویوها یا بحث Direction) باعث مدیریت بهتر و کاهش هزینه‌های پشتیبانی میشود بهتر است از روش دوم یا ترکیبی از این دو روش استفاده کنید. خودم به شخصه سعی میکنم از روش اول استفاده کنم. چون معتقدم استفاده از ویوهای مختلف باعث افزایش بیش از اندازه حجم کار میشود. اما در برخی موارد استفاده از روش دوم یا ترکیبی از دو روش میتواند بهتر باشد.

تولید فایلهای Resource

بهترین مکان برای نگهداری فایلهای Resource در یک پروژه جداگانه است. در پروژه‌های از نوع وب‌سایت پوشه‌هایی با نام App_GlobalResources یا App_LocalResources وجود دارد که میتوان از آنها برای نگهداری و مدیریت این نوع فایلها استفاده کرد. اما همانطور که در اینجا توضیح داده شده است این روش مناسب نیست. بنابراین ابتدا یک پروژه مخصوص نگهداری فایلهای Resource ایجاد کنید و سپس اقدام به تهیه این فایلها نمایید. سعی کنید که عنوان این پروژه به صورت زیر باشد. برای کسب اطلاعات بیشتر درباره نحوه نامگذاری اشیای مختلف در دات نت به این مطلب رجوع کنید.

SolutionName>.Resources>

برای افزودن فایلهای Resource به این پروژه ابتدا برای انتخاب زبان پیش فرض محصول خود تصمیم بگیرید. پیشنهاد میکنم که از زبان انگلیسی (en-US) برای اینکار استفاده کنید. ابتدا یک فایل Resource (با پسوند resx.) مثلا با نام Texts.resx به این پروژه اضافه کنید. با افزودن این فایل به پروژه، ویژوال استودیو به صورت خودکار یک فایل cs. حاوی کلاس متناظر با این فایل را به پروژه اضافه میکند. این کار توسط ابزار توکاری به نام ResXFileCodeGenerator انجام میشود. اگر به پراپرتی‌های این فایل resx. رجوع کنید میتوانید این عنوان را در پراپرتی Custom Tool ببینید. البته ابزار دیگری برای تولید این کلاسها نیز وجود دارد. این ابزارهای توکار برای سطوح دسترسی مخلف استفاده میشوند. ابزار پیش فرض در ویژوال استودیو یعنی همان ResXFileCodeGenerator، این کلاسها را با دسترسی internal تولید میکند که مناسب کار ما نیست. ابزار دیگری که برای اینکار درون ویژوال استودیو وجود دارد PublicResXFileCodeGenerator است و همانطور که از نامش پیداست از سطح دسترسی public استفاده میکند. برای تغییر این ابزار کافی است تا عنوان آن را دقیقا در پراپرتی Custom Tool تایپ کنید.

Solution Explorer
Properties
References
Controls
Global
Configs.fa.resx
Configs.resx
Exceptions.fa.resx
Exceptions.resx
Paths.fa.resx
Paths.resx
Texts.fa.resx
Texts.resx
ViewModels
Views
excludes.txt
Solution Explorer
Team Explorer
Properties
Texts.resx File Properties
Build Action
Embedded Resource
Copy to Output Directory
Do not copy
Custom Tool
PublicResXFileCodeGenerator
Custom Tool Namespace
File Name
Texts.resx
Full Path
[redacted]
Build Action
How the file relates to the build and deployment processes.

نکته: درباره پراپرتی مهم Build Action این فایلها در مطالب بعدی بیشتر بحث میشود.
برای تعیین سطح دسترسی Resource موردنظر به روشی دیگر، میتوانید فایل Resource را باز کرده و Access Modifier آن را به Public تغییر دهید.

سپس برای پشتیبانی از زبانی دیگر، یک فایل دیگر Resource به پروژه اضافه کنید. نام این فایل باید همنام فایل اصلی به همراه نام کالچر موردنظر باشد. مثلا برای زبان فارسی عنوان فایل باید Texts.fa-IR.resx یا به صورت ساده‌تر برای کالچر خنثی (بدون نام کشور) Texts.fa.resx باشد. دقت کنید اگر نام فایل را در همان پنجره افزودن فایل وارد کنید ویژوال استودیو این همنامی را به صورت هوشمند تشخیص داده و تغییراتی را در پراپرتی‌های پیش فرض فایل Resource ایجاد میکند.
نکته: این هوشمندی مرتبه نسبتا بالایی دارد. بدین صورت که تنها درصورتیکه عبارت بعد از نام فایل اصلی Resource (رشته بعد از نقطه مثلا fa در اینجا) متعلق به یک کالچر معتبر باشد این تغییرات اعمال خواهد شد.
مهمترین این تغییرات این است که ابزاری را برای پراپرتی Custom Tool این فایلها انتخاب نمیکند! اگر به پراپرتی فایل Texts.fa.resx مراجعه کنید این مورد کاملا مشخص است. در نتیجه دیگر فایل cs. حاوی کلاسی جداگانه برای این فایل ساخته نمیشود. همچنین اگر فایل Resource جدید را باز کنید میبینید که برای Access Modifier آن گزینه No Code Generation انتخاب شده است.

در ادامه شروع به افزودن عناوین موردنظر در این دو فایل کنید. در اولی (بدون نام زبان) رشته‌های مربوط به زبان انگلیسی و در دومی رشته‌های مربوط به زبان فارسی را وارد کنید. سپس در هرجایی که یک لیبل یا یک رشته برای نمایش وجود دارد از این کلیدهای Resource استفاده کنید مثل:

SolutionName>.Resources.Texts.Save>

SolutionName>.Resources.Texts.Cancel>

استفاده از Resource در ویومدل ها

دو خاصیت معروفی که در ویومدلها استفاده میشوند عبارتند از: DisplayName و Required. پشتیبانی از کلیدهای Resource به صورت توکار در خاصیت Required وجود دارد. برای استفاده از آنها باید به صورت زیر عمل کرد:

```
[Required(ErrorMessageResourceName = "ResourceKeyName", ErrorMessageResourceType =
typeof(<SolutionName>.Resources.<ResourceClassName>))]
```

در کد بالا باید از نام فایل Resource اصلی (فایل اول که بدون نام کالچر بوده و به عنوان منبع پیشفرض به همراه یک فایل cs. حاوی کلاس مربوطه نیز هست) برای معرفی ErrorMessageResourceType استفاده کرد. چون ابزار توکار ویژوال استودیو از نام این فایل برای تولید کلاس مربوطه استفاده میکند.
متاسفانه خاصیت DisplayName که در فضای نام System.ComponentModel (در فایل System.dll) قرار دارد قابلیت استفاده از کلیدهای Resource را به صورت توکار ندارد. در دات نت 4 خاصیت دیگری در فضای نام System.ComponentModel.DataAnnotations به نام Display (در فایل System.ComponentModel.DataAnnotations.dll) وجود دارد که این امکان را به صورت توکار دارد. اما قابلیت استفاده از این خاصیت تنها در MVC 3 وجود دارد. برای نسخه‌های قدیمیتر

MVC امکان استفاده از این خاصیت حتی اگر نسخه فریمورک هدف 4 باشد وجود ندارد، چون هسته این نسخه‌های قدیمی امکان استفاده از ویژگی‌های جدید فریمورک با نسخه بالاتر را ندارد. برای رفع این مشکل میتوان کلاس خاصیت DisplayName را برای استفاده از خاصیت Display به صورت زیر توسعه داد:

```
public class LocalizationDisplayNameAttribute : DisplayNameAttribute
  {
    private readonly DisplayAttribute _display;
    public LocalizationDisplayNameAttribute(string resourceName, Type resourceType)
    {
      _display = new DisplayAttribute { ResourceType = resourceType, Name = resourceName };
    }
    public override string DisplayName
    {
      get
      {
        try
        {
          return _display.GetName();
        }
        catch (Exception)
        {
          return _display.Name;
        }
      }
    }
  }
```

در این کلاس با ترکیب دو خاصیت نامبرده امکان استفاده از کلیدهای Resource فراهم شده است. در پیاده سازی این کلاس فرض شده است که نسخه فریمورک هدف حداقل برابر 4 است. اگر از نسخه‌های پایین‌تر استفاده میکنید در پیاده سازی این کلاس باید کاملا به صورت دستی کلید موردنظر را از Resource معرفی شده بدست آورید. مثلا به صورت زیر:

```
public class LocalizationDisplayNameAttribute : DisplayNameAttribute
{
    private readonly PropertyInfo nameProperty;
    public LocalizationDisplayNameAttribute(string displayNameKey, Type resourceType = null)
        : base(displayNameKey)
    {
        if (resourceType != null)
            nameProperty = resourceType.GetProperty(base.DisplayName, BindingFlags.Static |
BindingFlags.Public);
    }
    public override string DisplayName
    {
        get
        {
            if (nameProperty == null) base.DisplayName;
            return (string)nameProperty.GetValue(nameProperty.DeclaringType, null);
        }
    }
}
```

برای استفاده از این خاصیت جدید میتوان به صورت زیر عمل کرد:

```
[LocalizationDisplayName("ResourceKeyName", typeof(<SolutionName>.Resources.<ResourceClassName>))]
```

البته بیشتر خواص متداول در ویومدلها از ویژگی موردبحث پشتیبانی میکنند.

نکته: به کار گیری این روش ممکن است در پروژه‌های بزرگ کمی گیج کننده و دردسرساز بوده و باعث پیچیدگی بی‌مورد کد و نیز افزایش بیش از حد حجم کدنویسی شود. در مقاله آقای فیل هک (Model Metadata and Validation Localization using Conventions) روش بهتر و تمیزتری برای مدیریت پیامهای این خاصیت‌ها آورده شده است.

پشتیبانی از ویژگی چند زبانه

مرحله بعدی برای چندزبانه کردن پروژه‌های MVC تغییراتی است که برای مدیریت Culture جاری برنامه باید پیاده شوند. برای

اینکار باید خاصیت CurrentUICulture در ثرد جاری کنترل و مدیریت شود. یکی از مکانهایی که برای نگهداری زبان جاری استفاده میشود کوکی است. معمولا برای اینکار از کوکی‌های دارای تاریخ انقضای طولانی استفاده میشود. میتوان از تنظیمات موجود در فایل کانفیگ برای ذخیره زبان پیش فرض سیستم نیز استفاه کرد.
روشی که معمولا برای مدیریت زبان جاری میتوان از آن استفاده کرد پیاده سازی یک کلاس پایه برای تمام کنترلرها است. کد زیر راه حل نهایی را نشان میدهد:

```
public class BaseController : Controller
  {
    private const string LanguageCookieName = "MyLanguageCookieName";
    protected override void ExecuteCore()
    {
      var cookie = HttpContext.Request.Cookies[LanguageCookieName];
      string lang;
      if (cookie != null)
      {
        lang = cookie.Value;
      }
      else
      {
        lang = ConfigurationManager.AppSettings["DefaultCulture"] ?? "fa-IR";
        var httpCookie = new HttpCookie(LanguageCookieName, lang) { Expires = DateTime.Now.AddYears(1)
};
        HttpContext.Response.SetCookie(httpCookie);
      }
      Thread.CurrentThread.CurrentUICulture = CultureInfo.CreateSpecificCulture(lang);
      base.ExecuteCore();
    }
  }
```

راه حل دیگر استفاده از یک ActionFilter است که نحوه پیاده سازی یک نمونه از آن در زیر آورده شده است:

```
public class LocalizationActionFilterAttribute : ActionFilterAttribute
  {
    private const string LanguageCookieName = "MyLanguageCookieName";
    public override void OnActionExecuting(ActionExecutingContext filterContext)
    {
      var cookie = filterContext.HttpContext.Request.Cookies[LanguageCookieName];
      string lang;
      if (cookie != null)
      {
        lang = cookie.Value;
      }
      else
      {
        lang = ConfigurationManager.AppSettings["DefaultCulture"] ?? "fa-IR";
        var httpCookie = new HttpCookie(LanguageCookieName, lang) { Expires = DateTime.Now.AddYears(1)
};
        filterContext.HttpContext.Response.SetCookie(httpCookie);
      }
      Thread.CurrentThread.CurrentUICulture = CultureInfo.CreateSpecificCulture(lang);
      base.OnActionExecuting(filterContext);
    }
  }
```

نکته مهم: تعیین زبان جاری (یعنی همان مقداردهی پراپرتی CurrentCulture ثرد جاری) در یک اکشن فیلتر بدرستی عمل نمیکند.
برای بررسی بیشتر این مسئله ابتدا به تصویر زیر که ترتیب رخدادن رویدادهای مهم در ASP.NET MVC را نشان میدهد دقت کنید:

همانطور که در تصویر فوق مشاهده میکنید رویداد OnActionExecuting که در یک اکشن فیلتر به کار میرود بعد از عملیات مدل بایندینگ رخ میدهد. بنابراین قبل از تعیین کالچر جاری، عملیات validation و یافتن متن خطاها از فایلهای Resource انجام میشود که منجر به انتخاب کلیدهای مربوط به کالچر پیشفرض سرور (و نه آنچه که کاربر تنظیم کرده) خواهد شد. بنابراین استفاده از یک اکشن فیلتر برای تعیین کالچر جاری مناسب نیست. راه حل مناسب استفاده از همان کنترلر پایه است، زیرا متد ExecuteCore قبل از تمامی این عملیات صدا زده میشود. بنابراین همیشه کالچر تنظیم شده توسط کاربر به عنوان مقدار جاری آن در ثرد ثبت میشود.

امکان تعیین/تغییر زبان توسط کاربر

برای تعیین یا تغییر زبان جاری سیستم نیز روشهای گوناگونی وجود دارد. استفاده از زبان تنظیم شده در مرورگر کاربر، استفاده از عنوان زبان در آدرس صفحات درخواستی و یا تعیین زبان توسط کاربر در تنظیمات برنامه/سایت و ذخیره آن در کوکی یا دیتابیس و مواردی از این دست روشهایی است که معمولا برای تعیین زبان جاری از آن استفاده میشود. در کدهای نمونه ای که در بخشهای قبل آورده شده است فرض شده است که زبان جاری سیستم درون یک کوکی ذخیره میشود بنابراین برای استفاده از این روش میتوان از قطعه کدی مشابه زیر (مثلا در فایل _Layout.cshtml) برای تعیین و تغییر زبان استفاه کرد:

```
<select id="langs" onchange="languageChanged()">
  <option value="fa-IR">فارسی</option>
  <option value="en-US">انگلیسی</option>
</select>
<script type="text/javascript">
  function languageChanged() {
    setCookie("MyLanguageCookieName", $('#langs').val(), 365);
    window.location.reload();
  }
  document.ready = function () {
    $('#langs').val(getCookie("MyLanguageCookieName"));
  };
  function setCookie(name, value, exdays, path) {
    var exdate = new Date();
    exdate.setDate(exdate.getDate() + exdays);
    var newValue = escape(value) + ((exdays == null) ? "" : "; expires=" + exdate.toUTCString()) +
((path == null) ? "" : "; path=" + path) ;
    document.cookie = name + "=" + newValue;
  }
```

```
  function getCookie(name) {
    var i, x, y, cookies = document.cookie.split(";");
    for (i = 0; i < cookies.length; i++) {
      x = cookies[i].substr(0, cookies[i].indexOf("="));
      y = cookies[i].substr(cookies[i].indexOf("=") + 1);
      x = x.replace(/^\s+|\s+$/g, "");
      if (x == name) {
        return unescape(y);
      }
    }
  }
</script>
```

متدهای setCookie و getCookie جاوا اسکریپتی در کد بالا از اینجا گرفته شده اند البته پس از کمی تغییر.

نکته : مطلب Cookieها بحثی نسبتا مفصل است که در جای خودش باید به صورت کامل آورده شود. اما در اینجا تنها به همین نکته اشاره کنم که عدم توجه به پراپرتی path کوکی‌ها در این مورد خاص برای خود من بسیار گیج‌کننده و دردسرساز بود.

به عنوان راهی دیگر میتوان به جای روش ساده استفاده از کوکی، تنظیماتی در اختیار کاربر قرار داد تا بتواند زبان تنظیم شده را درون یک فایل یا دیتابیس ذخیره کرد البته با درنظر گرفتن مسائل مربوط به کش کردن این تنظیمات.

راه حل بعدی میتواند استفاده از تنظیمات مرورگر کاربر برای دریافت زبان جاری تنظیم شده است. مرورگرها تنظیمات مربوط به زبان را در قسمت Accept-Languages در HTTP Header درخواست ارسالی به سمت سرور قرار میدهند. بصورت زیر:

```
GET http://www.dotnettips.info HTTP/1.1
...
Accept-Language: fa-IR,en-US;q=0.5
...
```

این هم تصویر مربوط به Fiddler آن:

نکته: پارامتر q در عبارت مشخص شده در تصویر فوق relative quality factor نام دارد و به نوعی مشخص کننده اولویت زبان مربوطه است. مقدار آن بین 0 و 1 است و مقدار پیش فرض آن 1 است. هرچه مقدار این پارامتر بیشتر باشد زبان مربوطه اولویت

بالاتری دارد. مثلا عبارت زیر را درنظر بگیرید:

```
Accept-Language: fa-IR,fa;q=0.8,en-US;q=0.5,ar-BH;q=0.3
```

در این حالت اولویت زبان fa-IR برابر 1 و fa برابر 0.8 (fa;q=0.8) است. اولویت دیگر زبانهای تنظیم شده نیز همانطور که نشان داده شده است در مراتب بعدی قرار دارند. در تنظیم نمایش داده شده برای تغییر این تنظیمات در IE میتوان همانند تصویر زیر اقدام کرد:

در تصویر بالا زبان فارسی اولویت بالاتری نسبت به انگلیسی دارد. برای اینکه سیستم g11n دات نت به صورت خودکار از این مقادیر جهت زبان ثرد جاری استفاده کند میتوان از تنظیم زیر در فایل کانفیگ استفاده کرد:

```
<system.web>
    <globalization enableClientBasedCulture="true" uiCulture="auto" culture="auto"></globalization>
</system.web>
```

در سمت سرور نیز برای دریافت این مقادیر تنظیم شده در مرورگر کاربر میتوان از کدهای زیر استفاه کرد. مثلا در یک اکشن فیلتر:

```
var langs = filterContext.HttpContext.Request.UserLanguages;
```

پراپرتی UserLanguages از کلاس Request حاوی آرایه‌ای از استرینگ است. این آرایه درواقع از Split کردن مقدار Accept-Languages با کاراکتر ',' بدست می‌آید. بنابراین اعضای این آرایه رشته‌ای از نام زبان به همراه پارامتر q مربوطه خواهند بود (مثل "fa;q=0.8").

راه دیگر مدیریت زبانها استفاده از عنوان زبان در مسیر درخواستی صفحات است. مثلا آدرسی شبیه به www.MySite.com/fa/Employees نشان میدهد کاربر درخواست نسخه فارسی از صفحه Employees را دارد. نحوه استفاده از این عناوین و نیز موقعیت فیزیکی این عناوین در مسیر صفحات درخواستی کاملا به سلیقه برنامه نویس و یا کارفرما بستگی دارد. روش کلی بهره برداری از این روش در تمام موارد تقریبا یکسان است.

برای پیاده سازی این روش ابتدا باید یک route جدید در فایل Global.asax.cs اضافه کرد:

```
routes.MapRoute(
    "Localization", // Route name
    "{lang}/{controller}/{action}/{id}", // URL with parameters
    new { controller = "Home", action = "Index", id = UrlParameter.Optional } // Parameter defaults
);
```

دقت کنید که این route باید قبل از تمام routeهای دیگر ثبت شود. سپس باید کلاس پایه کنترلر را به صورت زیر پیاده سازی کرد:

```
public class BaseController : Controller
{
  protected override void ExecuteCore()
  {
    var lang = RouteData.Values["lang"];
    if (lang != null && !string.IsNullOrWhiteSpace(lang.ToString()))
    {
      Thread.CurrentThread.CurrentUICulture = CultureInfo.CreateSpecificCulture(lang.ToString());
    }
    base.ExecuteCore();
  }
}
```

این کار را در یک اکشن فیلتر هم میتوان انجام داد اما با توجه به توضیحاتی که در قسمت قبل داده شد استفاده از اکشن فیلتر برای تعیین زبان جاری کار مناسبی نیست.

نکته: به دلیل آوردن عنوان زبان در مسیر درخواستها باید کنترل دقیقتری بر کلیه مسیرهای موجود داشت!

استفاده از ویوهای جداگانه برای زبانهای مختلف

برای اینکار ابتدا ساختار مناسبی را برای نگهداری از ویوهای مختلف خود درنظر بگیرید. مثلا میتوانید همانند نامگذاری فایلهای Resource از نام زبان یا کالچر به عنوان بخشی از نام فایلهای ویو استفاده کنید و تمام ویوها را در یک مسیر ذخیره کنید. همانند تصویر زیر:

البته اینکار ممکن است به مدیریت این فایلها را کمی مشکل کند چون به مرور زمان تعداد فایلهای ویو در یک فولدر زیاد خواهد شد. روش دیگری که برای نگهداری این ویوها میتوان به کار برد استفاده از فولدرهای جداگانه با عناوین زبانهای موردنظر است. مانند تصویر زیر:

روش دیگری که برای نگهداری و مدیریت بهتر ویوهای زبانهای مختلف از آن استفاده میشود به شکل زیر است:

استفاه از هرکدام از این روشها کاملا به سلیقه و راحتی مدیریت فایلها برای برنامه نویس بستگی دارد. درهر صورت پس از

انتخاب یکی از این روشها باید اپلیکشن خود را طوری تنظیم کنیم که با توجه به زبان جاری سیستم، ویوی مربوطه را جهت نمایش انتخاب کند.

مثلا برای روش اول نامگذاری ویوها میتوان از روش دستکاری متد OnActionExecuted در کلاس پایه کنترلر استفاده کرد:

```
public class BaseController : Controller
{
  protected override void OnActionExecuted(ActionExecutedContext context)
  {
    var view = context.Result as ViewResultBase;
    if (view == null) return; // not a view
    var viewName = view.ViewName;
    view.ViewName = GetGlobalizationViewName(viewName, context);
    base.OnActionExecuted(context);
  }
  private static string GetGlobalizationViewName(string viewName, ControllerContext context)
  {
    var cultureName = Thread.CurrentThread.CurrentUICulture.Name;
    if (cultureName == "en-US") return viewName; // default culture
    if (string.IsNullOrEmpty(viewName))
      return context.RouteData.Values["action"] + "." + cultureName; // "Index.fa"
    int i;
    if ((i = viewName.IndexOf('.')) > 0) // ex: Index.cshtml
      return viewName.Substring(0, i + 1) + cultureName + viewName.Substring(i); // "Index.fa.cshtml"
    return viewName + "." + cultureName; // "Index" ==> "Index.fa"
  }
}
```

همانطور که قبلا نیز شرح داده شد، چون متد ExecuteCore قبل از OnActionExecuted صدا زده میشود بنابراین از تنظیم درست مقدار کالچر در ثرد جاری اطمینان داریم.

روش دیگری که برای مدیریت انتخاب ویوهای مناسب استفاده از یک ویوانجین شخصی سازی شده است. مثلا برای روش سوم نامگذاری ویوها میتوان از کد زیر استفاده کرد:

```
public sealed class RazorGlobalizationViewEngine : RazorViewEngine
  {
    protected override IView CreatePartialView(ControllerContext controllerContext, string partialPath)
    {
      return base.CreatePartialView(controllerContext, GetGlobalizationViewPath(controllerContext,
partialPath));
    }
    protected override IView CreateView(ControllerContext controllerContext, string viewPath, string
masterPath)
    {
      return base.CreateView(controllerContext, GetGlobalizationViewPath(controllerContext, viewPath),
masterPath);
    }
    private static string GetGlobalizationViewPath(ControllerContext controllerContext, string
viewPath)
    {
      //var controllerName = controllerContext.RouteData.GetRequiredString("controller");
      var request = controllerContext.HttpContext.Request;
      var lang = request.Cookies["MyLanguageCookie"];
      if (lang != null && !string.IsNullOrEmpty(lang.Value) && lang.Value != "en-US")
      {
        var localizedViewPath = Regex.Replace(viewPath, "^~/Views/",
string.Format("~/Views/Globalization/{0}/", lang.Value));
        if (File.Exists(request.MapPath(localizedViewPath))) viewPath = localizedViewPath;
      }
      return viewPath;
    }
```

و برای ثبت این ViewEngine در فایل Global.asax.cs خواهیم داشت:

```
protected void Application_Start()
{
  ViewEngines.Engines.Clear();
  ViewEngines.Engines.Add(new RazorGlobalizationViewEngine());
}
```

محتوای یک فایل Resource

ساختار یک فایل resx. به صورت XML استاندارد است. در زیر محتوای یک نمونه فایل Resource با پسوند resx. را مشاهده میکنید:

```
<?xml version="1.0" encoding="utf-8"?>
<root>
  <!--
    Microsoft ResX Schema ...
    -->
  <xsd:schema id="root" xmlns="" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata">
   ...
  </xsd:schema>
  <resheader name="resmimetype">
    <value>text/microsoft-resx</value>
  </resheader>
  <resheader name="version">
    <value>2.0</value>
  </resheader>
  <resheader name="reader">
    <value>System.Resources.ResXResourceReader, System.Windows.Forms, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b77a5c561934e089</value>
  </resheader>
  <resheader name="writer">
    <value>System.Resources.ResXResourceWriter, System.Windows.Forms, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b77a5c561934e089</value>
  </resheader>
  <data name="RightToLeft" xml:space="preserve">
    <value>false</value>
    <comment>RightToleft is false in English!</comment>
  </data>
</root>
```

در قسمت ابتدایی تمام فایلهای resx. که توسط ویژوال استودیو تولید میشود کامنتی طولانی وجود دارد که به صورت خلاصه به شرح محتوا و ساختار یک فایل Resource میپردازد. در ادامه تگ نسبتا طولانی xsd:schema قرار دارد. از این قسمت برای معرفی ساختار داده ای فایلهای XML استفاده میشود. برای آشنایی بیشتر با XSD (یا XML Schema) به اینجا مراجعه کنید. به صورت خلاصه میتوان گفت که XSD برای تعیین ساختار داده‌ها یا تعیین نوع داده ای اطلاعات موجود در یک فایل XML به کار میرود. درواقع تگهای XSD به نوعی فایل XML ما را Strongly Typed میکند. با توجه به اطلاعات این قسمت، فایلهای resx. شامل 4 نوع گره اصلی هستند که عبارتند از: metadata و assembly و data و resheader. در تعریف هر یک از گره‌ها در این قسمت مشخصاتی چون نام زیر گره‌های قابل تعریف در هر گره و نام و نوع خاصیتهای هر یک معرفی شده است.

بخش موردنظر ما در این مطلب قسمت انتهایی این فایلهاست (تگهای resheader و data). همانطور در بالا مشاهده میکنید تگهای reheader شامل تنظیمات مربوط به فایل resx. با ساختاری ساده به صورت name/value است. یکی از این تنظیمات resmimetype فایل resource را معرفی میکند که درواقع مشخص کننده نوع محتوای (Content Type) فایل XML است(^). برای فایلهای resx این مقدار برابر text/microsoft-resx است. تنظیم بعدی نسخه مربوط به فایل resx (یا Microsoft ResX Schema) را نشان میدهد. در حال حاضر نسخه جاری (در VS 2010) برابر 2.0 است. تنظیم بعدی مربوط به کلاسهای reader و writer تعریف شده برای استفاده از این فایلهاست. به نوع این کلاسهای خواننده و نویسنده فایلهای resx. و مکان فیزیکی و فضای نام آنها دقت کنید که در مطالب بعدی از آنها برای ویرایش و بروزرسانی فایلهای resource در زمان اجرا استفاده خواهیم کرد.

در پایان نیز تگهای data که برای نگهداری داده‌ها از آنها استفاده میشود. هر گره data شامل یک خاصیت نام (name) و یک زیرگره مقدار (value) است. البته امکان تعیین یک کامنت در زیرگره comment نیز وجود دارد که اختیاری است. هر گره data میتواند شامل خاصیت type و یا mimetype نیز باشد. خاصیت type مشخص کننده نوعی است که تبدیل text/value را با استفاده از ساختار TypeConverter پشتیبانی میکند. البته اگر در نوع مشخص شده این پشتیبانی وجود نداشته باشد، داده موردنظر پس از سریالایز شدن با فرمت مشخص شده در خاصیت mimetype ذخیره میشود. این mimetype اطلاعات موردنیاز را برای کلاس خواننده این فایلها (ResXResourceReader به صورت پیشفرض) جهت چگونگی بازیابی آبجکت موردنظر فراهم میکند. مشخص کردن این دو خاصیت برای انواع رشته ای نیاز نیست. انواع mimetype قابل استفاده عبارتند از:

- application/x-microsoft.net.object.binary.base64: آبجکت موردنظر باید با استفاده از کلاس System.Runtime.Serialization.Formatters.Binary.BinaryFormatter سریالایز شده و سپس با فرمت base64 به یک رشته انکد شود (راجع به انکدینگ base64 ^ و ^).
- application/x-microsoft.net.object.soap.base64: آبجکت موردنظر باید با استفاده از کلاس

System.Runtime.Serialization.Formatters.Soap.SoapFormatter سریالایز شده و سپس با فرمت base64 به یک رشته انکد شود.

- application/x-microsoft.net.object.bytearray.base64: آبجکت ابتدا باید با استفاده از یک System.ComponentModel.TypeConverter به آرایه ای از بایت سریالایز شده و سپس با فرمت base64 به یک رشته انکد شود.

نکته : امکان جاسازی کردن (embed) فایلهای resx. در یک اسمبلی یا کامپایل مستقیم آن به یک سَتِلایت اسمبلی (ترجمه مناسبی برای satellite assembly پیدا نکردم، چیزی شبیه به اسمبلی قمری یا وابسته و از این قبیل ...) وجود ندارد. ابتدا باید این فایلهای resx. به فایلهای resources. تبدیل شوند. اینکار با استفاده از ابزار Resource File Generator (نام فایل اجرایی آن resgen.exe است) انجام میشود (^ و ^). سپس میتوان با استفاده از Assembly Linker ستلایت اسمبلی مربوطه را تولید کرد (^). کل این عملیات در ویژوال استودیو با استفاده از ابزار msbuild به صورت خودکار انجام میشود!

نحوه یافتن کلیدهای Resource در بین فایلهای مختلف Resx توسط پرووایدر پیش فرض در دات نت

عملیات ابتدا با بررسی خاصیت CurrentUICulture از ثرد جاری آغاز میشود. سپس با استفاده از عنوان استاندارد کالچر جاری، فایل مناسب Resource یافته میشود. در نهایت بهترین گزینه موجود برای کلید درخواستی از منابع موجود انتخاب میشود. مثلا اگر کالچر جاری fa-IR و کلید درخواستی از کلاس Texts باشد ابتدا جستجو برای یافتن فایل Texts.fa-IR.resx آغاز میشود و اگر فایل موردنظر یا کلید درخواستی در این فایل یافته نشد جستجو در فایل Texts.fa.resx ادامه می‌یابد. اگر باز هم یافته نشد درنهایت این عملیات جستجو در فایل resource اصلی خاتمه می‌یابد و مقدار کلید منبع پیش فرض به عنوان نتیجه برگشت داده میشود. یعنی در تمامی حالات سعی میشود تا دقیقترین و بهترین و نزدیکترین نتیجه انتخاب شود. البته درصورتیکه از یک پرووایدر شخصی سازی شده برای کار خود استفاده میکنید باید چنین الگوریتمی را جهت یافتن کلیدهای منابع خود از فایلهای Resource (یا هرمنبع دیگر مثل دیتابیس یا حتی یک وب سرویس) درنظر بگیرید.

Globalization در کلاینت (javascript gl1n)

یکی دیگر از موارد استفاده gl1n در برنامه نویسی سمت کلاینت است. با وجود استفاده گسترده از جاوا اسکریپت در برنامه نویسی سمت کلاینت در وب اپلیکیشنها، متاسفانه تا همین اواخر عملا ابزار یا کتابخانه مناسبی برای مدیریت gl1n در این زمینه وجود نداشته است. یکی از اولین کتابخانه‌های تولید شده در این زمینه کتابخانه jQuery Globalization است که توسط مایکروسافت توسعه داده شده است (برای آشنایی بیشتر با این کتابخانه به ^ و ^ مراجعه کنید). این کتابخانه بعدا تغییر نام داده و اکنون با عنوان Globalize شناخته میشود. Globalize یک کتابخانه کاملا مستقل است که وابستگی به هیچ کتابخانه دیگر ندارد (یعنی برای استفاده از آن نیازی به jQuery نیست). این کتابخانه حاوی کالچرهای بسیاری است که عملیات مختلفی چون فرمت و parse انواع داده‌ها را نیز در سمت کلاینت مدیریت میکند. همچنین با فراهم کردن منابعی حاوی جفتهای key/culture میتوان از مزایایی مشابه مواردی که در این مطلب بحث شد در سمت کلاینت نیز بهره برد. نشانی این کتابخانه در github اینجا است. با اینکه خود این کتابخانه ابزار کاملی است اما در بین کالچرهای موجود در فایلهای آن متاسفانه پشتیبانی کاملی از زبان فارسی نشده است. ابزار دیگری که برای اینکار وجود دارد پلاگین jquery localize است که برای بحث gl1n رشته‌ها پیاده‌سازی بهتر و کاملتری دارد.

در مطالب بعدی به مباحث تغییر مقادیر کلیدهای فایلهای resource در هنگام اجرا با استفاده از روش مستقیم تغییر محتوای فایلها و کامپایل دوباره توسط ابزار msbuild و نیز استفاده از یک ResourceProvider شخصی سازی شده به عنوان یک راه حل بهتر برای اینکار میپردازم.

در تهیه این مطلب از منابع زیر استفاده شده است: Localization in ASP.NET MVC – 3 Days Investigation, 1 Day Job

ASP.NET MVC 3 Internationalization

Localization and skinning in ASP.NET MVC 3 web applications Simple ASP.Net MVC Globalization with Graceful Fallback

Globalization, Internationalization and Localization in ASP.NET MVC 3, JavaScript and jQuery - Part 1

نظرات خوانندگان

نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۱/۱۰/۲۱ ۲۳:۵

سلام آقای یوسف نژاد
من بعد از تلاش‌های زیاد توی پروژه‌های مختلف این مطالبی که شما نوشته اید رو پیاده سازی کردم ، ولی خیلی پراکنده.
ولی حالا می‌بینم شما به زیبایی این مطالب رو کنار هم قرار دادید.
می خواستم بابت مطلب خوب و مفیدتون و همچنین وقتی که گذاشتید تشکر کنم.
ممنونم بابت زحمات شما

اگر ممکنه برچسب MVC رو هم به مطلبتون اضافه کنید.

نویسنده: یوسف نژاد
تاریخ: ۱۳۹۱/۱۰/۲۱ ۲۳:۲۴

با سلام و تشکر بابت نظر لطف شما.
البته باید بگم که همه دوستانی که اینجا به عنوان نویسنده کمک میکنند هدفشون اشتراک مطالبی هست که یاد گرفته اند تا سایر دوستان هم استفاده کنند.

برچسب MVC هم اضافه شد. با تشکر از دقت نظر شما.

نویسنده: امیرحسین جلوداری
تاریخ: ۱۳۹۱/۱۰/۲۲ ۱:۳

کاملا مشخصه که مطلب از روی تجربه‌ی کاریه و بسیار عالی جمع آوری شده ... ممنون ... به طرز عجیبی منتظر قسمت بعدم :دی

نویسنده: پندار
تاریخ: ۱۳۹۱/۱۲/۰۸ ۲۱:۳۸

گویا در MVC 4 این روش پاسخ نمیدهد. لطفا در این مورد برای MVC 4 راه حلی بدهید

نویسنده: محسن
تاریخ: ۱۳۹۱/۱۲/۰۸ ۲۳:۶

MVC 4 فقط یک سری افزونه بیشتر از MVC3 داره. مثلا razor آن بهبود پیدا کرده، فشرده سازی فایل‌های CSS به اون اضافه شده یا Web API رو به صورت یکپارچه داره. از لحاظ کار با فایل‌های منبع فرقی نکرده.

نویسنده: پندار
تاریخ: ۱۳۹۱/۱۲/۰۹ ۹:۲۱

متن نشانی زیر را مطالعه کنید

http://geekswithblogs.net/shaunxu/archive/2012/09/04/localization-in-asp.net-mvc-ndash-upgraded.aspx

نویسنده: محسن
تاریخ: ۱۳۹۱/۱۲/۰۹ ۹:۴۳

مطلبی که لینک دادی در مورد آپدیت یک helper شخصی توسعه داده شده توسط شخص ثالث است از MVC2 به MVC4. اگر کسی

از این راه حل شخصی و خاص استفاده نکرده باشه، اصول فوق فرقی نکرده.

نویسنده: صابر فتح الهی
تاریخ: ۱۳۹۱/۱۲/۱۴ ۱۶:۵

مطلب خیلی خوبی بود کلی استفاده کردیم.
مهندس کالچر زبان کردی چی میشه؟ توی لیست منابعی که دادین گیر نیاوردم

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۲/۱۴ ۱۷:۲۲

kur هست مطابق استاندارد .

نویسنده: صابر فتح الهی
تاریخ: ۱۳۹۱/۱۲/۱۷ ۲:۱۱

سلام
اما مهندس کلاس Culture Info این مقدار قبول نمی‌کنه

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۲/۱۷ ۹:۳

می‌تونید کلاس فرهنگ سفارشی را ایجاد و استفاده کنید.

نویسنده: صابر فتح الهی
تاریخ: ۱۳۹۱/۱۲/۱۷ ۱۰:۱۲

اما روش گفته شده نیاز به دسترسی مدیریت دارد که روی سرورهای اشتراکی ممکن نیست

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۲/۱۷ ۱۱:۱۹

نحوه توسعه اکثر برنامه‌ها و کتابخانه‌ها در طول زمان، بر اساس تقاضا و پیگیری مصرف کننده است. اگر بعد از بیش از 10 سال، چنین فرهنگی اضافه نشده یعنی درخواستی نداشته. مراجعه کنید به محل پیگیری این نوع مسایل .

نویسنده: صابر فتح الهی
تاریخ: ۱۳۹۱/۱۲/۱۹ ۱۰:۱۴

سلام مهندس یوسف نژاد (ابتدا ممنونم از پست خوب شما)
با پیروی از پست شما
ابتدا فایل‌های ریسورس در پروژه جاری فولدر App_GlobalResources گذاشتم و پروژه در صفحات aspx با قالب زیر به راحتی تغییر زبان داده میشد:

```
<asp:Literal ID="Literal1" Text='<%$ Resources:resource, Title %>' runat="server" />
```

اما بعدش فایل هارو توی یک پروژه کتابخانه ای جدید گذاشتم و Build Action فایل‌های ریسورس روی Embedded Resource تنظیم کردم، پروژه با موفقیت اجرا شد و در سمت سرور با کد زیر راحت به مقادیر دسترسی دارم:

```
Literal1.Text=ResourcesManager.Resource.Title;
```

اما در سمت صفحات aspx با کد قبلی به شکل زیر نمایش نمیده و خطا صادر میشه:

```
<asp:Literal ID="Literal1"  runat="server" Text='<%$ ResourcesManager.Resource:resource, Title %>'/>
```

و خطای زیر صادر میشه:

```
Parser Error
Description: An error occurred during the parsing of a resource required to service this request.
Please review the following specific parse error details and modify your source file appropriately.

Parser Error Message: The expression prefix 'ResourcesManager.Resource' was not recognized.  Please
correct the prefix or register the prefix in the <expressionBuilders> section of configuration.

Source Error:
```

مراحل این پست روی هم دنبال کردم اما بازم نشد.
چه تنظیماتی ست نکردم ؟

نویسنده: یوسف نژاد
تاریخ: ۱۳۹۲/۰۱/۳۱ ۱۲:۴۴

ببخشید یه چند وقتی فعال نبودم و پاسخ این سوال رو دیر دارم میدم.
امکان استفاده از کلیدهای Resource برای مقداردهی خواص سمت سرور کنترلها در صفحات aspx به صورت مستقیم وجود ندارد. بنابراین برای استفاده از این کلیدها همانند روش پیش فرض موجود در ASP.NET باید از یکسری ExpressionBuilder استفاده شود که کار Parse عبارت وارده برای این خواص را در سمت سرور انجام میدهد. کلاس پیش فرض برای اینکار در ASP.NET Web Form که از پیشوند Resources استفاده میکند تنها برای Resourceهای محلی (Local) موجود در فولدرهای پیش فرض (App_GlobalResources و App_LocalResources) کاربرد دارد و برای استفاده از Resourceهای موجود در منابع ریفرنس داده شده به پروژه باید از روشی مثل اونچه که خود شما لینکش رو دادین استفاده کرد.
من این روش رو استفاده کردم و پیاده سازی موفقی داشتم. نمیدونم مشکل شما چیه...

نویسنده: یوسف نژاد
تاریخ: ۱۳۹۲/۰۱/۳۱ ۱۲:۵۲

اگر مشکلی در پیاده سازی روش بالا دارین، تمام مراحلی که من طی کردم دقیقا اینجا میارم:
ابتدا کلاس ExpressionBuilder رو به صورت زیر مثلا در خود پروژه Resources اضافه میکنیم:

```
using System.Web.Compilation;
using System.CodeDom;
namespace Resources
{
  [ExpressionPrefix("MyResource")]
  public class ResourceExpressionBuilder : ExpressionBuilder
  {
    public override System.CodeDom.CodeExpression GetCodeExpression(System.Web.UI.BoundPropertyEntry
entry, object parsedData, System.Web.Compilation.ExpressionBuilderContext context)
    {
      return new CodeSnippetExpression(entry.Expression);
    }
  }
}
```

سپس تنظیمات زیر رو به Web.config اضافه میکنیم:

```
<compilation debug="true" targetFramework="4.0">
    <expressionBuilders>
        <add expressionPrefix="MyResource" type="Resources.ResourceExpressionBuilder, Resources" />
```

```
    </expressionBuilders>
</compilation>
```

درنهایت به صورت زیر میتوان از این کلاس استفاده کرد:

```
<asp:Literal ID="Literal1"  runat="server" Text="<%$ MyResource: Resources.Resource1.String2 %>" />
```

هرچند ظاهرا مقدار پیشوند معرفی شده در Attribute کلاس ResourceExpressionBuilder اهمیت چندانی ندارد! امیدوارم مشکلتون حل بشه.

نویسنده: صابر فتح الهی
تاریخ: ۱۳۹۲/۰۲/۰۱ ۲:۲۹

ممنونم از پاسخ شما
همون روش شمارو دنبال کردم پاسخ گرفتم، اشکال از خودم بود
با تشکر از شما

نویسنده: صادق نجاتی
تاریخ: ۱۳۹۲/۱۲/۲۷ ۱۲:۰

با سلام
ضمن تشکر از مطلب بسیار خوبتون
خاصیت DisplayFormat قابلیت استفاده از کلیدهای Resource را ندارد !
لطفا راهنمایی فرمایید که چطور میشه از این خاصیت برای DisplayFormat استفاده کرد؟
من می‌خوام برای تاریخ در زبانه فارسی از فرمت {yyyy-MM-dd} و در زبانه انگلیسی از {yyyy-dd-MM} استفاده کنم.
با سپاس فراوان

عنوان: **آشنایی و بررسی ابزار MiniProfiler**
نویسنده: مجتبی کاویانی
تاریخ: ۱۳۹۱/۱۱/۱۵ ۱۶:۰
آدرس: www.dotnettips.info
گروه‌ها: ASP.Net, Entity framework, ASP.Net MVC, EF Code First, Oracle, Statistics, stackoverflow, MVC

در کنار کتابخانه elmah که وظیفه ثبت تمامی خطاهای برنامه را دارد کتابخانه MiniProfiler امکان یافتن مشکلات کارایی و تنگناهای وب سایت را در اختیارمان قرار می‌دهد. دو قابلیت عمده که این ابزار فراهم می‌نمایید
امکان مشاهده و بررسی کوئری‌های خام ADO.NET از قبیل SQL Server,Oracle و LINQ-to-SQL و EF/First Code و...
نمایش زمان اجرای عملی صفحات

برای استفاده از این ابزار کافیست تا آن را از nuget دریافت نمایید

```
PM> Install-Package MiniProfiler
```

در ASP.NET MVC در صفحه _Layout قبل از بسته شدن تگ body تابع RenderIncludes را مانند زیر صدا بزنید تا در همه صفحات نمایش داده شود

```
@using StackExchange.Profiling;
<head>
 ..
</head>
<body>
  ...
  @MiniProfiler.RenderIncludes()
</body>
```

در کلاس global کد زیر را برای اجرای MiniProfiler اضافه نمایید

```
protected void Application_BeginRequest()
{
    if (Request.IsLocal)
    {
        MiniProfiler.Start();
    }
}

protected void Application_EndRequest()
{
    MiniProfiler.Stop();
}
```

برای پیکربندی MiniProfiler در web.config کد زیر را اضافه نمایید

```
<system.webServer>
  ...
  <handlers>
    <add name="MiniProfiler" path="mini-profiler-resources/*" verb="*"
         type="System.Web.Routing.UrlRoutingModule"
         resourceType="Unspecified"
         preCondition="integratedMode" />
  </handlers>
</system.webServer>
```

یا کتابخانه MiniProfiler.MVC را از nuget دریافت نمایید

```
PM> Install-Package MiniProfiler.MVC
```

با اضافه شدن این کتابخانه همه پیکربندی بصورت صورت خودکار انجام می‌گیرد. حال وب سایت را اجرا کنید در بالای صفحه

مانند شکل زیر مدت زمان بارگذاری صفحه نمایش داده می‌شود که با کلیک بر روی آن اطلاعات بیشتری را مشاهده می‌نمایید

اگر در اکشن اجرا شده کوئری اجرا شد باشد ستونی به نام query times نمایش داده می‌شود که تعداد کوئری‌ها و مدت زمان آن را نمایش می‌دهد

حال بر روی گزینه sql کلیک کنید که صفحه دیگری باز شود و کوئری خام آن را مشاهد نمایید اگر کوئری تکرار شده باشد در کنار آن با DUPLICATE متمایز شده است

برای مشاهده کوئری‌های Entity Framework/First Code کتابخانه MiniProfiler.EF را اضافه نمایید

```
PM> Install-Package MiniProfiler.EF
```

اگر بصورت دستی MiniProfiler را پیکربندی کرده باشید می‌بایست در Application_Start دستور زیر را اجرا نمایید

```
protected void Application_BeginRequest()
{
    if (Request.IsLocal)
    {
        MiniProfiler.Start();
        MiniProfilerEF.Initialize();
    }
}
```

در حالت پیشرفته‌تر اگر قصد داشته باشید زمان یک قطعه کد را جداگانه محاسبه نمایید بصورت زیر عمل نمایید

```
public ActionResult Index()
{

    var profiler = MiniProfiler.Current;

    using (profiler.Step("Step 1"))
    {
        //code 1
    }

    using (profiler.Step("Step 2"))
    {
        //code 2
    }

    return View();
}
```

با این کار زمان هر step را بصورت جداگانه محاسبه می‌نماید. در ASP.NET Webforms دقیقا به همین صورت استفاده می‌شود فقط

کافیست در masterpage اصلی یا اگر از masterpage استفاده نمی‌کنیم در صفحه مورد نظر تابع RenderIncludes را بصورت زیر صدا بزنیم

```
<%= StackExchange.Profiling.MiniProfiler.RenderIncludes() %>
```

امیدوارم مفید واقع شده باشد.

نظرات خوانندگان

نویسنده: سعید
تاریخ: ۱۳۹۱/۱۱/۱۵ ۱۸:۳

ممنون از شما. آخرین باری که miniprofiler.ef رو تست کردم با ef5 کار نمی‌کرد و خطا می‌داد. شما تست کردید با آخرین نگارش ef code first؟

نویسنده: مجتبی کاویانی
تاریخ: ۱۳۹۱/۱۱/۱۵ ۱۸:۱۴

نه هیچ مشکلی ندارد من خودم با ef code first 5 تست کردم

نویسنده: Alireza Godazchian
تاریخ: ۱۳۹۱/۱۱/۱۸ ۸:۴۶

با سلام؛ آقای مهندس خیلی عالی توضیح دادید و بنده هم تست کردم و جواب گرفتم.
با تشکر فراوان

نویسنده: امیر
تاریخ: ۱۳۹۱/۱۱/۱۸ ۲۰:۵۲

سلام چرا تو دانت ای اس پی کار نمیکنه فقط تو ام وی سی جواب میده
MiniProfiler نمیشناسه

نویسنده: مجتبی کاویانی
تاریخ: ۱۳۹۱/۱۱/۱۸ ۲۲:۸

در ASP.NET هم بخوبی جواب می‌دهد آیا Handler را در Web.config اضافه نموده اید؟

نویسنده: هیمن صادقی
تاریخ: ۱۳۹۳/۰۴/۲۷ ۱۶:۵۳

با سلام و درود بی کران
من کار که فرمودید انجام دادم اما زمانی که اجرا می‌کنم خطا

```
No Entity Framework provider found for the ADO.NET provider with invariant name 'System.Data.Odbc'. Make sure"
the provider is registered in the 'entityFramework' section of the application config file. See
".http://go.microsoft.com/fwlink/?LinkId=260882 for more information
```

میدهد
از mvc4 , entity framework6 استفاده می‌کنیم
در صورت امکان روش رفع خطا می‌دانید بیان نماید تا بتوانم از این ابزار استفاده کنم

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۴/۲۷ ۱۸:۲۲

برای EF 6 یک بسته‌ی نیوگت جداگانه دارد. اطلاعات بیشتر

عنوان: **ساخت ActionResult سفارشی**
نویسنده: محمد صاحب
تاریخ: ۱۳۹۱/۱۱/۲۱ ۱۷:۵۵
آدرس: www.dotnettips.info
برچسب‌ها: Unit testing, ASP.Net MVC

پیشتر با انواع ActionResult آشنا شدید. حال فرض کنید می‌خواهید نوعی رو برگردونید که براش ActionResult موجود نباشه مثلا RSS و یا فایل از نوع Excel و...
خوب، فرض کنید می‌خواهید اکشن متدی رو بنویسید که قراره نام یک فایل متنی رو بگیره و انو تو مروگر به کاربر نمایش بده.
برای اینکار از کلاس ActionResult، کلاس دیگه‌ی رو بنام TextResult به ارث می‌بریم و از این ActionResult سفارشی شده، در اکشن متد مربوطه استفاده می‌کنیم:

```
public class TextResult : ActionResult
{
    public string FileName { get; set; }
    public override void ExecuteResult(ControllerContext context)
    {
        var filePath = Path.Combine(context.HttpContext.Server.MapPath(@"~/Files/"), FileName);
        var data = File.ReadAllText(filePath);
        context.HttpContext.Response.Write(data);
    }
}
```

نحوه استفاده

```
public ActionResult DownloadTextFile(string fileName)
    {
        return new TextResult { FileName = fileName };
    }
```

در واقع متد اصلی اینجا ExecuteResult هست که نتیجه‌ی کار یک اکشن رو می‌تونیم پردازش کنیم.
خوب، سوالی که اینجا پیش میاد اینه که چرا این همه کار اضافی، چرا از Return File استفاده نمی‌کنی؟

```
public ActionResult DownloadTextFile(string fileName)
    {
        var filePath = Path.Combine(HttpContext.Server.MapPath(@"~/Files/"), fileName);
        return File(filePath, "text");
    }
```

یا کلا دلیل استفاده از ActionResult سفارشی چیه؟

جلوگیری از پیچیدگی و تکرار کد
همیشه کار مثل مورد بالا راحت و کم کد! نیست.
به مثال زیر توجه کنید که قراره خروجی CSV بهمون بده.

```
public class CsvActionResult : ActionResult
{
    public IEnumerable ModelListing { get; set; }

    public CsvActionResult(IEnumerable modelListing)
    {
        ModelListing = modelListing;
    }

    public override void ExecuteResult(ControllerContext context)
    {
        byte[] data = new CsvFileCreator().AsBytes(ModelListing);
        var fileResult = new FileContentResult(data, "text/csv")
        {
            FileDownloadName = "CsvFile.csv"
        };
        fileResult.ExecuteResult(context);
    }
```

```
}
```

و نحوه استفاده:

```
public ActionResult ExportUsers()
    {
        IEnumerable<User> model = UserRepository.GetUsers();
        return new CsvActionResult(model);
    }
```

حال فرض کنید بخواهیم همه این کدها رو داخل اکشن متد داشته باشیم، یکم پیچیده میشه و یا فرض کنید کنترلر دیگه‌ای نیاز به خروجی CSV داشته باشه، تکرار کد زیاد میشه.

راحت کردن گرفتن تست واحد از اکشن‌ها متدها

کاربرد ActionResult سفارشی تو تست واحد اینه که وابستگی‌های یک اکشن رو که Mock کردنش سخته می‌بریم داخل ActionResult و هنگام نوشتن تست واحد درگیر کار با اون وابستگی نمی‌شیم.
به مثال زیر توجه کنید که قراره برای اکشن Logout تست واحد بنویسیم
ابتدا بردن وابستگی‌ها به خارج از اکشن به کمک ActionResult سفارشی

```
public class LogoutActionResult : ActionResult
{
    public RedirectToRouteResult ActionAfterLogout { get; set; }
    public LogoutActionResult(RedirectToRouteResult actionAfterLogout)
    {
        ActionAfterLogout = actionAfterLogout;
    }
    public override void ExecuteResult(ControllerContext context)
    {
        FormsAuthentication.SignOut();
        ActionAfterLogout.ExecuteResult(context);
    }
}
```

نحوه استفاده از ActionResult سفارشی

```
public ActionResult Logout()
    {
        var redirect = RedirectToAction("Index", "Home");
        return new LogoutActionResult(redirect);
    }
```

و سپس نحوه تست واحد نوشتن

```
[TestMethod]
    public void The_Logout_Action_Returns_LogoutActionResult()
    {
        //arrange
        var account = new AccountController();

        //act
        var result = account.Logout() as LogoutActionResult;

        //assert
        Assert.AreEqual(result.ActionAfterLogout.RouteValues["Controller"], "Home");
    }
```

خوب به راحتی ما میایم فراخوانی متد SignOut رو از داخل اکشن می‌کشیم بیرون و این کار از اجرای متد SignOut از داخل اکشن متد جلوگیری می‌کنه و همچنین با این کار هنگام تست واحد نوشتن نیاز نیست با Mock کردن کلاس FormsAuthentication سروکار داشته باشیم و فقط کافیه چک کنیم خروجی از نوع LogoutActionResult هست یا خیر و یا می‌تونیم ActionAfterLogout

رو چک کنیم.

منابع و مراجع: + و +

عنوان: **استایل دهی به ستون های header در WebGrid**
نویسنده: محسن.د
تاریخ: ۱۳۹۱/۱۱/۲۸ ۰:۲۵
آدرس: www.dotnettips.info
برچسب‌ها: ASP.Net MVC, MVC

Webgrid گرید توکار asp.net mvc 3 است که در سری آموزش‌های mvc جناب نصیری به خوبی بررسی شده است . WebGrid از طریق مجموعه ای از خواص امکان استایل دهی به ستون‌ها و ردیف‌ها را به توسعه دهنده می‌دهد . اما در این بخش مشکلی وجود دارد که در ادامه به آن خواهم پرداخت . کدهای زیر را در نظر بگیرید

مدل‌ها :

```
public class Customer
    {
        public int Id { get; set; }

        public string Name { get; set; }

        public string Email { get; set; }

        public string Website { get; set; }

        public string Phone { get; set; }
    }

    public class Customers
    {
        public IList<Customer> GetList()
        {
            return new List<Customer>()
            {
                new Customer()
                {
                     Id=1,
                     Name="mohsen.d",
                     Email="email@domain.com",
                     Website="domain.com",
                     Phone="213214321"
                }
            };
        }

        public IList<Customer> GetEmptyList()
        {
            return new List<Customer>();
        }
    }
```

و کنترلر :

```
public class HomeController : Controller
    {
        public ActionResult List()
        {
            var model = new Customers().GetList();
            return View(model);
        }

        public ActionResult EmptyList()
        {
            var model = new Customers().GetEmptyList();
            return View("list", model);
        }
    }
```

تابع کمکی برای ایجاد گرید :

```
@helper GenerateList(IEnumerable<object> items, List<WebGridColumn> columns)
{
    var grid = new WebGrid(items);

    <div>
        @grid.GetHtml(
                        tableStyle: "list",
                        headerStyle: "list-header",
                        footerStyle: "list-footer",
                        alternatingRowStyle: "list-alt",
                        selectedRowStyle: "list-selected",
                        rowStyle: "list-row",
                        htmlAttributes: new { id = "listItems" },
                        mode: WebGridPagerModes.All,
                        columns: columns
    )

    </div>
}
```

: view

```
@model IEnumerable<WebGridHeaderStyle.Models.Customer>

@{
    ViewBag.Title = "List";
}

<h2>List</h2>

@_List.GenerateList(
    Model,
    new List<WebGridColumn>()
    {
        new WebGridColumn(){
         ColumnName="Id",
         Header="Id",
         Style="list-small-field"
        },
        new WebGridColumn(){
         ColumnName="Name",
         Header="Name",
         Style="list-long-field"
        },
        new WebGridColumn(){
         ColumnName="Email",
         Header="Email",
         Style="list-mid-field"
        },
        new WebGridColumn(){
         ColumnName="Website",
         Header="Website",
         Style="list-mid-field"
        },
        new WebGridColumn(){
         ColumnName="Phone",
         Header="Phone",
         Style="list-mid-field"
        }
    }
)
```

ابتدا به مسیر Home/List می‌رویم

List

Id	Name	Email	Website	Phone
1	mohsen.d	email@domain.com	domain.com	213214321

خوب چندان بد نیست . با استفاده از استایل‌های تعریف شده برای فیلدها و ردیف‌ها ، لیست ساختار مناسبی دارد . اما حالا به Home/EmptyList می رویم :

List

Id	Name	Email	Website	Phone

همانطور که می‌بینید استایل هایی که برای هر ستون تعریف کرده بودیم اعمال نشده اند. مشکل هم همین جاست . WebGrid استایل تعریف شده را تنها به ستون‌های درون tbody اعمال میکند و thead از این تنظیمات بی نصیب می‌ماند (WebGrid از table برای ساختن لیست استفاده می‌کند) و در زمانی که رکوردی وجود نداشته باشد فرمت طراحی شده اعمال نمی‌شود .

```
<div>
  <table id="listItems" class="list">
      <thead>
          <tr class="list-header">
              <th scope="col">Id</th>
              <th scope="col">Name</th>
              <th scope="col">Email</th>
              <th scope="col">Website</th>
              <th scope="col">Phone</th>
          </tr>
      </thead>
      <tbody>
          <tr class="list-row item">
              <td class="list-small-field">1</td>
              <td class="list-long-field">mohsen.d</td>
              <td class="list-mid-field">email@domain.com</td>
              <td class="list-mid-field">domain.com</td>
              <td class="list-mid-field">213214321</td>
          </tr>
      </tbody>
  </table>
</div>
```

در وب ترفندهایی را برای این مشکل پیدا کردم که اصلا جالب نبودند . در نهایت راه حل زیر به نظرم رسید :

در زمان ساختن گرید ، استایل‌های تعریف شده را در یک فیلد hidden ذخیره و سپس با استفاده از jquery این استایل‌ها را به ستون‌های header اعمال می‌کنیم .

تابع ساختن فیلد hidden :

```
@helper SetHeaderColumnsStyle(IEnumerable<WebGridColumn> columns)
{
    var styles = new List<string>();

    foreach(var col in columns)
    {
        styles.Add(col.Style);
    }

    <input id="styles" type="hidden" value="@string.Join("#",styles)" />
}
```

این تابع را در تابع کمکی ساخت گرید فراخوانی می‌کنیم :

```
@SetHeaderColumnsStyle(columns)
```

و در view کد javascript زیر را اضافه می‌کنیم :

```
<script>

    $(document).ready(function () {

        var styles = $("#styles").attr("value").split('#');

        var $cols = $("#listItems th");

        $cols.each(function (i) {
            $(this).addClass(styles[i]);
        });
    });
</script>
```

حال اگر صفحه را بارگذاری کنید با اینکه رکوردی وجود ندارد اما ساختار گرید به همان شکل تعریف شده باقی مانده است .

List

Id	Name	Email	Website	Phone

```
<div>
    <table id="listItems" class="list">
        <thead>
            <tr class="list-header">
                <th class="list-small-field" scope="col">Id</th>
                <th class="list-long-field" scope="col">Name</th>
                <th class="list-mid-field" scope="col">Email</th>
                <th class="list-mid-field" scope="col">Website</th>
                <th class="list-mid-field" scope="col">Phone</th>
            </tr>
        </thead>
        <tbody></tbody>
    </table>
</div>
<input id="styles" type="hidden" value="list-small-field#list-long-field#list-mid-field#list-mid-field#list-
mid-field">
<script>
```

پروژه نمونه را می‌توانید از اینجا دانلود کنید .

عنوان: **ELMAH و حملات XSS**
نویسنده: داود زینی
تاریخ: ۱۳۹۱/۱۱/۲۸ ۱۳:۴۵
آدرس: www.dotnettips.info
برچسب‌ها: MVC, Security, ASP.Net MVC, ELMAH, xss

ASP.NET جهت مقابله با حملات XSS بطور پیش‌فرض از ورود تگ‌های HTML جلوگیری می‌کند و در صورتی که ورودی کاربر شامل این تگ‌ها باشد، HttpRequestValidationException صادر می‌گردد. لاگ کردن و بررسی این خطاها جهت آگاهی از وجود حمله بی اهمیت نیست. اما متأسفانه ELMAH که به عنوان معمول‌ترین ابزار ثبت خطاها کاربرد دارد این نوع Exceptionها را ثبت نمی‌کند. دلیل آن هم این است که ELMAH در رویه‌های درونی خود اقدام به خواندن ورودی‌های کاربر می‌کند و در این هنگام اگر ورودی کاربر نامعتبر باشد، Exception مذکور صادر می‌شود و فرصتی برای ادامه روند و ثبت خطا باقی نمی‌ماند. به هر حال ضروری است که این نقیصه را خودمان جبران کنیم. راه حل افزودن یک فیلتر سفارشی برای ثبت خطاها به شکل زیر است (ASP.NET MVC):

```
public class ElmahRequestValidationErrorFilter : IExceptionFilter
{
    public void OnException(ExceptionContext context)
    {
        if (context.Exception is HttpRequestValidationException)
           ErrorLog.GetDefault(HttpContext.Current).Log(new Error(context.Exception));
    }
}
```

فیلتر فوق باید در Global.asax معرفی شود:

```
public static void RegisterGlobalFilters (GlobalFilterCollection filters)
{
    filters.Add(new ElmahRequestValidationErrorFilter());
    filters.Add(new HandleErrorAttribute());
}
```

به این ترتیب HttpRequestValidationException هم بعد از این در سیستم ELMAH ثبت خواهد شد.

نظرات خوانندگان

نویسنده: یاسر مرادی
تاریخ: ۱۳۹۱/۱۱/۲۸ ۱۴:۳

با سلام، در چک لیست ASP.NET MVC مورد زیر وجود دارد
آیا مورد زیر کماکا معتبر است ؟
- فیلتر پیش فرض مدیریت خطاها حذف و بجای آن از ELMAH استفاده شود.

با سپاس

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۱/۲۸ ۱۴:۲۶

- به سطر HandleErrorAttribute پیش فرض نیازی نیست (البته اگر تنظیمات وب کانفیگ درستی داشته باشید). در قسمت 16 سری MVC توضیح دادم. وجود آن سبب می‌شود که ELMAH اصلا کار نکند و خطای مدیریت نشده‌ای به آن ارجاع داده نشود (چون قبلا مدیریت شده).
- بله. همچنان ELMAH معتبر است. نکته فوق را هم اضافه کنید، کاملتر خواهد شد.

نویسنده: داود زینی
تاریخ: ۱۳۹۱/۱۱/۲۸ ۱۴:۲۸

سلام
این چک لیست توسط آقای نصیری تهیه شده بود. این مورد هم از نظر من معتبر است.
با تشکر

عنوان: **حذف هدرهای مربوط به وب سرور از طریق برنامه نویسی**
نویسنده: سیروان عفیفی
تاریخ: ۱۳۹۱/۱۲/۱۹ ۲۳:۲۰
آدرس: www.dotnettips.info
گروه‌ها: ASP.Net, IIS, ASP.Net MVC, MVC, Security

در تکمیل این مطلب برای حذف هدرهای مربوط به وب سرور در برنامه‌های ASP.NET MVC از روش زیر می‌توانیم استفاده کنیم.

در حالت پیش فرض تمام پاسخهای که به سمت سرور ارسال میشوند به همراه خود یک سری جزئیات را نیز منتقل میکنند.

برای یک وب اپلیکیشن APS.NET MVC این هدرها را داریم :

Server: که توسط IIS اضافه میشود.

X-AspNet-Version: که در زمانFlush در httpresponse اضافه میشود.

X-AspNetMvc-Version: که توسط MvcHandler در System.Web.dll اضافه میشود.

X-Powered-By: این مورد نیز توسط IIS اضافه میشود.

هکرها از اینکه فریم ورک مورد استفاده چه چیزی است خوشحال خواهند شد: اگر سرور شما برای مدتی Update نشده باشد و یک آسیب پذیری امنیتی بزرگ برای ورژن فریم ورکی که استفاده می‌کنید پیدا شود در نتیجه به هکرها برای رسیدن به هدفشان

کمک کرده اید.

به علاوه این هدرها فضایی را برای تمام پاسخ‌ها در نظر میگیرند (البته در حد چندین بایت ولی در اینجا بحث برروی Optimization است).

برای حذف این هدرها باید مراحل زیر را انجام دهیم:

حذف کردن هدر Server : به Global.asax.cs رفته و رویداد Application_PreSendRequestHeaders با کد زیر را به آن اضافه کنید :

```
protected void Application_PreSendRequestHeaders(object sender, EventArgs e)
     {
         var app = sender as HttpApplication;
         if (app == null || !app.Request.IsLocal || app.Context == null)
             return;
         var headers = app.Context.Response.Headers;
         headers.Remove("Server");
     }
```

حذف کردن هدر X-AspNetMvc-Version: در فایل Global.asax.cs به رویداد Application_Start این کد زیر را اضافه کنید :

```
protected void Application_Start()
     {
         ...
         MvcHandler.DisableMvcResponseHeader = true;
         ...
     }
```

حذف کردن هدر X-AspNet-Version: به فایل Web.Config مراجعه کرده و این المنت را در داخل system.web اضافه کنید:

```
<system.web>
         ...
         <httpRuntime enableVersionHeader="false" />
         ...
</system.web>
```

حذف کردن هدر X-Powered-By: در داخل فایل Web.Config در داخل system.webServer این خطوط را اضافه کنید:

```
<system.webServer>
         ...
         <httpProtocol>
             <customHeaders>
                 <remove name="X-Powered-By" />
             </customHeaders>
         </httpProtocol>
         ...
</system.webServer>
```

با انجام مراحل فوق پاسخ‌های سرور سبک‌تر شده و در نهایت حاوی اطلاعات مهم در مورد ورژن فریم ورک نمی‌باشد.

نظرات خوانندگان

نویسنده: sysman
تاریخ: ۱۳۹۲/۰۱/۱۴ ۰:۲۲

ممنون از این مطلب مفید ولی من یک مشکلی دارم کاری که گفتید را انجام دادم وهمه چیز خوب انجام شد ولی در بخش GET modernizr-2.5.3.js باز هم اطلاعات سرور رو به من میدهد

نویسنده: علي
تاریخ: ۱۳۹۲/۰۱/۱۴ ۰:۳۶

فایل‌های استاتیک رو هم باید به موتور ASP.NET مپ کنید. تا زمانیکه مپ نباشند مستقیما توسط IIS سرو می‌شن و تنظیمات ASP.NET روی اون‌ها تاثیری نداره.

نویسنده: sysman
تاریخ: ۱۳۹۲/۰۱/۱۴ ۱۰:۴۰

نمی دونم این کاری که گفتین رو دقیقا چطور انجام بدهم ولی این کار باعث ایجاد کندی نمی‌شه؟ اگر در خود IIS تنظیمات این پست رو اعمال کنم به نتیجه میرسم؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۱/۱۴ ۱۱:۱۳

- برای فایل‌های جاوا اسکریپت توصیه من این است:
الف) اگر از Web forms استفاده می‌کنید: استفاده از Script manager (^ و ^)
ب) اگر از MVC استفاده می‌کنید: استفاده از Bundling & minification
در هر دو حالت نحوه ارائه اسکریپت‌ها تحت کنترل برنامه ASP.NET در خواهد آمد و مستقیما و بدون دخالت ASP.NET، توسط IIS توزیع نمی‌شوند.
- برای مپ کردن فایل‌های استاتیک به موتور ASP.NET می‌شود از StaticFileHandler استفاده کرد. اگر کش کردن اطلاعات استاتیک در سمت سرور فعال شود، این مساله بار اضافه‌ای را به سرور تحمیل نخواهد کرد.

```
<system.web>
   <httpHandlers>
      <add path="*.js" verb="*" type="System.Web.StaticFileHandler" />
   </httpHandlers>
```

نویسنده: یونس دوست
تاریخ: ۱۳۹۲/۰۶/۰۲ ۱۱:۳۴

سلام.من ازش استفاده کردم ولی موقع حذف Header مربوط به سرور ارور زیر رو میده:
This operation requires IIS integrated pipeline mode.
ضمنا من از iis 7.5 استفاده می‌کنم و توی Application Pools هم .Net AppPool Classic رو در حالت pipeline mode:Integrated قرار دادم هم DefaultAppPool ولی همچنان ارور رو دارم.مشکل از کجاست؟
ممنون.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۶/۰۲ ۱۱:۵۴

- محل تست کردن واقعی این کدها بر روی ویندوز سرور است و نه دیباگر و وب سرور آزمایشی ویژوال استودیو.
- شما اگر تا این حد دسترسی به IIS دارید، اصلا نیازی به کدنویسی برای حذف هدرهای وب سرور نخواهید داشت. به قسمت HTTP response headers کنسول مدیریتی مراجعه و مداخل موجود را حذف یا ویرایش کنید.
+ حذف هدر مربوط به نام Server کار ساده‌ای نیست. خیلی‌ها از روش HTTP module هم جواب نگرفته‌اند، اما با استفاده از URL Scan خود مایکروسافت قابل حذف است (این برنامه روی ویندوزهای سرور 2003 به بعد قابل نصب است). بعد از نصب به فایل C:\Windows\System32\inetsrv\urlscan\UrlScan.ini مراجعه و RemoveServerHeader را با 1 مقدار دهی کنید. ضمنا قبل از نصب URLScan تغییر زیر را هم امتحان کنید (بجای Remove از Set استفاده شده):

```
void OnPreSendRequestHeaders(object sender, EventArgs e)
{
   HttpContext.Current.Response.Headers.Set("Server", "CERN httpd");
}
```

نویسنده: پیام دات نت
تاریخ: ۱۳۹۳/۰۳/۱۵ ۲۰:۳۵

با سلام؛ ممنون از مطلب خوبتون. من تو IIS 8 Local اجرا کردم درست کار میکنه و بسیار عالیه. اما تو اجرای نهایی روی سرور هدر Server حذف نشد و از سمت سرور ارسال میشد. بعد از کلی جستجو به نتیجه زیر رسیدم.

```
<system.webServer>
    <rewrite>
      <outboundRules>
        <rule name="Remove RESPONSE_Server" >
          <match serverVariable="RESPONSE_Server" pattern=".+" />
          <action type="Rewrite" value="" />
        </rule>
      </outboundRules>
    </rewrite>
  </system.webServer>
```

نویسنده: دانش پژوه
تاریخ: ۱۳۹۳/۱۰/۰۲ ۱۱:۳۹

ممنون از مطلبتون

فقط یک نکته رو میخواستم بگم نام کوکی سشن بطور پیشفرض ASP.NET_SessionId هست و میشه از این نام فهمید که سرور asp.net هستش. از طریق کد زیر توی وب کانفیگ میتونید تغییرش بدید:

```
<system.web>
    <sessionState cookieName="foo" />
</system.web>
```

عنوان: **ایجاد helper برای Nivo Slider در Asp.net Mvc**
نویسنده: علی حق جو
تاریخ: ۱۳۹۲/۰۱/۲۹ ۱۶:۳۵
آدرس: www.dotnettips.info
برچسب‌ها: ASP.Net MVC, jQuery, NivoSlider, Mvc helper, MVC

کامپوننت‌های jQuery زیادی وجود دارند که توسط آنها میتوان تصاویر را بصورت زمانبندی شده و به همراه افکت‌های زیبا در سایت خود نشان داد. مانند اینجا .

در این قصد ایجاد helper برای کامپوننت NivoSlider را داریم.

1- یک پروژه Asp.net Mvc 4.0 ایجاد میکنیم.

2- سپس فایل jquery.nivo.slider.pack.js ، فایل‌های css مربوط به این کامپوننت و چهار تم موجود را از سایت این کامپوننت و یا درون سورس مثال ارائه شده دریافت می‌کنیم.

3- به کلاس BundleConfig رفته و کدهای زیر را اضافه میکنیم:

```
#region Nivo Slider

   bundles.Add(new StyleBundle("~/Content/NivoSlider").Include("~/Content/nivoSlider/nivo-
slider.css"));
   bundles.Add(new
```

```
StyleBundle("~/Content/NivoSliderDefaultTheme").Include("~/Content/nivoSlider/themes/default/default.cs
s"));
   bundles.Add(new
StyleBundle("~/Content/NivoSliderDarkTheme").Include("~/Content/nivoSlider/themes/dark/dark.css"));
   bundles.Add(new
StyleBundle("~/Content/NivoSliderLightTheme").Include("~/Content/nivoSlider/themes/light/light.css"));
   bundles.Add(new
StyleBundle("~/Content/NivoSliderBarTheme").Include("~/Content/nivoSlider/themes/bar/bar.css"));
   bundles.Add(new
ScriptBundle("~/bundles/NivoSlider").Include("~/Scripts/jquery.nivo.slider.pack.js"));

#endregion
```

سپس در فایل shared آنها را بصورت زیر اعمال میکنیم:

```
<!DOCTYPE html>
<html>
<head>
    <title>@ViewBag.Title</title>
    <script type="text/javascript" src="~/Scripts/jquery-1.9.1.min.js"></script>
    @Styles.Render("~/Content/NivoSlider",
"~/Content/NivoSliderDefaultTheme","~/Content/NivoSliderLightTheme")
    @Scripts.Render("~/bundles/NivoSlider")
</head>
<body>
    @RenderBody()
    @RenderSection("scripts", required: false)
</body>
</html>
```

4- یک پوشه با عنوان helper به پروژه اضافه میکنیم. سپس کلاس‌های زیر را به آن اضافه می‌کنیم :

```
public class NivoSliderHelper
 {
        #region Fields

        private string _id = "nivo1";
        private List<NivoSliderItem> _models = null;
        private string _width = "100%";
        private NivoSliderTheme _theme = NivoSliderTheme.Default;
        private string _effect = "random"; // Specify sets like= 'fold;fade;sliceDown'
        private int _slices = 15; // For slice animations
        private int _boxCols = 8; // For box animations
        private int _boxRows = 4; // For box animations
        private int _animSpeed = 500; // Slide transition speed
        private int _pauseTime = 3000; // How long each slide will show
        private int _startSlide = 0; // Set starting Slide (0 index)
        private bool _directionNav = true; // Next & Prev navigation
        private bool _controlNav = true; // 1;2;3... navigation
        private bool _controlNavThumbs = false; // Use thumbnails for Control Nav
        private bool _pauseOnHover = true; // Stop animation while hovering
        private bool _manualAdvance = false; // Force manual transitions
        private string _prevText = "Prev"; // Prev directionNav text
        private string _nextText = "Next"; // Next directionNav text
        private bool _randomStart = false; // Start on a random slide
        private string _beforeChange = ""; // Triggers before a slide transition
        private string _afterChange = ""; // Triggers after a slide transition
        private string _slideshowEnd = ""; // Triggers after all slides have been shown
        private string _lastSlide = ""; // Triggers when last slide is shown
        private string _afterLoad = "";

        #endregion

        string makeParameters()
        {
            var builder = new StringBuilder();
            builder.Append("{");
            builder.Append(string.Format("effect:'{0}'", _effect));
            builder.AppendLine(string.Format(",slices:{0}", _slices));
            builder.AppendLine(string.Format(",boxCols:{0}", _boxCols));
            builder.AppendLine(string.Format(",boxRows:{0}", _boxRows));
            builder.AppendLine(string.Format(",animSpeed:{0}", _animSpeed));
```

```
            builder.AppendLine(string.Format(",pauseTime:{0}", _pauseTime));
            builder.AppendLine(string.Format(",startSlide:{0}", _startSlide));
            builder.AppendLine(string.Format(",directionNav:{0}", _directionNav.ToString().ToLower()));
            builder.AppendLine(string.Format(",controlNav:{0}", _controlNav.ToString().ToLower()));
            builder.AppendLine(string.Format(",controlNavThumbs:{0}",
_controlNavThumbs.ToString().ToLower()));
            builder.AppendLine(string.Format(",pauseOnHover:{0}", _pauseOnHover.ToString().ToLower()));
            builder.AppendLine(string.Format(",manualAdvance:{0}",
_manualAdvance.ToString().ToLower()));
            builder.AppendLine(string.Format(",prevText:'{0}'", _prevText));
            builder.AppendLine(string.Format(",nextText:'{0}'", _nextText));
            builder.AppendLine(string.Format(",randomStart:{0}", _randomStart.ToString().ToLower()));
            builder.AppendLine(string.Format(",beforeChange:{0}", _beforeChange));
            builder.AppendLine(string.Format(",afterChange:{0}", _afterChange));
            builder.AppendLine(string.Format(",slideshowEnd:{0}", _slideshowEnd));
            builder.AppendLine(string.Format(",lastSlide:{0}", _lastSlide));
            builder.AppendLine(string.Format(",afterLoad:{0}", _afterLoad));
            builder.Append("}");
            return builder.ToString();
        }
        public NivoSliderHelper (
            string id,
            List<NivoSliderItem> models,
            string width = "100%",
            NivoSliderTheme theme = NivoSliderTheme.Default,
            string effect = "random",
            int slices = 15,
            int boxCols = 8,
            int boxRows = 4,
            int animSpeed = 500,
            int pauseTime = 3000,
            int startSlide = 0,
            bool directionNav = true,
            bool controlNav = true,
            bool controlNavThumbs = false,
            bool pauseOnHover = true,
            bool manualAdvance = false,
            string prevText = "Prev",
            string nextText = "Next",
            bool randomStart = false,
            string beforeChange = "function(){}",
            string afterChange = "function(){}",
            string slideshowEnd = "function(){}",
            string lastSlide = "function(){}",
            string afterLoad = "function(){}")
        {
            _id = id;
            _models = models;
            _width = width;
            _theme = theme;
            _effect = effect;
            _slices = slices;
            _boxCols = boxCols;
            _boxRows = boxRows;
            _animSpeed = animSpeed;
            _pauseTime = pauseTime;
            _startSlide = startSlide;
            _directionNav = directionNav;
            _controlNav = controlNav;
            _controlNavThumbs = controlNavThumbs;
            _pauseOnHover = pauseOnHover;
            _manualAdvance = manualAdvance;
            _prevText = prevText;
            _nextText = nextText;
            _randomStart = randomStart;
            _beforeChange = beforeChange;
            _afterChange = afterChange;
            _slideshowEnd = slideshowEnd;
            _lastSlide = lastSlide;
            _afterLoad = afterLoad;
        }
        public IHtmlString GetHtml()
        {
            var thm = "theme-" + _theme.ToString().ToLower();
            var sb = new StringBuilder();
            sb.AppendLine("<div style='width:" + _width + ";height:auto;margin:0px auto' class='" + thm
+ "'>");
            sb.AppendLine("<div id='" + _id + "' class='nivoSlider'>");
            foreach (var model in _models)
            {
                string img = string.Format("<img src='{0}' alt='{1}' title='{2}' />",
```

```
model.ImageFilePath, "", model.Caption);
                string item = "";
                if (model.LinkUrl.Trim().Length > 0 &&
                    Uri.IsWellFormedUriString(model.LinkUrl, UriKind.RelativeOrAbsolute))
                {
                    item = string.Format("<a href='{0}'>{1}</a>", model.LinkUrl, img);
                }
                else
                {
                    item = img;
                }
                sb.AppendLine(item);
            }
            sb.AppendLine("</div>");
            sb.AppendLine("</div>");

            sb.AppendLine("<script type='text/javascript'>");
            //sb.AppendLine("$('#" + _id + "').parent().ready(function () {");
            sb.Append("$(document).ready(function(){");
            sb.AppendLine("$('#" + _id + "').nivoSlider(" + makeParameters() + ");");
            //sb.AppendLine("$('.nivo-controlNav a').empty();");//semi hack for rtl layout
            //sb.AppendLine("$('#" + _id + "').nivoSlider();");
            sb.AppendLine("});");
            sb.AppendLine("</script>");

            return new HtmlString(sb.ToString());
        }
    }
```

```
public enum NivoSliderTheme
    {
        Default, Light, Dark, Bar,
    }
```

```
public class NivoSliderItem
    {
        public string ImageFilePath { get; set; }
        public string Caption { get; set; }
        public string LinkUrl { get; set; }
    }
```

-6سپس برای استفاده از این helper یک کنترلر به پروژه اضافه میکنیم مانند Home :

```
public class HomeController : Controller
    {
        public ActionResult Index()
        {
            return View();
        }
        public virtual ActionResult NivoSlider()
        {
            var models = new List<NivoSliderItem>
            {
                new NivoSliderItem()
                    {
                        ImageFilePath =Url.Content("~/Images/img1.jpg"),
                        Caption = "عنوان اول",
                        LinkUrl = "http://www.google.com",
                    },
                new NivoSliderItem()
                    {
                        ImageFilePath =Url.Content("~/Images/img2.jpg"),
                        Caption = "#htmlcaption",
                        LinkUrl = "",
                    },
                new NivoSliderItem()
                    {
                        ImageFilePath =Url.Content("~/Images/img3.jpg"),
                        Caption = "عنوان سوم",
                        LinkUrl = "",
                    },
                new NivoSliderItem()
                    {
                        ImageFilePath =Url.Content("~/Images/img4.jpg"),
                        Caption = "عنوان چهارم",
                        LinkUrl = "",
```

```
                    },
                new NivoSliderItem()
                    {
                        ImageFilePath =Url.Content("~/Images/img5.jpg"),
                        Caption = "عنوان پنجم",
                        LinkUrl = "",
                    }
            };
            return PartialView("_NivoSlider", models);
        }
    }
```

7- سپس ویوی Home را نیز ایجاد میکنیم:

```
@{
    ViewBag.Title = "Index";
}

<h2>Nivo Slider Index</h2>

@{ Html.RenderAction("NivoSlider", "Home");}
```

8- یک پارشال ویو نیز برای رندر کردن NivoSliderهای خود ایجاد میکنیم:

```
@using NivoSlider.Helper
@model List<NivoSliderItem>
@{
    var nivo1 = new NivoSliderHelper("nivo1", Model.Skip(0).Take(3).ToList(), theme:
NivoSliderTheme.Default, width: "500px");
    var nivo2 = new NivoSliderHelper("nivo2", Model.Skip(3).Take(2).ToList(), theme:
NivoSliderTheme.Light, width: "500px");
}
@nivo1.GetHtml()

<div id="htmlcaption">
    <strong>This</strong> is an example of a <em>HTML</em> caption with <a href="#">a link</a>.
</div>

<br />
<br />
@nivo2.GetHtml()
```

نکته اول : اگر بخواهیم بر روی تصویر موردنظر متنی نمایش دهیم کافیست خاصیت Caption را مقداردهی کنیم. برای نمایش توضیحاتی پیچیده‌تر (مانند نمایش یک div و لینک و غیره) بعنوان توضیحات یک تصویر خاص باید خاصیت Caption را بصورت "#{نام عنصر}" مقداردهی کنیم و ویژگی class مربوط به عنصر موردنظر را با nivo-html-caption مقداردهی کنیم. (همانند مثال بالا)

نکته دوم : این کامپوننت به همراه 4 تم (deafult- light- bar- dark) ارائه شده است. موقع استفاده از تم‌های دیگر باید رفرنس مربوط به آن تم را نیز اضافه کنید.

امیدوارم مفید واقع شود. دریافت پروژه نمونه این مقاله

نظرات خوانندگان

نویسنده: bahman
تاریخ: ۱۳۹۲/۰۲/۰۸ ۱۱:۴۴

سلام تشکر میکنم .خیلی عالی بود.
فقط یه مشکلی هست اونم نبودن WebGrease درون DLLهای شماست.
چطوری میتونم پیداش کنم؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۲/۰۸ ۱۱:۴۷

این dll جزو فایل‌های استاندارد یک پروژه ASP.NET MVC **4** است. در اینجا پیوست نشده، چون نیازی نیست هربار تمام این فایل‌های تکراری را دریافت کنید.
فایل‌های آن در دو پوشه زیر بر روی سیستم شما موجود هستند: (اگر ASP.NET MVC4 را نصب کرده باشید)
C:\Program Files\Microsoft ASP.NET\ASP.NET MVC **4** \Packages
C:\Program Files\Microsoft ASP.NET\ASP.NET Web Pages\v **2.0** \Packages

نویسنده: bahman
تاریخ: ۱۳۹۲/۰۲/۰۸ ۱۱:۵۴

مشکل با nuget حل شد.

عنوان: آشنایی با Fluent Html Helpers در MVC
نویسنده: حمیدرضا صادقیان
تاریخ: ۱۳۹۲/۰۱/۳۰ ۱۸:۲۰
آدرس: www.dotnettips.info
گروه‌ها: ASP.Net MVC, Fluent Html Helpers, MVC

همان طور که قبلا نیز اشاره شد اینجا به صورت خلاصه هدف Fluent API فراهم آوردن روشی است که بتوان متدها را زنجیر وار فراخوانی کرد و به این ترتیب خوانایی کد نوشته شده را بالا برد.

اما در این مقاله سعی شده تا کاربرد آن در یک برنامه MVC رو به صورت استفاده در helperها شرح دهیم.
در اینجا مثالی رو شرح میدهیم که در آن کنترل هایی از جنس input به صورت helper ساخته و برای فرستادن ویژگی‌های اچ تی ام ال (HTML Attributes) آن از Fluent Html Helpers بهره میگیریم بدیهی است که از این Fluent میتوان برای helperهای دیگر هم استفاده کرد.
در ابتدا یک نگاه کلی به کد ایجاد شده می‌اندازیم :

```
@Html.RenderInput(
        Attributes.Configure()
            .AddType("text")
            .AddId("UserId")
            .AddName("UserId")
            .AddCssClass("TxtBoxCssClass")
    )
```

که باعث ساخته شدن یک کنترل تکست باکس با مشخصات زیر در صفحه می‌شود ، همان طور که مشاهده می‌کنیم تمام ویژگی‌ها برای کنترل ساخته شده اند.

```
<input class="TxtBoxCssClass" id="UserId" name="UserId" type="text">
```

در ادامه به سراغ پیاده سازی کلاس Attributes برای این مثال می‌رویم ، این کلاس باید از کلاس RouteValueDictionary ارث بری داشته باشد، این کلاس یک دیکشنری برای ما آماده می‌کند تا مقادیرمون رو در آن بریزیم در اینجا برای ویژگی‌ها و مقادیرشون.

```
public class Attributes : RouteValueDictionary    {
        /// <summary>
        /// Configures this instance.
        /// </summary>
        /// <returns></returns>
        public static Attributes Configure()
        {
            return new Attributes();
        }

        /// <summary>
        /// Adds the type.
        /// </summary>
        /// <param name="value">The value.</param>
        public Attributes AddType(string value)
        {
            this.Add("type", value);
            return this;
        }

        /// <summary>
        /// Adds the name.
        /// </summary>
        /// <param name="value">The value.</param>
        public Attributes AddName(string value)
        {
            this.Add("name", value);
            return this;
        }

        /// <summary>
        /// Adds the id.
        /// </summary>
```

```
        /// <param name="value">The value.</param>
        public Attributes AddId(string value)
        {
            this.Add("id", value);
            return this;
        }

        /// <summary>
        /// Adds the value.
        /// </summary>
        /// <param name="value">The value.</param>
        public Attributes AddValue(string value)
        {
            this.Add("value", value);
            return this;
        }

        /// <summary>
        /// Adds the CSS class.
        /// </summary>
        /// <param name="value">The value.</param>
        public Attributes AddCssClass(string value)
        {
            this.Add("class", value);
            return this;
        }
    }
```

متد استاتیک Configure همیشه به عنوان شروع کننده fluent و از اون برای ساختن یک وهله جدید از کلاس Attributes استفاده میشه و بقیه متدها هم کار اضافه کردن مقادیر رو مثل یک دیکشنری انجام میدهند.

حالا به سراغ پیاده سازی helper extension مون میرویم

```
public static MvcHtmlString RenderInput(this HtmlHelper htmlHelper, Attributes attributes)
        {
            TagBuilder input = new TagBuilder("input");
            input.MergeAttributes(attributes);
            return new MvcHtmlString(input.ToString(TagRenderMode.SelfClosing));
        }
```

با استفاده از کلاس tagbuilder تگ input را ساخته و ویژگی‌های فرستاده شده به helper رو با اون ادغام می‌کنیم (با استفاده از MergeAttributes)

این یک مثال ساده بود از این کار امیدوارم مفید واقع شده باشد

مثال پروژه

عنوان: Globalization در ASP.NET MVC - قسمت دوم
نویسنده: یوسف نژاد
تاریخ: ۱۳۹۲/۰۱/۳۱ ۱۳:۵
آدرس: www.dotnettips.info
برچسب‌ها: MVC, ASP.Net MVC, Globalization, Resource, Updateable Resource

به‌روزرسانی فایلهای Resource در زمان اجرا یکی از ویژگیهای مهمی که در پیاده سازی محصول با استفاده از فایلهای Resource باید به آن توجه داشت، امکان بروز رسانی محتوای این فایلها در زمان اجراست. از آنجاکه احتمال اینکه کاربران سیستم خواهان تغییر این مقادیر باشند بسیار زیاد است، بنابراین درنظر گرفتن چنین ویژگی‌ای برای محصول نهایی میتواند بسیار تعیین کننده باشد. متاسفانه پیاده سازی چنین امکانی درباره فایلهای Resource چندان آسان نیست. زیرا این فایلها همانطور که در قسمت قبل توضیح داده شد پس از کامپایل به صورت اسمبلی‌های ستلایت (Satellite Assembly) درآمده و دیگر امکان تغییر محتوای آنها بصورت مستقیم و به آسانی وجود ندارد.

نکته: البته نحوه پیاده سازی این فایلها در اسمبلی نهایی (و در حالت کلی نحوه استفاده از هر فایلی در اسمبلی نهایی) در ویژوال استودیو توسط خاصیت Build Action تعیین میشود. برای کسب اطلاعات بیشتر راجع به این خاصیت به اینجا رجوع کنید.

یکی از روشهای نسبتا من‌درآوردی که برای ویرایش و به روزرسانی کلیدهای Resource وجود دارد بدین صورت است:
- ابتدا باید اصل فایلهای Resource به همراه پروژه پابلیش شود. بهترین مکان برای نگهداری این فایلها فولدر App_Data است. زیرا محتویات این فولدر توسط سیستم FCN (همان *File Change Notification*) در ASP.NET رصد نمیشود.
نکته: علت این حساسیت این است که FCN در ASP.NET تقریبا تمام محتویات فولدر سایت در سرور (فولدر App_Data یکی از معدود استثناهاست) را تحت نظر دارد و رفتار پیشفرض این است که با هر تغییری در این محتویات، AppDomain سایت Unload میشود که پس از اولین درخواست دوباره Load میشود. این اتفاق موجب از دست دادن تمام سشن‌ها و محتوای کش‌ها و ... میشود (اطلاعات بیشتر و کاملتر درباره نحوه رفتار FCN در اینجا).
- سپس با استفاده یک مقدار کدنویسی امکاناتی برای ویرایش محتوای این فایلها فراهم شود. ازآنجا که محتوای این فایلها به صورت XML ذخیره میشود بنابراین براحتی میتوان با امکانات موجود این ویژگی را پیاده سازی کرد. اما در فضای نام System.Windows.Forms کلاسهایی وجود دارد که مخصوص کار با این فایلها طراحی شده اند که کار نمایش و ویرایش محتوای فایلهای Resource را ساده‌تر میکند. به این کلاسها در قسمت قبلی اشاره کوتاهی شده بود.
- پس از ویرایش و به روزرسانی محتوای این فایلها باید کاری کنیم تا برنامه از این محتوای تغییر یافته به عنوان منبع جدید بهره بگیرد. اگر از این فایلهای Rsource به صورت embed استفاده شده باشد در هنگام build پروژه محتوای این فایلها به صورت Satellite Assembly در کنار کتابخانه‌های دیگر تولید میشود. اسمبلی مربوط به هر زبان هم در فولدری با عنوان زبان مربوطه ذخیره میشود. مسیر و نام فایل این اسمبلی‌ها مثلا به صورت زیر است:

```
bin\fa\Resources.resources.dll
```

بنابراین در این روش برای استفاده از محتوای به روز رسانی شده باید عملیات Build این کتابخانه دوباره انجام شود و کتابخانه‌های جدیدی تولید شود. راه حل اولی که به ذهن میرسد این است که از ابزارهای پایه و اصلی برای تولید این کتابخانه‌ها استفاده شود. این ابزارها (همانطور که در قسمت قبل نیز توضیح داده شد) عبارتند از Resource Generator و Assembly Linker. اما استفاده از این ابزارها و پیاده سازی روش مربوطه سختتر از آن است که به نظر می‌آید. خوشبختانه درون مجموعه عظیم دات نت ابزار مناسبتری برای این کار نیز وجود دارد که کار تولید کتابخانه‌های موردنظر را به سادگی انجام میدهد. این ابزار با عنوان Microsoft Build شناخته میشود که در اینجا توضیح داده شده است.

خواندن محتویات یک فایل resx.

همانطور که در بالا توضیح داده شد برای راحتی کار میتوان از کلاس زیر که در فایل System.Windows.Forms.dll قرار دارد استفاده کرد:

```
System.Resources.ResXResourceReader
```

این کلاس چندین کانستراکتور دارد که مسیر فایل resx. یا استریم مربوطه به همراه چند گزینه دیگر را به عنوان ورودی میگیرد. این کلاس یک Enumator دارد که یک شی از نوع IDictionaryEnumerator برمیگرداند. هر عضو این enumerator از نوع object است. برای استفاده از این اعضا ابتدا باید آنرا به نوع DictionaryEntry تبدیل کرد. مثلا بصورت زیر:

```
private void TestResXResourceReader()
{
  using (var reader = new ResXResourceReader("Resource1.fa.resx"))
  {
    foreach (var item in reader)
    {
      var resource = (DictionaryEntry)item;
      Console.WriteLine("{0}: {1}", resource.Key, resource.Value);
    }
  }
}
```

همانطور که ملاحظه میکنید استفاده از این کلاس بسیار ساده است. ازآنجاکه DictionaryEntry یک struct است، به عنوان یک راه حل مناسبتر بهتر است ابتدا کلاسی به صورت زیر تعریف شود:

```
public class ResXResourceEntry
{
  public string Key { get; set; }
  public string Value { get; set; }
  public ResXResourceEntry() { }
  public ResXResourceEntry(object key, object value)
  {
    Key = key.ToString();
    Value = value.ToString();
  }
  public ResXResourceEntry(DictionaryEntry dictionaryEntry)
  {
    Key = dictionaryEntry.Key.ToString();
    Value = dictionaryEntry.Value != null ? dictionaryEntry.Value.ToString() : string.Empty;
  }
  public DictionaryEntry ToDictionaryEntry()
  {
    return new DictionaryEntry(Key, Value);
  }
}
```

سپس با استفاده از این کلاس خواهیم داشت:

```
private static List<ResXResourceEntry> Read(string filePath)
{
  using (var reader = new ResXResourceReader(filePath))
  {
    return reader.Cast<object>().Cast<DictionaryEntry>().Select(de => new
ResXResourceEntry(de)).ToList();
  }
}
```

حال این متد برای استفاده‌های آتی آماده است.

نوشتن در فایل resx.

برای نوشتن در یک فایل resx. میتوان از کلاس ResXResourceWriter استفاده کرد. این کلاس نیز در کتابخانه System.Windows.Forms در فایل System.Windows.Forms.dll قرار دارد:

System.Resources.ResXResourceWriter

متاسفانه در این کلاس امکان افزودن یا ویرایش یک کلید به تنهایی وجود ندارد. بنابراین برای ویرایش یا اضافه کردن حتی یک کلید کل فایل باید دوباره تولید شود. برای استفاده از این کلاس نیز میتوان به شکل زیر عمل کرد:

```
private static void Write(IEnumerable<ResXResourceEntry> resources, string filePath)
{
  using (var writer = new ResXResourceWriter(filePath))
  {
```

```
    foreach (var resource in resources)
    {
      writer.AddResource(resource.Key, resource.Value);
    }
  }
}
```

در متد فوق از همان کلاس ResXResourceEntry که در قسمت قبل معرفی شد، استفاده شده است. از متد زیر نیز میتوان برای حالت کلی حذف یا ویرایش استفاده کرد:

```
private static void AddOrUpdate(ResXResourceEntry resource, string filePath)
{
  var list = Read(filePath);
  var entry = list.SingleOrDefault(l => l.Key == resource.Key);
  if (entry == null)
  {
    list.Add(resource);
  }
  else
  {
    entry.Value = resource.Value;
  }
  Write(list, filePath);
}
```

در این متد از متدهای Read و Write که در بالا نشان داده شده‌اند استفاده شده است.

حذف یک کلید در فایل resx.

برای اینکار میتوان از متد زیر استفاده کرد:

```
private static void Remove(string key, string filePath)
{
  var list = Read(filePath);
  list.RemoveAll(l => l.Key == key);
  Write(list, filePath);
}
```

در این متد، از متد Write که در قسمت معرفی شد، استفاده شده است.

راه حل نهایی

قبل از بکارگیری روشهای معرفی شده در این مطلب بهتر است ابتدا یکسری قرارداد بصورت زیر تعریف شوند:

- طبق راهنماییهای موجود در قسمت قبل یک پروژه جداگانه با عنوان Resources برای نگهداری فایلهای resx. ایجاد شود.
- همواره آخرین نسخه از محتویات موردنیاز از پروژه Resources باید درون فولدری با عنوان Resources در پوشه App_Data قرار داشته باشد.
- آخرین نسخه تولیدی از محتویات موردنیاز پروژه Resource در فولدری با عنوان Defaults در مسیر App_Data\Resources برای فراهم کردن امکان "بازگرداندن به تنظیمات اولیه" وجود داشته باشد.

برای فراهم کردن این موارد بهترین راه حل استفاده از تنظیمات Post-build event command line است. اطلاعات بیشتر درباره Build Eventها در اینجا .

برای اینکار من از دستور xcopy استفاده کردم که نسخه توسعه یافته دستور copy است. دستورات استفاده شده در این قسمت عبارتند از:

```
xcopy $(ProjectDir)*.* $(SolutionDir)MvcApplication1\App_Data\Resources /e /y /i
/exclude:$(ProjectDir)excludes.txt
xcopy $(ProjectDir)*.* $(SolutionDir)MvcApplication1\App_Data\Resources\Defaults /e /y /i
/exclude:$(ProjectDir)excludes.txt
xcopy $(ProjectDir)$(OutDir)*.* $(SolutionDir)MvcApplication1\App_Data\Resources\Defaults\bin /e /y /i
```

در دستورات فوق آرگومان e/ برای کپی تمام فولدرها و زیرفولدرها، y/ برای تایید تمام کانفیرم ها، و i/ برای ایجاد خودکار فولدرهای موردنیاز استفاده میشود. آرگومان exclude/ نیز همانطور که از نامش پیداست برای خارج کردن فایلها و فولدرهای موردنظر از لیست کپی استفاده میشود. این آرگومان مسیر یک فایل متنی حاوی لیست این فایلها را دریافت میکند. در تصویر زیر یک نمونه از این فایل و مسیر و محتوای مناسب آن را مشاهده میکنید:

با استفاده از این فایل excludes.txt فولدرهای bin و obj و نیز فایلهای با پسوند user. و vspscc. (مربوط به TFS) و نیز خود فایل excludes.txt از لیست کپی دستور xcopy حذف میشوند و بنابراین کپی نمیشوند. درصورت نیاز میتوانید گزینه‌های دیگری نیز به این فایل اضافه کنید.

همانطور که در اینجا اشاره شده است، در تنظیمات Post-build event command line یکسری متغیرهای ازپیش تعریف شده (Macro) وجود دارند که از برخی از آنها در دستوارت فوق استفاده شده است:

$(ProjectDir) : مسیر کامل و مطلق پروژه جاری به همراه یک کاراکتر \ در انتها

$(SolutionDir) : مسیر کامل و مطلق سولوشن به همراه یک کاراکتر \ در انتها

$(OutDir) : مسیر نسبی فولدر Output پروژه جاری به همراه یک کاراکتر \ در انتها

نکته: این دستورات باید در Post-Build Event پروژه Resources افزوده شوند.

با استفاده از این تنظیمات مطمئن میشویم که پس از هر Build آخرین نسخه از فایلهای موردنیاز در مسیرهای تعیین شده کپی میشوند. درنهایت با استفاده از کلاس ResXResourceManager که در زیر آورده شده است، کل عملیات را ساماندهی میکنیم:

```
public class ResXResourceManager
{
  private static readonly object Lock = new object();
  public string ResourcesPath { get; private set; }
  public ResXResourceManager(string resourcesPath)
  {
    ResourcesPath = resourcesPath;
  }
  public IEnumerable<ResXResourceEntry> GetAllResources(string resourceCategory)
  {
    var resourceFilePath = GetResourceFilePath(resourceCategory);
    return Read(resourceFilePath);
  }
  public void AddOrUpdateResource(ResXResourceEntry resource, string resourceCategory)
  {
    var resourceFilePath = GetResourceFilePath(resourceCategory);
    AddOrUpdate(resource, resourceFilePath);
  }
  public void DeleteResource(string key, string resourceCategory)
  {
    var resourceFilePath = GetResourceFilePath(resourceCategory);
    Remove(key, resourceFilePath);
```

```
  }
  private string GetResourceFilePath(string resourceCategory)
  {
    var extension = Thread.CurrentThread.CurrentUICulture.TwoLetterISOLanguageName == "en" ? ".resx" :
".fa.resx";
    var resourceFilePath = Path.Combine(ResourcesPath, resourceCategory.Replace(".", "\\") +
extension);
    return resourceFilePath;
  }
  private static void AddOrUpdate(ResXResourceEntry resource, string filePath)
  {
    var list = Read(filePath);
    var entry = list.SingleOrDefault(l => l.Key == resource.Key);
    if (entry == null)
    {
      list.Add(resource);
    }
    else
    {
      entry.Value = resource.Value;
    }
    Write(list, filePath);
  }
  private static void Remove(string key, string filePath)
  {
    var list = Read(filePath);
    list.RemoveAll(l => l.Key == key);
    Write(list, filePath);
  }
  private static List<ResXResourceEntry> Read(string filePath)
  {
    lock (Lock)
    {
      using (var reader = new ResXResourceReader(filePath))
      {
        var list = reader.Cast<object>().Cast<DictionaryEntry>().ToList();
        return list.Select(l => new ResXResourceEntry(l)).ToList();
      }
    }
  }
  private static void Write(IEnumerable<ResXResourceEntry> resources, string filePath)
  {
    lock (Lock)
    {
      using (var writer = new ResXResourceWriter(filePath))
      {
        foreach (var resource in resources)
        {
          writer.AddResource(resource.Key, resource.Value);
        }
      }
    }
  }
}
```

در این کلاس تغییراتی در متدهای معرفی شده در قسمتهای بالا برای مدیریت دسترسی همزمان با استفاده از بلاک lock ایجاد شده است.

با استفاده از کلاس BuildManager عملیات تولید کتابخانه‌ها مدیریت میشود. (در مورد نحوه استفاده از MSBuild در اینجا توضیحات کافی آورده شده است):

```
public class BuildManager
{
  public string ProjectPath { get; private set; }
  public BuildManager(string projectPath)
  {
    ProjectPath = projectPath;
  }
  public void Build()
  {
    var regKey = Registry.LocalMachine.OpenSubKey(@"SOFTWARE\Microsoft\MSBuild\ToolsVersions\4.0");
    if (regKey == null) return;
    var msBuildExeFilePath = Path.Combine(regKey.GetValue("MSBuildToolsPath").ToString(),
"MSBuild.exe");
    var startInfo = new ProcessStartInfo
    {
      FileName = msBuildExeFilePath,
```

```
      Arguments = ProjectPath,
      WindowStyle = ProcessWindowStyle.Hidden
    };
    var process = Process.Start(startInfo);
    process.WaitForExit();
  }
}
```

درنهایت مثلا با استفاده از کلاس ResXResourceFileManager مدیریت فایلهای این کتابخانه‌ها صورت میپذیرد:

```
public class ResXResourceFileManager
{
  public static readonly string BinPath =
Path.GetDirectoryName(Assembly.GetExecutingAssembly().GetName().CodeBase.Replace("file:///", ""));
  public static readonly string ResourcesPath = Path.Combine(BinPath, @"..\App_Data\Resources");
  public static readonly string ResourceProjectPath = Path.Combine(ResourcesPath, "Resources.csproj");
  public static readonly string DefaultsPath = Path.Combine(ResourcesPath, "Defaults");
  public static void CopyDlls()
  {
    File.Copy(Path.Combine(ResourcesPath, @"bin\debug\Resources.dll"), Path.Combine(BinPath,
"Resources.dll"), true);
    File.Copy(Path.Combine(ResourcesPath, @"bin\debug\fa\Resources.resources.dll"),
Path.Combine(BinPath, @"fa\Resources.resources.dll"), true);
    Directory.Delete(Path.Combine(ResourcesPath, "bin"), true);
    Directory.Delete(Path.Combine(ResourcesPath, "obj"), true);
  }
  public static void RestoreAll()
  {
    RestoreDlls();
    RestoreResourceFiles();
  }
  public static void RestoreDlls()
  {
    File.Copy(Path.Combine(DefaultsPath, @"bin\Resources.dll"), Path.Combine(BinPath, "Resources.dll"),
true);
    File.Copy(Path.Combine(DefaultsPath, @"bin\fa\Resources.resources.dll"), Path.Combine(BinPath,
@"fa\Resources.resources.dll"), true);
  }
  public static void RestoreResourceFiles(string resourceCategory)
  {
    RestoreFile(resourceCategory.Replace(".", "\\"));
  }
  public static void RestoreResourceFiles()
  {
    RestoreFile(@"Global\Configs");
    RestoreFile(@"Global\Exceptions");
    RestoreFile(@"Global\Paths");
    RestoreFile(@"Global\Texts");

    RestoreFile(@"ViewModels\Employees");
    RestoreFile(@"ViewModels\LogOn");
    RestoreFile(@"ViewModels\Settings");

    RestoreFile(@"Views\Employees");
    RestoreFile(@"Views\LogOn");
    RestoreFile(@"Views\Settings");
  }

  private static void RestoreFile(string subPath)
  {
    File.Copy(Path.Combine(DefaultsPath, subPath + ".resx"), Path.Combine(ResourcesPath, subPath +
".resx"), true);
    File.Copy(Path.Combine(DefaultsPath, subPath + ".fa.resx"), Path.Combine(ResourcesPath, subPath +
".fa.resx"), true);
  }
}
```

در این کلاس از مفهومی با عنوان resourceCategory برای استفاده راحتتر در ویوها استفاده شده است که بیانگر فضای نام نسبی فایلهای Resource و کلاسهای متناظر با آنهاست که براساس استانداردها باید برطبق مسیر فیزیکی آنها در پروژه باشد مثل Global.Texts یا Views.LogOn. همچنین در متد RestoreResourceFiles نمونه هایی از مسیرهای این فایلها آورده شده است.

پس از اجرای متد Build از کلاس BuildManager، یعنی پس از build پروژه Resource در زمان اجرا، باید ابتدا فایلهای تولیدی به

مسیرهای مربوطه در فولدر bin برنامه کپی شده سپس فولدرهای تولیدشده توسط msbuild، حذف شوند. این کار در متد CopyDlls از کلاسResXResourceFileManager انجام میشود. هرچند در این قسمت فرض شده است که فایل csprj. موجود برای حالت debug تنظیم شده است.

نکته: دقت کنید که در این قسمت بلافاصله پس از کپی فایلها در مقصد با توجه به توضیحات ابتدای این مطلب سایت Restart خواهد شد که یکی از ضعفهای عمده این روش به شمار میرود.

سایر متدهای موجود نیز برای برگرداندن تنظیمات اولیه بکار میروند. در این متدها از محتویات فولدر Defaults استفاده میشود.

نکته : درصورت ساخت دوباره اسمبلی و یا بازگرداندن اسمبلی‌های اولیه، از آنجاکه وب‌سایت Restart خواهد شد، بنابراین بهتر است تا صفحه جاری بلافاصله پس از اتمام عملیات،دوباره بارگذاری شود. مثلا اگر از ajax برای اعمال این دستورات استفاده شده باشد میتوان با استفاده از کدی مشابه زیر در پایان فرایند صفحه را دوباره بارگذاری کرد:

```
window.location.reload();
```

در قسمت بعدی راه حل بهتری با استفاده از فراهم کردن پرووایدر سفارشی برای مدیریت فایلهای Resource ارائه میشود.

نظرات خوانندگان

نویسنده: بهمن خلفی
تاریخ: ۱۳۹۲/۰۲/۰۱ ۹:۲۲

با سلام خدمت شما
مطلب بسیار مفیدی است و امیدوارم ادامه دهید...

عنوان: **تولید SiteMap استاندارد و ایجاد یک ActionResult اختصاصی برای Return کردن SiteMap تولید شده**
نویسنده: سید مهران موسوی
تاریخ: ۱۳۹۲/۰۲/۲۹ ۲۱:۱۵
آدرس: www.dotnettips.info
گروه‌ها: ASP.Net MVC, SEO, sitemap, MVC

یکی از itemهای مهم در بهینه سازی SEO یک وب‌سایت وجود یک SiteMap استاندارد متشکل از لینک‌های موجود در سایت هست که در وب‌سایت‌های داینامیک معمولا این لینک‌ها بر اساس داده‌های موجود در بانک اطلاعاتی ایجاد میشه. برای مثال مطالب، اخبار و ....

در اینجا بنده قبلا یک کلاس برای تولید SiteMap آماده کردم که در پروژه‌های خودم ازش استفاده میکنم. توسط این کلاس میتونید به صورت داینامیک SiteMap وب‌سایت مبتنی بر ASP.NET MVC خودتون رو ایجاد کنید.

برای آشنایی با ساختار یک SiteMap استاندارد میتونید به لینک رسمی روبرو مراجعه کنید :
http://www.sitemaps.org/de/protocol.html

بنده کلاس‌های زیر رو بر مبنای لینک مذکور در سایت رسمی SiteMaps تولید کردم. بعد از تولید SiteMap نیاز دارید که اون رو مثلا به عنوان خروجی یک ActionResult بازگردونید. برای این کار هم یک کلاس با نام XmlResult مشتق شده از ActionResult آماده سازی کردم که کلاس تولید شده SiteMap رو Serialize میکنه و به عنوان نتیجه‌ی یک Action باز می‌گردونه .

```
using System;
using System.Collections;
using System.Web.Mvc;
using System.Xml.Serialization;

namespace Neoox.Core.SeoTools
{
    [XmlRoot("urlset", Namespace = "http://www.sitemaps.org/schemas/sitemap/0.9")]
    public class Sitemap
    {
        private ArrayList map;

        public Sitemap()
        {
            map = new ArrayList();
        }

        [XmlElement("url")]
        public Location[] Locations
        {
            get
            {
                Location[] items = new Location[map.Count];
                map.CopyTo(items);
                return items;
            }
            set
            {
                if (value == null)
                    return;
                Location[] items = (Location[])value;
                map.Clear();
                foreach (Location item in items)
                    map.Add(item);
            }
        }

        public int Add(Location item)
        {
            return map.Add(item);
        }
    }

    public class Location
    {
        public enum eChangeFrequency
        {
            always,
            hourly,
```

```
            daily,
            weekly,
            monthly,
            yearly,
            never
        }

        [XmlElement("loc")]
        public string Url { get; set; }

        [XmlElement("changefreq")]
        public eChangeFrequency? ChangeFrequency { get; set; }
        public bool ShouldSerializeChangeFrequency() { return ChangeFrequency.HasValue; }

        [XmlElement("lastmod")]
        public DateTime? LastModified { get; set; }
        public bool ShouldSerializeLastModified() { return LastModified.HasValue; }

        [XmlElement("priority")]
        public double? Priority { get; set; }
        public bool ShouldSerializePriority() { return Priority.HasValue; }
    }

    public class XmlResult : ActionResult
    {
        private object objectToSerialize;

        public XmlResult(object objectToSerialize)
        {
            this.objectToSerialize = objectToSerialize;
        }

        public object ObjectToSerialize
        {
            get { return this.objectToSerialize; }
        }

        public override void ExecuteResult(ControllerContext context)
        {
            if (this.objectToSerialize != null)
            {
                context.HttpContext.Response.Clear();
                var xs = new System.Xml.Serialization.XmlSerializer(this.objectToSerialize.GetType());
                context.HttpContext.Response.ContentType = "text/xml";
                xs.Serialize(context.HttpContext.Response.Output, this.objectToSerialize);
            }
        }
    }
}
```

و اما نحوه استفاده از این کلاس‌ها هم خیلی سادست. به مثال زیر توجه کنید ... فقط این نکته رو در نظر داشته باشید که item هایی که به sitemap اضافه میشه واکشی شده از بانک اطلاعاتی هست، در این مثال چون خواستم ساده توضیح داده بشه نحوه استفاده از این کلاس‌ها، این داده‌ها به صورت static در نظر گرفته شد ولی شما میتونید داده‌ها رو بر اساس ساختار بانک اطلاعاتی خودتون واکشی کرده و به SiteMap اضافه کنید تا یک SiteMap کاملا پویا و Dynamic داشته باشید...

```
public ActionResult Sitemap()
        {
            Sitemap sm = new Sitemap();
            sm.Add(new Location()
            {
                Url = string.Format("http://www.TechnoDesign.ir/Articles/{0}/{1}", 1, "SEO-in-ASP.NET-
MVC"),
                LastModified = DateTime.UtcNow,
                Priority = 0.5D
            });
            return new XmlResult(sm);
        }
```

نظرات خوانندگان

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۳/۰۱ ۱۴:۳۹

یک نکته تکمیلی:

ایجاد خودکار آدرس sitemap.xml از روی آدرس کنترلر SiteMap با افزودن مسیریابی مرتبط با آن: (در حقیقت ایجاد آدرس، برای فایلی که وجود خارجی ندارد و نگاشت آن به یک کنترلر و اکشن متد خاصی در آن)

```
routes.MapRoute(
                "SiteMap_route", // Route name
                "sitemap.xml", // URL with parameters
                new { controller = "Sitemap", action = "index", name = UrlParameter.Optional, area = ""
} // Parameter defaults
            );
```

نویسنده: سید مهران موسوی
تاریخ: ۱۳۹۲/۰۳/۰۱ ۱۴:۵۶

این کار که 100% باید انجام بشه وگرنه بی فایدست تمام ماجرا . ممنون از آقای نصیری بابت پیوست این نکته. دوستان حتما توجه داشته باشن ....

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۳/۰۱ ۱۶:۴۱

```
in robots.txt file:
Sitemap: http://www.site.com/Sitemap
```

علاوه بر نکته فوق باید در فایل robots.txt سایت هم یک سطر با آدرس مطلق (و نه نسبی) به مسیر سایت‌مپ اشاره کنه تا موتورهای جستجو بدونند کجا رو باید بررسی کنند

نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۲/۰۵/۲۸ ۲۲:۱۹

با سلام؛ چند سوال داشتم:

1- به نظرتون اگر تعداد صفحات وب سایت زیاد باشه این کار (ساخت sitemap به ازای هر درخواست) رو سرور فشار ایجاد نمی‌کنه؟

2 -همان طور که می‌دونید حداکثر می‌شود 50000 هزار لینک در یک sitemap قرار داد و به ازای مقادیر بیشتر باید چندین sitemap ایجاد کرد. http://googlewebmastercentral.blogspot.nl/2006/10/multiple-sitemaps-in-same-directory.html

برای این سناریو چطور می‌شه رفتار کرد؟

3- و سوال آخر اینکه چرا آقای نصیری لینک‌های محدودی را در sitemap سایت قرار داده اند؟ دلیل خاصی داشته است؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۵/۲۸ ۲۲:۲۸

1 - از OutputCache حتما باید استفاده کنید.

2 و 3 - موتورهای جستجو به دنبال لینک‌های جدید در سایت شما هستند. زمانیکه موارد قبلی ایندکس شدند و در بانک اطلاعاتی آن‌ها موجودند، فقط موارد جدیدتر را در سایت مپ قرار بدید. وجود موارد قدیمی ایندکس شده مهم نیستند و مدام چک خواهند شد (جهت بررسی صحت اطلاعات بانک اطلاعاتی موتورهای جستجو).

این مورد را اگر خواستید بررسی کنید، از ابزار وب مستر گوگل بهره بگیرید. هر روز آمار می‌دهد، چه مواردی کلا ایندکس شده

(همچنین چه تعدادی دقیقا از طریق سایت مپ ایندکس شده)، چه مشکلاتی داشته، موارد قدیمی که در بانک اطلاعاتی آن هست و در سایت شما یافت نشده، کدام‌ها هستند و غیره.

نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۲/۰۵/۲۹ ۱۸:۲۲

تشکر بابت پاسختون 2 - سایتی که من بر روی وب می‌گذارم با حدود یک میلیون صفحه در وب قرار می‌گیرد! (و قابل ذکر هست که بیش از 50000 لینک در سایت مپ ممکن نیست) بر اساس مواردی که شما بیان کردید، باید هر روز مقداری از این لینک‌ها را در سایت مپ قرار بدهم و بعد از ایندکس شدن توسط گوگل آن‌ها را تغییر دهم که عملا برای این حجم بالا ممکن نیست.
می تونم بپرسم جرا با ساختن فایل‌های فیزیکی برای سایت مپ موافق نیستید؟
(البته در شرایط صورت مسئله ای که بیان کردم) در این صورت می‌شود طوری برنامه ریزی کرد که این فایل ها(به دلیل بالا بودن تعداد لینک از چندین فایل استفاده شود) در بازه زمانی مشخص دوباره باز سازی شوند و هیچ بار اضافه ای برای ایجاد هرباره آن‌ها به وجود نیاید.

باز هم تشکر بابت وقتی که می‌گذارید...

نویسنده: امیرحسین مرجانی
تاریخ: ۱۳۹۲/۰۵/۲۹ ۱۸:۴۲

و البته سوال دیگه هم که داشتم اینکه:
من از همین Routing که شما توی کامنت اول بیان کردید استفاده کردم. (که طبیعتا درسته) ولی توی پروژه بنده IIS ایرور می‌ده که این فایل (sitemap.xml)رو پیدا نکرده! فکر می‌کنم عمل routing به درستی انجام نمی‌شه.
من از mvc4 در vs2102 استفاده می‌کنم.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۵/۲۹ ۱۸:۴۵

کش کردن یک روش است برای کاهش ساخت خروجی پویا به ازای هر درخواست (سؤال مطرح شده) و از آن برداشت نمی‌شود

که روش‌های دیگر خوب هستند یا خیر.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۵/۲۹ ۱۸:۵۷

نکات مطلب « تنظیمات ثانویه پس از فعال سازی RouteExistingFiles » احتمالا مفید خواهد بود.

نویسنده: آیمو
تاریخ: ۱۳۹۲/۰۸/۰۴ ۱۶:۳۷

منم رفتم اون مطالبو خوندم ولی چیزی دستگیرم نشد .و هنوز همین مشکلو داردم. نمیشه بیشتر راهنمایی کنین؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۸/۰۴ ۱۶:۵۰

Route سفارشی زمانی عمل می‌کنه که قبل از Route پیش فرض تعریف شود. route پیش فرض اصطلاحا catch all است و نوبت به بقیه نخواهد رسید.

نویسنده: آیمو
تاریخ: ۱۳۹۲/۰۸/۰۴ ۱۷:۵۹

ایییییییییییییول...! خیلی باحالین. دقیقا مشکل همین بود . ممنون.

نویسنده: مهرسا
تاریخ: ۱۳۹۲/۱۱/۲۳ ۱۰:۲۳

سلام امکان استفاده این در ASP.NET Web forms هست چون من میخوام این کارو با استفاده از یک httphandler بکنم ؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۱/۲۳ ۱۰:۲۸

بله. مفاهیم و کدهای XmlResult تهیه شده را مستقیما در یک فایل ashx هم می‌شود بکار برد. context مورد استفاده، قابل انتقال است.

نویسنده: محسن
تاریخ: ۱۳۹۳/۰۲/۰۵ ۱۴:۳۹

با سلام. آقای نصیری ممنون میشم در مورد این نکته ای که فرمودید بیشتر توضیح بدید.
ممنونم

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۲/۰۵ ۱۷:۹

- توضیحات سیستم مسیریابی در ASP.NET MVC
- روش دیباگ سیستم مسیریابی

نویسنده: مجید
تاریخ: ۱۳۹۳/۰۵/۲۲ ۱۵:۳۳

سلام .خیلی ممنون به خاطر مطالبتون.
میخواستم بدونم در چه زمان و در کجا باید این Action رو صدا بزنیم.
با تشکر

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۵/۲۲ ۱۵:۳۹

موتورهای جستجو بر اساس تنظیمات فایل robots.txt قرار است آن‌را یافته و صدا بزنند. کاربرد شخصی ندارد.

نویسنده: حمید حسین وند
تاریخ: ۱۳۹۳/۰۶/۳۱ ۲۲:۲۶

سلام
چه جوری میشه از این کلاس توی asp.net webform استفاده کرد؟
اگه میشه نحوه استفاده از این کدها رو توی webform بذارید. ممنون

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۶/۳۱ ۲۳:۰

اکثر کدهای آن یکی است منهای کلاس XmlResult آن.
برای پیاده سازی و جایگزینی آن در وب فرم‌ها یک Generic handler استاندارد را اضافه کنید و در متد ProcessRequest آن، کدهای متد ExecuteResult فوق را قرار دهید.

```
public void ProcessRequest (HttpContext context)
```

البته چون در اینجا HttpContext را مستقیما دارید، قسمت‌های context.HttpContext متد ExecuteResult می‌شوند context.
Sitemap_ashx.cs

نویسنده: حسین
تاریخ: ۱۳۹۳/۰۸/۱۹ ۱۲:۱۰

با تشکر
آیا راهی وجود داره بدون تعریف کردن route بتوان به اکشن مورد نظر دسترسی پیدا کرد؟
چون برای فعال کردن این قابلیت باید اجازه وارد کردن dot در url رو داد که برامون امکان پذیر نیست

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۸/۱۹ ۱۲:۱۵

کمی بالاتر در نظرات عنوان شده، موتورهای جستجو این مسیر را (هر آدرسی با هر شکل یا نامی که دارد؛ مهم نیست) صرفا از فایل robots.txt دریافت می‌کنند.

عنوان: **نحوه ایجاد یک تصویر امنیتی (Captcha) با حروف فارسی در ASP.Net MVC**
نویسنده: پژمان پارسائی
تاریخ: ۱۳۹۲/۰۳/۰۳ ۱۷:۵
آدرس: www.dotnettips.info
گروه‌ها: MVC, ASP.Net MVC, CAPTCHA, Security

در این مطلب، سعی خواهیم کرد تا همانند تصویر امنیتی این سایت که موقع ورود نمایش داده می‌شود، یک نمونه مشابه به آنرا در ASP.Net MVC ایجاد کنیم. ذکر این نکته ضروری است که قبلا آقای پایروند در یک مطلب دو قسمتی کاری مشابه را انجام داده بودند، اما در مطلبی که در اینجا ارائه شده سعی کرده ایم تا تفاوتهایی را با مطلب ایشان داشته باشد.

همان طور که ممکن است بدانید، اکشن متدها در کنترلرهای MVC می‌توانند انواع مختلفی را برگشت دهند که شرح آن در مطالب این سایت به مفصل گذشته است. یکی از این انواع، نوع ActionResult می‌باشد. این یک کلاس پایه برای انواع برگشتی توسط اکشن متدها مثل FileResult، JsonResult می‌باشد. (اطلاعات بیشتر را اینجا بخوانید) اما ممکن است مواقعی پیش بیاید که بخواهید نوعی را توسط یک اکشن متد برگشت دهید که به صورت توکار تعریف نشده باشد. مثلا زمانی را در نظر بگیرید که بخواهید یک تصویر امنیتی را برگشت دهید. یکی از راه حل‌های ممکن به این صورت است که کلاسی ایجاد شود که از کلاس پایه ActionResult ارث بری کرده باشد. بدین صورت:

```
using System;
using System.Web.Mvc;

namespace MVCPersianCaptcha.Models
{
    public class CaptchaImageResult : ActionResult
    {
        public override void ExecuteResult(ControllerContext context)
        {
            throw new NotImplementedException();
        }
    }
}
```

همان طور که مشاهده می‌کنید، کلاسی به اسم CaptchaImageResult تعریف شده که از کلاس ActionResult ارث بری کرده است. در این صورت باید متد ExecuteResult را override کنید. متد ExecuteResult به صورت خودکار هنگامی که از CaptchaImageResult به عنوان یک نوع برگشتی اکشن متد استفاده شود اجرا می‌شود. به همین خاطر باید تصویر امنیتی توسط این متد تولید شود و به صورت جریان (stream) برگشت داده شود

کدهای اولیه برای ایجاد یک تصویر امنیتی به صورت خیلی ساده از کلاس‌های فراهم شده توسط GDI+ ، که در دات نت فریمورک وجود دارند استفاده خواهند کرد. برای این کار ابتدا یک شیء از کلاس Bitmap با دستور زیر ایجاد خواهیم کرد:

```
// Create a new 32-bit bitmap image.
Bitmap bitmap = new Bitmap(width, height, PixelFormat.Format32bppArgb);
```

پارامترهای اول و دوم به ترتیب عرض و ارتفاع تصویر امنیتی را مشخص خواهند کرد و پارامتر سوم نیز فرمت تصویر را بیان کرده است. Format32bppArgb یعنی یک تصویر که هر کدام از پیکسل‌های آن 32 بیت فضا اشغال خواهند کرد ، 8 بیت اول میزان آلفا، 8 بیت دوم میزان رنگ قرمز، 8 بیت سوم میزان رنگ سبز، و 8 تای آخر نیز میزان رنگ آبی را مشخص خواهند کرد

سپس شیئی از نوع Graphics برای انجام عملیات ترسیم نوشته‌های فارسی روی شیء bitmap ساخته می‌شود:

```
// Create a graphics object for drawing.
Graphics gfxCaptchaImage = Graphics.FromImage(bitmap);
```

خصوصیات مورد نیاز ما از gfxCaptchaImage را به صورت زیر مقداردهی می‌کنیم:

```
gfxCaptchaImage.PageUnit = GraphicsUnit.Pixel;
gfxCaptchaImage.SmoothingMode = SmoothingMode.HighQuality;
gfxCaptchaImage.Clear(Color.White);
```

واحد اندازه گیری به پیکسل، کیفیت تصویر تولید شده توسط دو دستور اول، و در دستور سوم ناحیه ترسیم با یک رنگ سفید پاک می‌شود.

سپس یک عدد اتفاقی بین 1000 و 9999 با دستور زیر تولید می‌شود:

```
// Create a Random Number from 1000 to 9999
int salt = CaptchaHelpers.CreateSalt();
```

متد CreateSalt در کلاس CaptchaHelpers قرار گرفته است، و نحوه پیاده سازی آن بدین صورت است:

```
public int CreateSalt()
{
   Random random = new Random();
   return random.Next(1000, 9999);
}
```

سپس مقدار موجود در salt را برای مقایسه با مقداری که کاربر وارد کرده است در session قرار می‌دهیم:

```
HttpContext.Current.Session["captchastring"] = salt;
```

سپس عدد اتفاقی تولید شده باید تبدیل به حروف شود، مثلا اگر عدد 4524 توسط متد CreateSalt تولید شده باشد، رشته "چهار هزار و پانصد و بیست و چهار" معادل آن نیز باید تولید شود. برای تبدیل عدد به حروف، آقای نصیری کلاس خیلی خوبی نوشته اند که چنین کاری را انجام می‌دهد. ما نیز از همین کلاس استفاده خواهیم کرد:

```
string randomString = (salt).NumberToText(Language.Persian);
```

در دستور بالا، متد الحاقی NumberToText با پارامتر Language.Persian وظیفه تبدیل عدد salt را به حروف فارسی معادل خواهد داشت.

به صورت پیش فرض نوشته‌های تصویر امنیتی به صورت چپ چین نوشته خواهند شد، و با توجه به این که نوشته ای که باید در تصویر امنیتی قرار بگیرد فارسی است، پس بهتر است آنرا به صورت راست به چپ در تصویر بنویسیم، بدین صورت:

```
// Set up the text format.
var format = new StringFormat();
int faLCID = new System.Globalization.CultureInfo("fa-IR").LCID;
format.SetDigitSubstitution(faLCID, StringDigitSubstitute.National);
format.Alignment = StringAlignment.Near;
format.LineAlignment = StringAlignment.Near;
format.FormatFlags = StringFormatFlags.DirectionRightToLeft;
```

و همچنین نوع و اندازه فونت که در این مثال tahoma می‌باشد:

```
// Font of Captcha and its size
Font font = new Font("Tahoma", 10);
```

خوب نوشته فارسی اتفاقی تولید شده آماده ترسیم شدن است، اما اگر چنین تصویری تولید شود احتمال خوانده شدن آن توسط روبات‌های پردازش گر تصویر شاید زیاد سخت نباشد. به همین دلیل باید کاری کنیم تا خواندن این تصویر برای این روبات‌ها سخت‌تر شود، روش‌های مختلفی برای این کار وجود دارند: مثل ایجاد نویز در تصویر امنیتی یا استفاده از توابع ریاضی سینوسی و

کسینوسی برای نوشتن نوشته‌ها به صورت موج. برای این کار اول یک مسیر گرافیکی در تصویر یا موج اتفاقی ساخته شود و به شیء gfxCaptchaImage نسبت داده شود. برای این کار اول نمونه ای از روی کلاس GraphicsPath ساخته می‌شود،

```
// Create a path for text
GraphicsPath path = new GraphicsPath();
```

و با استفاده از متد AddString ، رشته اتفاقی تولید شده را با فونت مشخص شده، و تنظیمات اندازه دربرگیرنده رشته مورد نظرر، و تنظیمات فرمت بندی رشته را لحاظ خواهیم کرد.

```
path.AddString(randomString,
                font.FontFamily,
                (int)font.Style,
                (gfxCaptchaImage.DpiY * font.SizeInPoints / 72),
                new Rectangle(0, 0, width, height), format);
```

با خط کد زیر شیء path را با رنگ بنقش با استفاده از شیء gfxCaptchaImage روی تصویر bitmap ترسیم خواهیم کرد:

```
gfxCaptchaImage.DrawPath(Pens.Navy, path);
```

برای ایجاد یک منحنی و موج از کدهای زیر استفاده خواهیم کرد:

```
//-- using a sin ware distort the image
int distortion = random.Next(-10, 10);
using (Bitmap copy = (Bitmap)bitmap.Clone())
{
          for (int y = 0; y < height; y++)
          {
              for (int x = 0; x < width; x++)
              {
                  int newX = (int)(x + (distortion * Math.Sin(Math.PI * y / 64.0)));
                  int newY = (int)(y + (distortion * Math.Cos(Math.PI * x / 64.0)));
                  if (newX < 0 || newX >= width) newX = 0;
                 if (newY < 0 || newY >= height) newY = 0;
                 bitmap.SetPixel(x, y, copy.GetPixel(newX, newY));
              }
         }
 }
```

موقع ترسیم تصویر امنیتی است:

```
//-- Draw the graphic to the bitmap
gfxCaptchaImage.DrawImage(bitmap, new Point(0, 0));

gfxCaptchaImage.Flush();
```

تصویر امنیتی به صورت یک تصویر با فرمت jpg به صورت جریان (stream) به مرورگر باید فرستاده شوند:

```
HttpResponseBase response = context.HttpContext.Response;
response.ContentType = "image/jpeg";
bitmap.Save(response.OutputStream, ImageFormat.Jpeg);
```

و در نهایت حافظه‌های اشغال شده توسط اشیاء فونت و گرافیک و تصویر امنیتی آزاد خواهند شد:

```
// Clean up.
font.Dispose();
gfxCaptchaImage.Dispose();
bitmap.Dispose();
```

برای استفاده از این کدها، اکشن متدی نوشته می‌شود که نوع CaptchaImageResult را برگشت می‌دهد:

```
public CaptchaImageResult CaptchaImage()
{
     return new CaptchaImageResult();
}
```

اگر در یک View خصیصه src یک تصویر به آدرس این اکشن متد مقداردهی شود، آنگاه تصویر امنیتی تولید شده نمایش پیدا می‌کند:

```
<img src="@Url.Action("CaptchaImage")"/>
```

بعد از پست کردن فرم مقدار text box تصویر امنیتی خوانده شده و با مقدار موجود در session مقایسه می‌شود، در صورتی که یکسان باشند، کاربر می‌تواند وارد سایت شود (در صورتی که نام کاربری یا کلمه عبور خود را درست وارد کرده باشد) یا اگر از این captcha در صفحات دیگری استفاده شود عمل مورد نظر می‌تواند انجام شود. در مثال زیر به طور ساده اگر کاربر در کادر متن مربوط به تصویر امنیتی مقدار درستی را وارد کرده باشد یا نه، پیغامی به او نشان داده می‌شود.

```
[HttpPost]
public ActionResult Index(LogOnModel model)
{
      if (!ModelState.IsValid) return View(model);

      if (model.CaptchaInputText == Session["captchastring"].ToString())
            TempData["message"] = "تصویر امنتی را صحیح وارد کرده اید";
      else
            TempData["message"] = "تصویر امنیتی را اشتباه وارد کرده اید";

      return View();
}
```

کدهای کامل مربوط به این مطلب را به همراه یک مثال از لینک زیر دریافت نمائید:

MVC-Persian-Captcha.zip

نظرات خوانندگان

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۳/۰۳ ۱۹:۳۶

ممنون. میشه قسمت بررسی نهایی در اکشن متد رو هم کپسوله کرد (چیزی شبیه به امکانات AOP سرخود در MVC). مثلا یک ویژگی جدید به نام ValidateCaptcha درست کرد که به اکشن متد اعمال شود و کار بررسی صحت اطلاعات ورودی مخصوص Captcha رو انجام و نهایتا اطلاعات ModelState رو بر اساس اطلاعات ورودی و Session ایی که در اینجا تعریف شده، به روز کنه:

```
[AttributeUsage(AttributeTargets.Method, AllowMultiple = false)]
    public sealed class ValidateCaptchaAttribute : ActionFilterAttribute
    {
        public override void OnActionExecuting(ActionExecutingContext filterContext)
        {
            var controllerBase = filterContext.Controller;

            var captchaInputTextProvider = controllerBase.ValueProvider.GetValue("CaptchaInputText");
            if (captchaInputTextProvider == null)
            {
                controllerBase.ViewData.ModelState.AddModelError("CaptchaInputText", "لطفا تصویر امنیتی
را وارد کنید");
                base.OnActionExecuting(filterContext);
                return;
            }
            var inputText = captchaInputTextProvider.AttemptedValue;

            if (inputText != Session["captchastring"].ToString())
               controllerBase.ViewData.ModelState.AddModelError("CaptchaInputText", "تصویر امنیتی را
اشتباه وارد کرده اید");
        }
    }
```

به این صورت (با استفاده از ویژگی فوق) همان بررسی متداول ModelState.IsValid در یک اکشن متد کافی خواهد بود.

نویسنده: پژمان پارسائی
تاریخ: ۱۳۹۲/۰۳/۰۴ ۰:۲۷

بسیار عالی و سازنده، خیلی نکته مفیدی بود

نویسنده: میثم خوشقدم
تاریخ: ۱۳۹۲/۰۳/۰۴ ۱۳:۵۷

ممنون - مطلب کاربردی و مفیدی بود

شخصا این Captcha را به انواع دیگر ترجیح میدم چرا که:

1- شخصی سازی شده

2- ارجاع به سایت واسطی نداره

3- هیچ گونه سرباری نداره

4- اعمال هر گونه تغییر و اصلاح مناسب با فرهنگ و کشور و یا حتی نوع سایت قابل انجام است

بازهم تشکر می‌کنم

نویسنده: میثم خوشقدم
تاریخ: ۱۳۹۲/۰۳/۰۴ ۱۵:۱۷

سلام

اگر کد شما را گسترش دهیم و به جای پیغام‌های خطای ارسال شده، مقادیر Attributeها یا متادیتاهایی مثل Required را خواند و مستقیم پیغامی به کنترل و ModelState پاس نداد، فکر می‌کنم کامل‌تر باشه .

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۳/۰۴ ۱۶:۳۸

حالت کاملتر زمانی است که در سازنده کلاس ویژگی، این خطاها رو دریافت کنید و به کاربر اجازه بدید، پیغام‌های دلخواهی رو تنظیم کنه. چون Required فقط یک حالت است. حالت بعدی عدم تساوی مقدار ورودی با مقدار اصلی Captcha است که نیاز به پیغام دومی دارد.

نویسنده: پژمان پارسائی
تاریخ: ۱۳۹۲/۰۳/۰۵ ۸:۴۱

خواهش می‌کنم دوست عزیز. لطف دارید

موارد بیشتری که می‌توان به آن افزود:
استفاده از فونت‌ها به صورت اتفاقی برای نوشتن متن تصویر امنیتی،
دادن رنگ‌ها به متن و پس زمینه به صورت اتفاقی به تصویر امنیتی،
معتبر نبودن session نگهدارنده عدد تصویر امنیتی بعد از یک میزان مشخص،
و غیره

نویسنده: حسینی
تاریخ: ۱۳۹۲/۰۳/۰۶ ۱۹:۴

سلام و مرسی از آموزش خوب و کاربردیتون
اما یکی از گزینه‌های چک لیست برنامه‌های ASP.NET MVC که آقای نصیری فرمودند عدم استفاده از سشن در برنامه‌ها بود و استفاده از کوکی‌های رمزنگاری شده پیشنهاد شده بود.

نویسنده: پژمان پارسائی
تاریخ: ۱۳۹۲/۰۳/۰۶ ۲۱:۲

سلام. ممنون از نظر شما
چک لیست رو مطالعه کردم و عملا هم استفاده کردم.
استفاده از سشن راحت‌تر هست و به همین دلیل اون رو انتخاب کردم. نسخه بعدی این تصویر امنیتی رو با استفاده از نظرات همین مطلب ارائه می‌کنم.

نویسنده: پژمان پارسائی
تاریخ: ۱۳۹۲/۰۳/۱۵ ۶:۲۸

سلام
نسخه بعدی این تصویر امنیتی رو از لینک زیر دریافت نمائید:
MVCPersianCaptcha-2.zip

امکاناتی که اضافه کردم:

- استفاده از کوکی رمزنگاری شده جهت ذخیره کردن مقدار عدد معادل تصویر امنیتی
- اضافه کردن ویژگی ValidateCaptcha جهت تعیین اعتبار کوکی و مقداری که کاربر وارد کرده
- اضافه کردن نویزهای اتفاقی
- تعیین یک میزان 30 ثانیه ای (قابل تغییر است) جهت معتبر بودن مقدار ارسالی توسط کاربر
- ایجاد قابلیت تازه سازی (refresh) تصویر امنیتی
- تغییر کلید رمزنگاری و رمزگشایی اطلاعات به ازاء هر روز

و غیره

نویسنده: پژمان پارسائی
تاریخ: ۱۳۹۲/۰۳/۲۱ ۲۲:۵۲

پوشه‌های packages، bin، و obj از نسخه‌های اول و دوم این تصویر امنیتی حذف شده اند تا حجم فایلهای مربوطه کمتر شوند. به همین دلیل، بعد از اجرای هر کدام از این برنامه‌ها خطایی صادر می‌شود مبنی بر این که بسته ای به اسم Newtonsoft.Json در پروژه وجود ندارد. لطفا برای حل این مشکل به فایل Global.asax.cs مراجعه کنید و خط کد زیر را از آن حذف کنید: (این خط کد در این پروژه غیر ضروری است و نیازی به آن نیست)

```
WebApiConfig.Register(GlobalConfiguration.Configuration);
```

روش دیگر برای حل این نوع مشکلات، در مطلب بازسازی کامل پوشه packages بسته‌های NuGet به صورت خودکار بیان شده است.

نویسنده: بهمن خلفی
تاریخ: ۱۳۹۲/۰۴/۱۱ ۱۰:۳۴

با سلام و تشکر از همه دوستان

بنده در اولین استفاده یک تغییر داشتم که خواستم بقیه هم از آن استفاده کنند:

```
public ValidateCaptchaAttribute()
        {
            ErrorWasHappened = "خطایی اتفاق افتاده است";
            CaptchaCodeIsRequired = "لطفا کد امنیتی را وارد کنید";
            CaptchaCodeIsIncorrect = "کد امنیتی را اشتباه وارد کرده اید";
            CookieMustEnabled = "باید ابتدا قابلیت کوکی‌ها را در مرورگر خود فعال کنید";
            ExpireTimeCaptchaCodeBySeconds = 60;
            TimeIsExpired = string.Format("حداکثر مهلت وارد کردن کد امنیتی {0} ثانیه
است",ExpireTimeCaptchaCodeBySeconds.ToString());

        }
```

باز ممنون!

نویسنده: ایلیا اکبری فرد
تاریخ: ۱۳۹۲/۰۴/۱۴ ۲۱:۳۸

با سلام.

سپاس از مقاله خوبتون.

در کلاینت همه چی درست کار میکند ولی وقتی آنرا درون هاست واقعی publish میکنم، تصویر نمایش داده نمی‌شود و خطایی که توسط elmah لاگ می‌شود به صورت زیر است:

```
The system cannot find the file specified " source="mscorlib"
detail="System.Security.Cryptography.CryptographicException: The system cannot find the file specified.
```

با تشکر.

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۴/۱۴ ۲۳:۲

```
<system.web>
 <trust level="Medium" originUrl=".*" />
```

شاید هاست شما مدیوم تراست هست. برای تست روی لوکال این تنظیم بالا رو به وب کانفیگ اضافه کنید تا خطا رو بتونید لوکال دیباگ کنید.

نویسنده: ایلیا اکبری فرد
تاریخ: ۱۳۹۲/۰۴/۱۴ ۲۳:۱۶

سپاس از پاسخ شما.

ولی با اضافه کردن آن ، همواره خطای زیر رخ می‌دهد:

```
The application attempted to perform an operation not allowed by the security policy.  To grant this
application the required permission please contact your system administrator or change the
application's trust level in the configuration file
```

نویسنده: پژمان پارسائی
تاریخ: ۱۳۹۲/۰۴/۱۵ ۶:۷

سلام، از نظر شما متشکرم

همون طور که آقای محسن خان گفت، احتمالا هاست شما medium trust هست. اما رو کامپیوتر خودتون full trust برنامه نویسی می‌کنید.

چون هویت کاربر هنوز مشخص نشده پیغامی مبنی بر این لاگ میشه که اسمبلی mscorlib وجود نداره. در واقع وجود داره ولی نه برای کاربر anonymous ! همچنین این به دلیل medium trust بودن هم میتونه باشه. برای حل این مشکل کارهای زیر رو انجام بدین:

1) به فایل web.config برین و کد زیر رو اضافه کنید:

```
<trust level="Full" originUrl=".*" />
```

2) باید تغییری رو در متدهای الحاقی Encrypt و Decrypt بدین، که از این متدها برای رمزنگاری و رمزگشایی محتوای کوکی تصویر امنیتی استفاده میشه. قبل از هر کدوم از این متدها flag زیر رو اضافه کنید:

```
[System.Security.Permissions.PermissionSet(System.Security.Permissions.SecurityAction.Assert,
Unrestricted = true)]
```

همچنین، یک خط داخل بدنه هر کدوم از متدهای الحاقی Encrypt و Decrypt هست، منظورم این خط کد هست:

```
var cspp = new CspParameters { KeyContainerName = key };
```

که باید به خط کد زیر تبدیل بشه:

```
var cspp = new CspParameters { KeyContainerName = key, Flags = CspProviderFlags.UseMachineKeyStore };
```

نویسنده: ایلیا اکبری فرد
تاریخ: ۱۳۹۲/۰۴/۱۵ ۹:۷

بله کار کرد. متشکرم.

نویسنده: ایلیا اکبری فرد
تاریخ: ۱۳۹۲/۰۴/۲۹ ۱۸:۲۹

با سلام.
برای اولین بار که فرمی لود می‌شود همواره حتی با وارد کردن مقدار صحیح متن کپچا باز هم خطای نادرست بودن می‌دهد. ولی وقتی یکبار بر روی رفرش مربوط به کپچا کلیک میکنم تا کپچای جدیدی تولید کند ، درست کار می‌کند؟

نویسنده: پژمان پارسائی
تاریخ: ۱۳۹۲/۰۵/۰۱ ۲۱:۵۱

سلام دوست عزیز
بابت تاخیر بوجود آمده در پاسخ دهی عذرخواهی می‌کنم.

من همین کپچا رو دارم استفاده می‌کنم و این مشکلو هم باهاش ندارم. یک حدسی که می‌زنم اینه که شاید کوکی بار اول درست مقداردهی نشه. لطفا کل عملیات از تشکیل تصویر امنیتی، تا مقداری که درون کوکی قرار می‌گیره، و بعد رمز گزاری میشه ، تا مقداری که درون کوکی مرورگر ذخیره میشه (برای مشاهده کوکی‌های مرورگر از این روش می‌تونید استفاده کنید) . سپس ببینید بعد از پست شدن فرم حاوی کپچا به سرور مقداری که از درون کوکی خونده میشه و می‌خواد decrypt بشه با مقداری که از اول ذخیره شده یکسان باشه.

امیدوارم مشکلتون حل شه دوست عزیز.

نویسنده: کوروش ت
تاریخ: ۱۳۹۲/۰۹/۲۸ ۱۰:۳

باسلام، این مطلب خیلی کاربردی و خوب بود، فقط نکته ای که در فرم Register پیش فرض MVC وجود داره، اینکه این View به کنترلر Account وصله که هنگامی که CaptchaImageResult را درون آن قرار می‌دهیم، در فرم Register اجرا نمی‌شود.
می‌خواستم راهنمایی کنید که در این فرم چگونه باید عمل کنم. مرسی!

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۹/۲۸ ۱۰:۹

ویژگی AllowAnonymous را بالای آن قرار دهید. همچنین Url.Action دو پارامتر نام کنترلر و نام اکشن متد را دریافت می‌کند (برای استفاده از آن در سایر Viewهایی که در کنترلر جاری نیستند) .

نویسنده: کوروش ت
تاریخ: ۱۳۹۲/۱۰/۱۵ ۹:۱۳

با تشکر از راهنمایی شما، این مورد به خوبی در حالت localhost کار می‌کند و تغییر تصویر نیز به خوبی عمل می‌کند. هنگامی که سایت را Publish میکنم، تصاویر نمایش داده نمی‌شود. ممنون میشم راهنمایی کنید.

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۱۰/۱۵ ۹:۵۷

برای دیباگ کردن این مساله در سمت سرور بهتر است از ELMAH استفاده کنید. لاگ آن‌را بررسی کنید شاید جایی در پشت صحنه خطایی صادر می‌شود.

نویسنده: کوروش ت
تاریخ: ۱۳۹۲/۱۰/۱۵ ۱۰:۲۷

با سلام، Elmah Log رو چک کردم، مشکل دسترسی داشت که در قسمت Web.config تگ Trust رو اضافه کردم و فعلاً ظاهر شد. اضافه کردن trust مشکل امنیتی ایجاد نمی‌کنه؟

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۱۰/۱۵ ۱۰:۳۶

کدوم قسمتش مشکل دسترسی داشت؟ چه خطایی گرفتید دقیقا؟ چه trust ایی رو اضافه کردید؟ (در هاستی که بتونید trust level رو تغییر بدید، یعنی عملا از لحاظ امنیتی درست تنظیم نشده و مداخل رو قفل نکرده)

نویسنده: کوروش ت
تاریخ: ۱۳۹۲/۱۰/۱۵ ۱۰:۴۸

پیغام:

```
The system cannot find the file specified " source=" mscorlib "
```

کد تراست

```
< trust level = "Medium" originUrl = ".*" />
```

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۱۰/۱۵ ۱۱:۳۷

عجیبه که کار کرده. چون با حالت medium مشکل داره .

نویسنده: کوروش ت
تاریخ: ۱۳۹۲/۱۰/۱۵ ۱۳:۰

هنوز کامل باهاش سر و کله نزدم، ولی یک ثبت نام کردم دیدم داره درست کار می‌کنه. پس خطری نداره این کد.

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۱۰/۱۵ ۱۴:۵۶

trust level یعنی کد شما مثلا نتونه دایرکتوری فونت ویندوز رو روی سرور لیست کنه یا نتونه به کلیدهای رمزنگاری سطح ماشین دسترسی پیدا کنه. از این دید باید به آن نگاه کرد نه از دید کد خطرناک. این پروژه که سورس باز هست و می‌تونید کدهای آن را بررسی کنید.

نویسنده: کوروش ت
تاریخ: ۱۳۹۲/۱۰/۱۵ ۱۵:۲۶

منظور من این نبود، منظورم خطری برای سایر Third Party APIها بود که دادن تراست لول خطری برای ای پی آی‌های دیگه نداشته باشه. در مورد این کد هیچ چیز مبهمی وجود نداره و هم توضیحات خوب داده شده بود و هم از متدهای رادست و قابل فهم استفاده شده بود. از راهنمایی‌های شما هم بسیار ممنونم.

نویسنده: ایلیا
تاریخ: ۱۳۹۲/۱۰/۱۵ ۱۶:۲۰

با سلام.
من از این کپچا در یک سرور استفاده کردم، بر روی پورت 80 هیچ خطایی نمیگیرم و تصویر امنیتی بدرستی نمایش داده می‌شود ولی روی پورت 8080 تصویر کپچا لود نمیشود و خطای زیر را در کنسول کروم لاگ میکند:

```
System.Security.Cryptography.CryptographicException: Object already exists
```

```
Resource interpreted as Image but transferred with MIME type text/html:
"http://site:8080/Error/Index?aspxerrorpath=/Shared/CaptchaImage"
```

با تشکر.

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۱۰/۱۵ ۱۷:۴۲

اگر به سرور دسترسی دارید این دستور را اجرا کنید:

```
aspnet_regiis -pa "SampleKeys" "NT AUTHORITY\NETWORK SERVICE"
```

نویسنده: محسن
تاریخ: ۱۳۹۳/۰۱/۰۷ ۵:۳۵

من در یک پروژه می‌خواستم از این Captcha استفاده کنم. زمانی که صفحه اصلی لود میشه Captcha درست نمایش داده میشه ولی زمانی که کاربر از سایت SignOut می کنه دیگه محتوای Captcha نمایش داده نمی‌شود و اصلاً کنترولر CaptchaImage فراخوانی نمی‌شود ممنون میشم راهنمایی کنید

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۱/۰۷ ۱۱:۴۶

اگر فراخوانی نمی‌شود، یعنی کش شده. برای رفع این مشکل، ویژگی زیر را بر روی اکشن متد CaptchaImage قرار دهید:

```
[OutputCache(Location = OutputCacheLocation.None, NoStore = true)]
```

برای بررسی دقیق‌تر از افزونه‌ی فایرباگ استفاده کنید.

نویسنده: فواد عبداللهی
تاریخ: ۱۳۹۳/۰۵/۱۰ ۱۱:۴۸

سلام
میدونم پست نسبتا قدیمی!
ولی من یه مورد پیدا کردم و اونم اینه که در صورتی که captcha رو درست بزنی و برید به صفحه جدید (مثلا بعد لاگین برید به صفحه مدیر) دوباره back بزنی همون مقدار رو می‌بینید و متاسفانه دوباره کار هم میکنه.
کش رو هم غیر فعال کردم ولی تاثیری نداشت!

اگر راه حلی دارد ممنون میشم در میان بگذارید.

نویسنده: محسن خان
تاریخ: ۱۳۹۳/۰۵/۱۰ ۱۲:۷

نکته‌ی غیرفعال کردن کش مرورگر در MVC رو تست کردید؟

نویسنده: فواد عبداللهی
تاریخ: ۱۳۹۳/۰۵/۱۰ ۱۲:۱۲

بله جواب نداد!

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۵/۱۰ ۱۳:۲۴

1- فایل CaptchaImageResult.cs نگارش دوم را باز کنید.
- در انتهای آن تغییر زیر را اعمال کنید:

```
// در فایل CaptchaImageResult.cs
HttpResponseBase response = context.HttpContext.Response;
response.ContentType = "image/jpeg";
context.HttpContext.DisableBrowserCache(); // این سطر جدید است
bitmap.Save(response.OutputStream, ImageFormat.Jpeg);
```

متد الحاقی DisableBrowserCache در مطلب غیرفعال کردن کش مرورگر در MVC ارائه شده.

2- امضای متد Index کنترل Home نیاز به NoBrowserCache دارد (کل صفحه‌ی لاگین کش نشود):

```
public class HomeController : Controller
    {
        [NoBrowserCache]
        public ActionResult Index()
```

3- امضای متد CaptchaImage را در کنترلر Home به نحو زیر تغییر دهید (آدرس خاص تصویر نمایش داده شده، کش نشود):

```
[NoBrowserCache]
[OutputCache(Location = OutputCacheLocation.None, NoStore = true, Duration = 0, VaryByParam = "None")]
public CaptchaImageResult CaptchaImage(string rndDate)
```

سپس در محل استفاده در View به صورت زیر باید استفاده شود:

```
<img src="@Url.Action("CaptchaImage", "Home", routeValues: new{ rdnDate = DateTime.Now.Ticks })"/>
```

نویسنده: افشار محبی
تاریخ: ۱۳۹۳/۰۶/۲۵ ۱۰:۸

من هم همین مشکل را داشتم. قبل از این که توضیحات اینجا را ببینم، msdn را دیده بودم:

http://social.msdn.microsoft.com/Forums/vstudio/en-US/7ea48fd0-8d6b-43ed-b272-1a0249ae490f/systemsecuritycryptographycryptographicexception-the-system-cannot-find-the-file-specified?forum=clr

روشی که آنجا گفته بود باید application pool را از integrated به classic می‌بردی و از network service برای identity

استفاده می‌کردی. هر چند که این روش هم جواب داد ولی روش توضیح داده اینجا، یعنی اصلاح کد خیلی بهتر است.
البته در مورد من نیاز به اضافه کردن trust level به web.config نبود.

نویسنده: افشار محبی
تاریخ: ۱۳۹۳/۰۶/۲۵ ۱۰:۱۰

کد خیلی مفید و خوبی است. کاش آن را در github یا codeplex هم قرار می‌دادید تا همه به روز رسانی‌ها و pull requestهای احتمالی از همانجا انجام می‌شد. در هر صورت بابت این کد مفید تشکر می‌کنم.

نویسنده: محسن درپرستی
تاریخ: ۱۳۹۳/۰۹/۰۱ ۱۴:۴۵

یک مورد دیگه که باعث بروز این مشکل میشه افزونه فایرباگ هست. علتش رو نمی‌دونم اما در زمان فعال بودن فایرباگ دو بار درخواست برای تصویر امنیتی ارسال و مقدار درون کوکی با تصویری که می‌بینید متفاوت میشه.
درحالی که در زمان خاموش بودن فایرباگ این مشکل وجود نداره.

تا حالا با این پنجره حتما مواجه شدین:

دارید اطلاعات یک فرم داخل صفحه رو به سمت سرور میفرستید و پس از اتمام عملیات، صفحه دوباره نمایش داده میشه. در این حالت اگه دکمه F5 یا دکمه Refresh مرور گر رو بزنید، با این پنجره مواجه میشید که میگه دارید اطلاعات قبلی رو دوباره به سمت سرور میفرستید. بعضی وقت‌ها کاربران به هر دلیل دوباره صفحه رو Refresh میکنند و با این پنجره روبرو میشن بدون اینکه بدونن جریان از چه قراره، دوباره اطلاعات رو به سمت سرور میفرستن و این کار باعث ثبت اطلاعات تکراری میشه. برای جلوگیری از این کار الگویی به نام Post/Redirect/Get هست که راه حلی رو برای اینکار پیشنهاد میده.

راه حل به این صورت هست که پس از پست شدن فرم به سمت سرور و انجام عملیات، بجای اینکه صفحه، دوباره با استفاده از متد GET به کاربر نشون داده بشه، کاربر Redirect بشه به صفحه. برای توضیح این مسئله به سراغ AccountController که بصورت پش فرض وقتی یک پروژه ASP.NET MVC رو از نوع Internet ایجاد میکنید، وجود داره.

اکشن Register از نوع GET صفحه ثبت نام کاربر رو نمایش میده.

```
[HttpGet]
[AllowAnonymous]
public ActionResult Register()
{
      return View();
}
```

پس از اینکه کاربر اطلاعات داخل فرم رو پر کرد و به سمت سرور فرستاد و صحت اطلاعات فرستاده معتبر و عمل ثبت موفقیت آمیز بود برای ادامه کار به دو روش میتوان عمل کرد:

۱- کاربر به صفحه دیگری منتقل بشه و در اون صفحه پیام موفقیت آمیز بودن عملیات نشون داده بشه. مثلا معمولا پس از انجام عمل ثبت نام، کاربر به صفحه شخصی یا صفحه اصلی سایت منتقل میشه و یا در موقع ویرایش اطلاعات پش از انجام عمل ویرایش کاربر به صفحه دیگری که لیستی از آیتمها که کاربر یکی از آنها را ویرایش کرده باز گردانده میشه.

```
[HttpPost]
[AllowAnonymous]
[ValidateAntiForgeryToken]
public ActionResult Register(RegisterModel model)
{
    if (ModelState.IsValid)
```

```
    {
        // Attempt to register the user
        try
        {
            WebSecurity.CreateUserAndAccount(model.UserName, model.Password);
            WebSecurity.Login(model.UserName, model.Password);
            ViewBag.Message = "Successfully Registered!";

            // PRG has been maintained
            return RedirectToAction("Index", "Home");
        }
        catch (MembershipCreateUserException e)
        {
            ModelState.AddModelError("", ErrorCodeToString(e.StatusCode));
        }
    }
    // If we got this far, something failed, redisplay form
    return View(model);
}
```

۲- نمایش دوباره صفحه ولی با تغییر هدر صفحه به کد 303 . کد 303 به مروگر اعلام میکنه صفحه ریدایرکت شده است

```
[HttpPost]
[AllowAnonymous]
[ValidateAntiForgeryToken]
public ActionResult Register(RegisterModel model)
{
    if (ModelState.IsValid)
    {
        // Attempt to register the user
        try
        {
            WebSecurity.CreateUserAndAccount(model.UserName, model.Password);
            WebSecurity.Login(model.UserName, model.Password);
            ViewBag.Message = "Successfully Registered!";

            // PRG has been maintained
            return RedirectToAction("Register");
        }
        catch (MembershipCreateUserException e)
        {
            ModelState.AddModelError("", ErrorCodeToString(e.StatusCode));
        }
    }
    // If we got this far, something failed, redisplay form
    return View(model);
}
```

در این حالت دوباره صفحه ثبت نام نمایش داده میشه ولی با زدن دکمه رفرش، اطلاعات دوباره به سمت سرور فرستاده نمیشه

نظرات خوانندگان

نویسنده: محمد رضا ابراهیم راد
تاریخ: ۱۳۹۲/۰۳/۰۷ ۱۲:۲

با تشکر از مطلبی که نوشتید ولی من تو این کدها که 303 نمی‌بینم.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۳/۰۷ ۱۳:۳۲

البته روش توضیح داده شده همان روش متداول PRG است. اگر حتما نیاز به 303 دارید به روش زیر باید عمل کرد:

```
using System.Web.Mvc;

namespace TestMvcPRG.Helper
{
    public class Redirect303 : ActionResult
    {
        private string _url;

        public Redirect303(string url)
        {
            _url = url;
        }

        public override void ExecuteResult(ControllerContext context)
        {
            context.HttpContext.Response.StatusCode = 303; // redirect using GET
            context.HttpContext.Response.RedirectLocation = _url;
        }
    }

    public abstract class BaseController : Controller
    {
        public Redirect303 Redirect303(string actionName)
        {
            return new Redirect303(Url.Action(actionName));
        }

        public Redirect303 Redirect303(string actionName, object routeValues)
        {
            return new Redirect303(Url.Action(actionName, routeValues));
        }

        public Redirect303 Redirect303(string actionName, string controllerName)
        {
            return new Redirect303(Url.Action(actionName, controllerName));
        }

        public Redirect303 Redirect303(string actionName, string controllerName, object routeValues)
        {
            return new Redirect303(Url.Action(actionName, controllerName, routeValues));
        }
    }
}
```

و بعد برای استفاده:

```
using System.Web.Mvc;
using TestMvcRPG.Helper;

namespace TestMvcPRG.Controllers
{
    public class HomeController : BaseController
    {
        [HttpGet]
        public ActionResult Index()
        {
            return View();
        }
```

```
        [HttpPost]
        public ActionResult Index(string data)
        {
            if (ModelState.IsValid)
            {
                return Redirect303("Index"); // post-redirect-get
            }
            return View();
        }
    }
}
```

عنوان: **اعتبارسنجی سایتهای چند زبانه در ASP.NET MVC - قسمت اول**
نویسنده: محسن اسماعیل پور
تاریخ: ۱۳۹۲/۰۳/۰۸ ۰:۵
آدرس: www.dotnettips.info
برچسب‌ها: ASP.Net MVC, jQuery, Validation, Localization, Globalization, MVC

اگر در حال تهیه یک سایت چند زبانه هستید و همچنین سری مقالات Globalization در ASP.NET MVC رو دنبال کرده باشید میدانید که با تغییر Culture فایلهای Resource مورد نظر بارگذاری و نوشته‌های سایت تغییر میابند ولی با تغییر Culture رفتار اعتبارسنجی در سمت سرور نیز تغییر و اعتبارسنجی بر اساس Culture فعلی سایت انجام میگیرد. بررسی این موضوع را با یک مثال شروع میکنیم.

یک پروژه وب بسازید سپس به پوشه Models یک کلاس با نام ValueModel اضافه کنید. تعریف کلاس به شکل زیر هست:

```
public class ValueModel
{
    [Required]
    [Display(Name = "Decimal Value")]
    public decimal DecimalValue { get; set; }

    [Required]
    [Display(Name = "Double Value")]
    public double DoubleValue { get; set; }

    [Required]
    [Display(Name = "Integer Value")]
    public int IntegerValue { get; set; }

    [Required]
    [Display(Name = "Date Value")]
    public DateTime DateValue { get; set; }
}
```

به سراغ کلاس HomeController بروید و کدهای زیر را اضافه کنید:

```
[HttpPost]
public ActionResult Index(ValueModel valueModel)
{
    if (ModelState.IsValid)
    {
        return Redirect("Index");
    }

    return View(valueModel);
}
```

Culture را به fa-IR تغییر میدهیم، برای اینکار در فایل web.config در بخش system.web کد زیر اضافه نمایید:

```
<globalization culture="fa-IR" uiCulture="fa-IR" />
```

و در نهایت به سراغ فایل Index.cshtml بروید کدهای زیر رو اضافه کنید:

```
@using (Html.BeginForm())
{
    <ol>
        <li>
            @Html.LabelFor(m => m.DecimalValue)
            @Html.TextBoxFor(m => m.DecimalValue)
            @Html.ValidationMessageFor(m => m.DecimalValue)
        </li>
```

```
        <li>
            @Html.LabelFor(m => m.DoubleValue)
            @Html.TextBoxFor(m => m.DoubleValue)
            @Html.ValidationMessageFor(m => m.DoubleValue)
        </li>
        <li>
            @Html.LabelFor(m => m.IntegerValue)
            @Html.TextBoxFor(m => m.IntegerValue)
            @Html.ValidationMessageFor(m => m.IntegerValue)
        </li>
        <li>
            @Html.LabelFor(m => m.DateValue)
            @Html.TextBoxFor(m => m.DateValue)
            @Html.ValidationMessageFor(m => m.DateValue)
        </li>
        <li>
            <input type="submit" value="Submit"/>
        </li>
    </ol>
}
```

پرژه را اجرا نمایید و در ۲ تکست باکس اول ۲ عدد اعشاری را و در ۲ تکست باکس آخر یک عدد صحیح و یک تاریخ وارد نمایید و سپس دکمه Submit را بزنید. پس از بازگشت صفحه از سمت سرور در در ۲ تکست باکس اول با این پیامها روبرو میشوید که مقادیر وارد شده نامعتبر میباشند.

اگر پروژه رو در حالت دیباگ اجرا کنیم و نگاهی به داخل ModelState بیاندازیم، میبینیم که کاراکتر جدا کننده قسمت اعشاری برای fa-IR '/' میباشد که در اینجا برای اعداد مورد نظر کاراکتر '.' وارد شده است.

برای فایق شدن بر این مشکل یا باید سمت سرور اقدام کرد یا در سمت کلاینت. در بخش اول راه حل سمت کلاینت را بررسی مینماییم.

در سمت کلاینت برای اینکه کاربر را مجبور به وارد کردن کاراکترهای مربوط به Culture فعلی سایت نماییم باید مقادیر وارد شده را اعتبارسنجی و در صورت معتبر نبودن مقادیر پیام مناسب نشان داده شود. برای اینکار از کتابخانه jQuery Globalize استفاده میکنیم. برای اضافه کردن jQuery Globalize از طریق کنسول nuget فرمان زیر اجرا نمایید:

```
PM> Install-Package jquery-globalize
```

پس از نصب کتابخانه اگر به پوشه Scripts نگاهی بیاندازید میبینید که پوشای با نام jquery.globalize اضافه شده است. درداخل پوشه زیر پوشی دیگری با نام cultures وجود دارد که در آن Culture های مختلف وجود دارد و بسته به نیاز میتوان از آنها استفاده کرد. دوباره به سراغ فایل Index.cshtm بروید و فایلهای جاوا اسکریپتی زیر را به صفحه اضافه کنید:

```
<script src="~/Scripts/jquery.validate.js"> </script>
<script src="~/Scripts/jquery.validate.unobtrusive.js"> </script>
<script src="~/Scripts/jquery.globalize/globalize.js"> </script>
<script src="~/Scripts/jquery.globalize/cultures/globalize.culture.fa-IR.js"> </script>
```

در فایل globalize.culture.fa-IR.js کاراکتر جدا کننده اعشاری '.' در نظر گرفته شده است که مجبور به تغییر آن هستیم. برای اینکار فایل را باز کرده و numberFormat را پیدا کنید و آن را به شکل زیر تغییر دهید:

```
numberFormat: {
    pattern: ["n-"],
    ".": "/",
    currency: {
        pattern: ["$n-", "$ n"],
        ".": "/",
        symbol: "ریال"
    }
},
```

و در نهایت کدهای زیر را به فایل Index.cshtml اضافه کنید و برنامه را دوباره اجرا نمایید :

```
Globalize.culture('fa-IR');
$.validator.methods.number = function(value, element) {
    if (value.indexOf('.') > 0) {
        return false;
    }
    var splitedValue = value.split('/');
    if (splitedValue.length === 1) {
        return !isNaN(Globalize.parseInt(value));
    } else if (splitedValue.length === 2 && $.trim(splitedValue[1]).length === 0) {
        return false;
    }
    return !isNaN(Globalize.parseFloat(value));
};
};
```

در خط اول Culture را ست مینمایم و در ادامه نحوه اعتبارسنجی را در unobtrusive validation تغییر میدهیم . از آنجایی که برای اعتبارسنجی عدد وارد شده از تابع parseFloat استفاده میشود، کاراکتر جدا کننده قسمت اعشاری قابل قبول برای این تابع '.' است پس در داخل تابع دوباره '/' به '.' تبدیل میشود و سپس اعتبارسنجی انجام میشود از اینرو اگر کاربر '.' را نیز وارد نماید قابل قبول است به همین دلیل با این خط کد (if (value.indexOf('.') > 0 وجود نقطه را بررسی میکنیم تا در صورت وجود '.' پیغام خطا نشان داده شود.در خط بعدی بررسی مینماییم که اگر عدد وارد شده اعشاری نباشد از تابع parseInt استفاده نماییم. در خط بعدی این حالت را بررسی مینماییم که اگر کاربر عددی همچون ۱۲/ وارد کرد پیغام خطا صادر شود.

برای اعتبارسنجی تاریخ شمسی متاسفانه توابع کمکی برای تبدیل تاریخ در فایل globalize.culture.fa-IR.js وجود ندارد ولی اگر نگاهی به فایلهای Culture عربی بیاندازید همه دارای توابع کمکی برای تبدیل تاریخ هجری به میلادی هستند به همین دلیل امکان اعتبارسنجی تاریخ شمسی با استفاده از jQuery Globalize میسر نمیباشد. من خودم تعدادی توابع کمکی را به globalize.culture.fa-IR.js اضافه کردام که از تقویم فارسی آقای علی فرهادی برداشت شده است و با آنها کار اعتبارسنجی را انجام میدهیم. لازم به ذکر است این روش ۱۰۰% تست نشده است و شاید راه کاملا اصولی نباشد ولی به هر حال در اینجا توضیح میدهم. در فایل globalize.culture.fa-IR.js قسمت Gregorian_Localized را پیدا کنید و آن را با کدهای زیر جایگزین کنید:

```
Gregorian_Localized: {
    firstDay: 6,
    days: {
        names: ["شنبه", "جمعه", "پنجشنبه", "چهارشنبه", "سه شنبه", "دوشنبه", "یکشنبه"],
        namesAbbr: ["شنبه", "جمعه", "پنجشنبه", "چهارشنبه", "سه شنبه", "دوشنبه", "یکشنبه"],
        namesShort: ["ش", "ج", "پ", "چ", "س", "د", "ی"]
    },
    months: {
        names: ["ژانویه", "فوریه", "مارس", "آوریل", "می", "ژوئن", "ژوئیه", "اوت", "سپتامبر", "اکتبر", "نوامبر", "دسامبر", "", ""],
        namesAbbr: ["ژانویه", "فوریه", "مارس", "آوریل", "می", "ژوئن", "ژوئیه", "اوت", "سپتامبر", "اکتبر", "نوامبر", "دسامبر", "", ""]
    },
    AM: ["ق.ظ", "ق.ظ", "ق.ظ"],
    PM: ["ب.ظ", "ب.ظ", "ب.ظ"],
    patterns: {
        d: "yyyy/MM/dd",
        D: "yyyy/MM/dd",
        t: "hh:mm tt",
        T: "hh:mm:ss tt",
        f: "yyyy/MM/dd hh:mm tt",
        F: "yyyy/MM/dd hh:mm:ss tt",
        M: "dd MMMM"
    },
    JalaliDate: {
        g_days_in_month: [31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31],
        j_days_in_month: [31, 31, 31, 31, 31, 31, 30, 30, 30, 30, 30, 29]
    },
    gregorianToJalali: function (gY, gM, gD) {
        gY = parseInt(gY);
        gM = parseInt(gM);
        gD = parseInt(gD);
        var gy = gY - 1600;
        var gm = gM - 1;
        var gd = gD - 1;

        var gDayNo = 365 * gy + parseInt((gy + 3) / 4) - parseInt((gy + 99) / 100) + parseInt((gy + 399) / 400);

        for (var i = 0; i < gm; ++i)
            gDayNo += Globalize.culture().calendars.Gregorian_Localized.JalaliDate.g_days_in_month[i];
        if (gm > 1 && ((gy % 4 == 0 && gy % 100 != 0) || (gy % 400 == 0)))
            /* leap and after Feb */
            ++gDayNo;
        gDayNo += gd;

        var jDayNo = gDayNo - 79;

        var jNp = parseInt(jDayNo / 12053);
        jDayNo %= 12053;

        var jy = 979 + 33 * jNp + 4 * parseInt(jDayNo / 1461);

        jDayNo %= 1461;

        if (jDayNo >= 366) {
            jy += parseInt((jDayNo - 1) / 365);
            jDayNo = (jDayNo - 1) % 365;
        }

        for (var i = 0; i < 11 && jDayNo >= Globalize.culture().calendars.Gregorian_Localized.JalaliDate.j_days_in_month[i]; ++i) {
            jDayNo -= Globalize.culture().calendars.Gregorian_Localized.JalaliDate.j_days_in_month[i];
        }
        var jm = i + 1;
        var jd = jDayNo + 1;

        return [jy, jm, jd];
    },
    jalaliToGregorian: function (jY, jM, jD) {
```

```
        jY = parseInt(jY);
        jM = parseInt(jM);
        jD = parseInt(jD);
        var jy = jY - 979;
        var jm = jM - 1;
        var jd = jD - 1;

        var jDayNo = 365 * jy + parseInt(jy / 33) * 8 + parseInt((jy % 33 + 3) / 4);
        for (var i = 0; i < jm; ++i) jDayNo +=
Globalize.culture().calendars.Gregorian_Localized.JalaliDate.j_days_in_month[i];

        jDayNo += jd;

        var gDayNo = jDayNo + 79;

        var gy = 1600 + 400 * parseInt(gDayNo / 146097); /* 146097 = 365*400 + 400/4 - 400/100 +
400/400 */
        gDayNo = gDayNo % 146097;

        var leap = true;
        if (gDayNo >= 36525) /* 36525 = 365*100 + 100/4 */ {
            gDayNo--;
            gy += 100 * parseInt(gDayNo / 36524); /* 36524 = 365*100 + 100/4 - 100/100 */
            gDayNo = gDayNo % 36524;

            if (gDayNo >= 365)
                gDayNo++;
            else
                leap = false;
        }

        gy += 4 * parseInt(gDayNo / 1461); /* 1461 = 365*4 + 4/4 */
        gDayNo %= 1461;

        if (gDayNo >= 366) {
            leap = false;

            gDayNo--;
            gy += parseInt(gDayNo / 365);
            gDayNo = gDayNo % 365;
        }

        for (var i = 0; gDayNo >=
Globalize.culture().calendars.Gregorian_Localized.JalaliDate.g_days_in_month[i] + (i == 1 && leap) ;
i++)
            gDayNo -= Globalize.culture().calendars.Gregorian_Localized.JalaliDate.g_days_in_month[i] +
(i == 1 && leap);
        var gm = i + 1;
        var gd = gDayNo + 1;

        return [gy, gm, gd];
    },
    checkDate: function (jY, jM, jD) {
        return !(jY < 0 || jY > 32767 || jM < 1 || jM > 12 || jD < 1 || jD >
            (Globalize.culture().calendars.Gregorian_Localized.JalaliDate.j_days_in_month[jM - 1] + (jM
== 12 && !((jY - 979) % 33 % 4))));
    },
    convert: function (value, format) {
        var day, month, year;

        var formatParts = format.split('/');
        var dateParts = value.split('/');
        if (formatParts.length !== 3 || dateParts.length !== 3) {
            return false;
        }

        for (var j = 0; j < formatParts.length; j++) {
            var currentFormat = formatParts[j];
            var currentDate = dateParts[j];
            switch (currentFormat) {
                case 'dd':
                    if (currentDate.length === 2 || currentDate.length === 1) {
                        day = currentDate;
                    } else {
                        year = currentDate;
                    }
                    break;
                case 'MM':
                    month = currentDate;
                    break;
                case 'yyyy':
```

```
                    if (currentDate.length === 4) {
                        year = currentDate;
                    } else {
                        day = currentDate;
                    }
                    break;
                default:
                    return false;
            }
        }

        year = parseInt(year);
        month = parseInt(month);
        day = parseInt(day);
        var isValidDate = Globalize.culture().calendars.Gregorian_Localized.checkDate(year, month,
day);
        if (!isValidDate) {
            return false;
        }

        var grDate = Globalize.culture().calendars.Gregorian_Localized.jalaliToGregorian(year, month,
day);
        var shDate = Globalize.culture().calendars.Gregorian_Localized.gregorianToJalali(grDate[0],
grDate[1], grDate[2]);

        if (year === shDate[0] && month === shDate[1] && day === shDate[2]) {
            return true;
        }

        return false;
    }
},
```

روال کار در تابع convert به اینصورت است که ابتدا تاریخ وارد شده را بررسی مینماید تا معتبر بودن آن معلوم شود به عنوان مثال اگر تاریخی مثل 31/12/1392 وارد شده باشد و در ادامه برای بررسی بیشتر تاریخ یک بار به میلادی و تاریخ میلادی دوباره به شمسی تبدیل میشود و با تاریخ وارد شده مقایسه میشود و در صورت برابری تاریخ معتبر اعلام میشود. در فایل Index.cshtml کدهای زیر اضافی نمایید:

```
$.validator.methods.date = function (value, element) {
    return Globalize.culture().calendars.Gregorian_Localized.convert(value, 'yyyy/MM/dd');
};
```

برای اعتبارسنجی تاریخ میتوانید از ۲ فرمت استفاده کنید:

۱ - yyyy/MM/dd

۲ - dd/MM/yyyy

البته از توابع اعتبارسنجی تاریخ میتوانید به صورت جدا استفاده نمایید و لزومی ندارد آنها را همراه با jQuery Globalize بکار ببرید. در آخر خروجی کار به این شکل است:

Decimal Value

1/3

Double Value

1.3 The field Double Value must be a number.

Integer Value

11/ The field Integer Value must be a number.

Date Value

1392/12/30 The field Date Value must be a date.

Submit

در کل استفاده از jQuery Globalize برای اعتبارسنجی در سایتهای چند زبانه به نسبت خوب میباشد و برای هر زبان میتوانید از culture مورد نظر استفاده نمایید. در قسمت دوم این مطلب به بررسی بخش سمت سرور میپردازیم.

عنوان: پلاگین DataTables کتابخانه jQuery - قسمت چهارم
نویسنده: پژمان پارسائی
تاریخ: ۱۳۹۲/۰۴/۲۵ ۷:۵
آدرس: www.dotnettips.info
گروه‌ها: ASP.Net MVC, jQuery, DataTables, jQuery-Plugins

همان طور که قبلا اشاره کردیم، این پلاگین می‌تواند از یک زبان برنامه نویسی سمت سرور داده‌های مورد نیاز خودش را دریافت کند. می‌توانید داده‌ها را با استفاده از AJAX و به صورت JSON از سرور دریافت کرده و با استفاده از DataTables آنها را در جدول تزریق کنید. در این قسمت سعی خواهیم کرد تا با استفاده از jQuery DataTables یک گرید را در MVC ایجاد کنیم. البته برای حذف جزئیات داده‌ها به جای این که از یک بانک اطلاعاتی دریافت شوند، در حافظه ساخته می‌شوند. در هر صورت اساس کار یکی است.

قصد داریم تا مانند مثال قسمت قبل، مجموعه ای از اطلاعات مربوط به مرورگرهای مختلف را در یک جدول نشان دهیم، اما این بار منبع داده ما فرق می‌کند. منبع داده از طرف سرور فراهم می‌شود. هر مرورگر - همان طور که در قسمت قبل مشاهده نمودید - شامل اطلاعات زیر خواهد بود:

موتور رندرگیری (Engine)

نام مرورگر (Name)

پلتفرم (Platform)

نسخه موتور (Version)

نمره سی اس اس (Grade)

به همین دلیل در سمت سرور، کلاسی خواهیم ساخت که نمایانگر یک مرورگر باشد. بدین صورت:

```
public class Browser
{
    public int Id { get; set; }
    public string Engine { get; set; }
    public string Name { get; set; }
    public string Platform { get; set; }
    public float Version { get; set; }
    public string Grade { get; set; }
}
```

استفاده از روش server side processing برای دریافت داده‌ها از سرور

این روش، یکی از امکانات jQuery DataTables است که با استفاده از آن، کلاینت تنها یک مصرف کننده صرف خواهد بود و وظیفه پردازش اطلاعات - یعنی تعداد رکوردهایی که برگشت داده می‌شود، صفحه بندی، مرتب سازی، جستجو، و غیره - به عهده سرور خواهد بود.

برای به کار گیری این روش، اولین کار این است که ویژگی bServerSide را true کنیم، مثلا بدین صورت:

```
var $table = $('#browsers-grid');
$table.dataTable({
      "bServerSide": true,
      "sAjaxSource": "/Home/GetBrowsers"
 });
```

همچنین ویژگی sAjaxSource را به Url ی که باید داده‌ها از آن دریافت شوند مقداردهی می‌کنیم.

به صورت پیش فرض مقدار ویژگی bServerSide مقدار false است؛ که یعنی منبع داده این پلاگین از سمت سرور خوانده نشود. اگر true باشد منبع داده و خیلی اطلاعات دیگر مربوط به داده‌های درون جدول باید از سرور به مرورگر کاربر پس فرستاده شوند. با true کردن مقدار bServerSide، آنگاه DataTables اطلاعاتی را راجع به شماره صفحه جاری، اندازه هر صفحه، شروط فیلتر کردن داده ها، مرتب سازی ستون ها، و غیره را به سرور می‌فرستد. همجنین انتظار می‌رود تا سرور در پاسخ به این درخواست، داده‌های مناسبی را به فرمت JSON به مرورگر پس بفرستد. در حالتی که bServerSide مقدار true به خود بگیرد، پلاگین فقط رابطه متقابل بین کاربر و سرور را مدیریت می‌کند و هیچ پردازشی را انجام نمی‌دهد.

در این درخواست XHR یا Ajax ی پارامترهایی که به سرور ارسال می‌شوند این‌ها هستند:

iDisplayStart عدد صحیح
نقطه شروع مجموعه داده جاری

iDisplayLength عدد صحیح
تعداد رکوردهایی که جدول می‌تواند نمایش دهد. تعداد رکوردهایی که از طرف سرور برگشت داده می‌شود باید با این عدد یکسان باشند.

iColumns عدد صحیح
تعداد ستونهایی که باید نمایش داده شوند.

sSearch رشته
فیلد جستجوی عمومی

bRegex بولین
اگر true باشد معنی آن این است که می‌توان از عبارات باقاعده برای جستجوی عبارتی خاص در کل ستون‌های جدول استفاده کرد. مثلا در کادر جستجو نوشت :

```
^[1-5]$
```

که یعنی 1 و 5 همه عددهای بین 1و 5.

bSearchable_(int) بولین
نمایش می‌دهد که یک ستون در طرف کاربر قابلیت searchable آن true هست یا نه.

sSearch_(int) رشته
فیلتر مخصوص هر ستون. اگر از ویژگی multi column filtering پلاگین استفاده شود به صورت ، sSearch1 ، sSearch0 sSeach2 و ... به طرف سرور ارسال می‌شوند. شماره انتهای هر کدام از پارامترها بیانگر شماره ستون جدول است.

bRegex_(int) بولین
اگر true باشد، بیان می‌کند که می‌توان از عبارت با قاعده در ستون شماره int جهت جستجو استفاده کرد.

bSortable_(int) بولین

مشخص می‌کند که آیا یک ستون در سمت کلاینت، قابلیت مرتب شدن بر اساس آن وجود دارد یا نه. (در اینجا int اندیس ستون را مشخص می‌کند)

iSortingCols عدد صحیح
تعداد ستون هایی که باید مرتب سازی بر اساس آنها صورت پذیرد. در صورتی که از امکان multi column sorting استفاده کنید این مقدار می‌تواند بیش از یکی باشد.

iSortCol_(int) عدد صحیح
شماره ستونی که باید بر اساس آن عملیات مرتب سازی صورت پذیرد.

sSortDir_(int) رشته
نحوه مرتب سازی ؛ شامل صعودی (asc) یا نزولی (desc)

mDataProp_(int) رشته
اسم ستون‌های درون جدول را مشخص می‌کند.

sEcho رشته
اطلاعاتی که datatables از آن برای رندر کردن جدول استفاده می‌کند.

شکل زیر نشان می‌دهد که چه پارامترهایی به سرور ارسال می‌شوند.

Parameters application/x-www-form-urlencoded

bRegex	false
bRegex_0	false
bRegex_1	false
bRegex_2	false
bRegex_3	false
bRegex_4	false
bSearchable_0	true
bSearchable_1	true
bSearchable_2	true
bSearchable_3	true
bSearchable_4	true
bSortable_0	true
bSortable_1	true
bSortable_2	true
bSortable_3	true
bSortable_4	true
iColumns	5
iDisplayLength	25
iDisplayStart	0
iSortCol_0	1
iSortingCols	1
mDataProp_0	Engine
mDataProp_1	Name
mDataProp_2	Platform
mDataProp_3	Version
mDataProp_4	Grade
sColumns	
sEcho	2
sSearch	
sSearch_0	
sSearch_1	
sSearch_2	
sSearch_3	
sSearch_4	
sSortDir_0	desc

شکل ب) پارامترهای ارسالی به سرور به صورت json

بعضی از این پارامترها بسته به تعداد ستون‌ها قابل تغییر هستند. (آن پارامترهایی که آخرشان یک عدد هست که نشان دهنده شماره ستون مورد نظر می‌باشد)

در پاسخ به هر درخواست XHR که datatables به سرور می‌فرستد، انتظار دارد تا سرور نیز یک شیء json را با فرمت مخصوص که شامل پارامترهای زیر می‌شود به او پس بفرستد:

iTotalRecords عدد صحیح
تعداد کل رکوردها (قبل از عملیات جستجو) یا به عبارت دیگر تعداد کل رکوردهای درون آن جدول از دیتابیس که داده‌ها باید از آن دریافت شوند. تعداد کل رکوردهایی که در طرف سرور وجود دارند. این مقدار فقط برای نمایش به کاربر برگشت داده می‌شود و نیز از آن برای صفحه بندی هم استفاده می‌شود.

iTotalDisplayRecords عدد صحیح
تعداد کل رکوردها (بعد از عملیات جستجو) یا به عبارت دیگر تعداد کل رکوردهایی که بعد از عملیات جستجو پیدا می‌شوند نه فقط آن تعداد رکوردی که به کاربر پس فرستاده می‌شوند. تعداد کل رکوردهایی که با شرط جستجو مطابقت دارند. اگر کاربر چیزی را جستجو نکرده باشد مقدار این پارامتر با پارامتر iTotalRecords یکسان خواهد بود.

sEcho عدد صحیح
یک عدد صحیح است که در قالب رشته در تعامل بین سرور و کلاینت جا به جا می‌شود. این مقدار به ازاء هر درخواست تغییر می‌کند. همان مقداری که مرورگر به سرور می‌دهد را سرور هم باید به مرورگر تحویل بدهد. برای جلوگیری از حملات XSS باید آن را تبدیل به عدد صحیح کرد. پلاگین DataTables مقدار این پارامتر را برای هماهنگ کردن و منطبق کردن درخواست ارسال شده و جواب این درخواست استفاده می‌کند. همان مقداری که مروگر به سرور می‌دهد را باید سرور تحویل به مرورگر بدهد.

sColumns رشته
اسم ستون‌ها که با استفاده از کاما از هم جدا شده اند. استفاده از آن اختیاری است و البته منسوخ هم شده است و در نسخه‌های جدید jQuery DataTables از آن پشتیبانی نمی‌شود.

aaData آرایه
همان طور که قبلا هم گفتیم، مقادیر سلول هایی را که باید در جدول نشان داده شوند را در خود نگهداری می‌کند. یعنی در واقع داده‌های جدول در آن ریخته می‌شوند. هر وقت که DataTables داده‌های مورد نیازش را دریافت می‌کند، سلول‌های جدول html مربوطه اش را از روی آرایه aaData ایجاد می‌کند. تعداد ستون‌ها در این آرایه دو بعدی، باید با تعداد ستون‌های جدول html مربوطه به آن یکسان باشد

شکل زیر پارامترها دریافتی از سرور را نشان می‌دهند:

شکل ب) پارامترهای دریافتی از سرور به صورت json

استفاده از روش server side processing در mvc

همان طور که گفتیم، کلاینت به سرور یک سری پارامترها را ارسال می‌کند و آن پارامترها را هم شرح دادیم. برای دریافت این پارامترها طرف سرور، احتیاج به یک مدل هست. این مدل به صورت زیر پیاده سازی خواهد شد:

```
/// <summary>
/// Class that encapsulates most common parameters sent by DataTables plugin
/// </summary>
public class jQueryDataTableParamModel
{
    /// <summary>
    /// Request sequence number sent by DataTable,
    /// same value must be returned in response
    /// </summary>
    public string sEcho { get; set; }
    /// <summary>
    /// Text used for filtering
    /// </summary>
    public string sSearch { get; set; }
    /// <summary>
    /// Number of records that should be shown in table
    /// </summary>
    public int iDisplayLength { get; set; }
    /// <summary>
    /// First record that should be shown(used for paging)
    /// </summary>
    public int iDisplayStart { get; set; }
    /// <summary>
    /// Number of columns in table
    /// </summary>
    public int iColumns { get; set; }
    /// <summary>
    /// Number of columns that are used in sorting
    /// /// </summary>
    public int iSortingCols { get; set; }
    /// <summary>
    /// Comma separated list of column names
    /// </summary>
    public string sColumns { get; set; }
}
```

مدل بایندر mvc وظیفه مقداردهی به خصوصیات درون این کلاس را بر عهده دارد، بقیه پارامترهایی که به سرور ارسال می‌شوند و در این کلاس نیامده اند، از طریق شیء Request در دسترس خواهند بود.

اکشن متدی که مدل بالا را دریافت می‌کند، می‌تواند به صورت زیر پیاده سازی شود. این اکشن متد وظیفه پاسخ دادن به درخواست DataTables بر اساس پارامترهای ارسال شده در مدل DataTablesParam را دارد. خروجی این اکشن متد شامل پارارمترهای مورد نیاز پلاگین DataTables برای تشکیل جدول است که آنها را هم شرح دادیم.

```
public JsonResult GetBrowsers(jQueryDataTableParamModel param)
{
        IQueryable<Browser> allBrowsers = new Browsers().CreateInMemoryDataSource().AsQueryable();

        IEnumerable<Browser> filteredBrowsers;

        // Apply Filtering
        if (!string.IsNullOrEmpty(param.sSearch))
        {
                filteredBrowsers = new Browsers().CreateInMemoryDataSource()
                    .Where(x => x.Engine.Contains(param.sSearch)
                                || x.Grade.Contains(param.sSearch)
                                || x.Name.Contains(param.sSearch)
                                || x.Platform.Contains(param.sSearch)
                    ).ToList();
                float f;
                if (float.TryParse(param.sSearch, out f))
                {
                    filteredBrowsers = filteredBrowsers.Where(x => x.Version.Equals(f));
                }
        }
        else
        {
                filteredBrowsers = allBrowsers;
        }

        // Apply Sorting
        var sortColumnIndex = Convert.ToInt32(Request["iSortCol_0"]);
        Func<Browser, string> orderingFunction = (x => sortColumnIndex == 0 ? x.Engine :
                                                        sortColumnIndex == 1 ? x.Name :
                                                        sortColumnIndex == 2 ? x.Platform :
                                                        sortColumnIndex == 3 ? x.Version.ToString()
:
                                                        sortColumnIndex == 4 ? x.Grade :
                                                            x.Name);

        var sortDirection = Request["sSortDir_0"]; // asc or desc
        filteredBrowsers = sortDirection == "asc" ? filteredBrowsers.OrderBy(orderingFunction) :
filteredBrowsers.OrderByDescending(orderingFunction);

        // Apply Paging
        var enumerable = filteredBrowsers.ToArray();
        IEnumerable<Browser> displayedBrowsers = enumerable.Skip(param.iDisplayStart).
                Take(param.iDisplayLength).ToList();

        return Json(new
        {
                sEcho = param.sEcho,
                iTotalRecords = allBrowsers.Count(),
                iTotalDisplayRecords = enumerable.Count(),
                aaData = displayedBrowsers
        }, JsonRequestBehavior.AllowGet);
}
```

تشریح اکشن متد GetBrowsers :

این اکشن متد از مدل jQueryDataTableParamModel به عنوان پارامتر ورودی خود استفاده می‌کند. این مدل همان طور هم که گفتیم، شامل یک سری خصوصیت است که توسط پلاگین jQuery DataTables مقداردهی می‌شوند و همچنین مدل بایندر mvc وظیفه بایند کردن این مقادیر به خصوصیات درون این کلاس را بر عهده خواهد داشت. درون بدنه اکشن متد GetBrowsers داده‌ها بعد از اعمال عملیات فیلترینگ، مرتب سازی، و صفحه بندی به فرمت مناسبی درآمده و به طرف مرورگر فرستاده خواهند شد.

برای پیاده سازی کدهای طرف کلاینت نیز، درون یک View کدهای زیر قرار خواهند گرفت:

```
$(function () {
        var $table = $('#browsers-grid');
        $table.dataTable({
                "bProcessing": true,
                "bStateSave": true,
                "bServerSide": true,
                "bFilter": true,
                "sDom": 'T<"clear">lftipr',
                "aLengthMenu": [[5, 10, 25, 50, -1], [5, 10, 25, 50, "All"]],
                "bAutoWidth": false,
                "sAjaxSource": "/Home/GetBrowsers",
                "fnServerData": function (sSource, aoData, fnCallback) {
                    $.ajax({
                        "dataType": 'json',
                        "type": "POST",
                        "url": sSource,
                        "data": aoData,
                        "success": fnCallback
                    });
                },
                "aoColumns": [
                    { "mDataProp": "Engine" },
                    { "mDataProp": "Name" },
                    { "mDataProp": "Platform" },
                    { "mDataProp": "Version" },
                    { "mDataProp": "Grade" }
                ],
                "oLanguage": {
                        "sUrl": "/Content/dataTables.persian.txt"
                }
        });
});
```

تشریح کدها:

fnServerData :

این متد، در واقع نحوه تعامل سرور و کلاینت را با استفاده از درخواستهای XHR مشخص خواهد کرد.

oLanguage :

برای فعال سازی زبان فارسی، فیلدهای مورد نیاز ترجمه شده و در یک فایل متنی قرار داده شده اند. کافی است آدرس این فایل متنی به ویژگی oLanguage اختصاص داده شوند.

مثال این قسمت را از لینک زیر دریافت کنید:
DataTablesTutorial04.zip

لازم به ذکر است پوشه bin، obj، و packages جهت کاهش حجم این مثال از solution حذف شده اند. برای اجرای این مثال از اینجا کمک بگیرید.

مطالعه بیشتر

برای مطالعه بیشتر در مورد این پلاگین و نیز پیاده سازی آن در MVC می‌توانید به لینک زیر نیز مراجعه بفرمائید که بعضی از قسمتهای این مطلب هم از مقاله زیر استفاده کرده است:
jQuery DataTables and ASP.NET MVC Integration - Part I

نظرات خوانندگان

نویسنده: Sorosh
تاریخ: ۱۳۹۲/۱۱/۱۱ ۸:۴۹

با سلام و تشکر. می‌خوام یک ستون ردیف به این جدول اضافه کنم که به ازای صفحه، ردیف جلو بره یعنی از 1 نشه دوباره و یکی هم اضافه کردن یک Attr خاص و جدید به هر سطر در جدول مثل ProjectCode که اون هم داخلش Id اون سطر در دیتا بیس هستش . ممنونم اگه کمکم کنید

کد کارم رو هم می‌دهم . البته من در WebForm کار کردم.

```
var $table = $('#browsers-grid');
$table.dataTable({
"bJQueryUI": true,
"bProcessing": true,
"bSortClasses": false,
"bServerSide": true,
"bFilter": true,
"sPaginationType": "full_numbers",
"sScrollY": 400,
"sScrollX": "100%",
"sScrollXInner": "110%",
"bLengthChange": false,
"iDisplayLength": 20,

"aLengthMenu": [[5, 10, 25, 50, -1], [5, 10, 25, 50, "All"]],
"bAutoWidth": false,
"sAjaxSource": "Commands.aspx?cmd=all",
"fnServerData": function (sSource, aoData, fnCallback) {
  $.ajax({
"dataType": 'json',
"type": "POST",
"url": sSource,
"data": aoData,
"success": fnCallback
  });
},

"aoColumns":
  [

  { "mDataProp": "Code" },
  { "mDataProp": "Caption" },
  { "mDataProp": "Comment" }
  ],

"oLanguage":
  {
"sUrl": "dataTables.persian.txt"
  }
});
```

نویسنده: پژمان پارسائی
تاریخ: ۱۳۹۲/۱۱/۱۱ ۹:۴۷

سلام، خواهش می‌کنم. می‌تونید در سمت سرور بعد از واکشی اطلاعات از دیتابیس اونو داخل یک منبع داده درون حافظه ای بریزید و هر تعداد ستون که لازم دارید به اون منبع داده جدید اضافه کنید. و با مقدارهای مناسبی هر مدخل از اون منبع داده رو پر کنید. مثلا فرضا اگه جدول دیتابیس شما دارای سه ستون Code و Caption و Comment هست کلاس جدیدی بسازید که این سه تا ستون رو داره (به عنوان پروپرتی) و پروپرتی‌های دلخواه دیگه ای هم داره. مثلا پروپرتی RowNumber . بعد لیستی از داده‌ها رو که از دیتابیس واکشی کردید داخل لیستی از ViewModel ساخته شده بریزید و خصوصیت RowNumber هر ViewModel رو مقداردهی مناسبی کنید.

نویسنده: Sorosh
تاریخ: ۱۳۹۲/۱۱/۱۱ ۱۱:۶

متشکرم از پاسخ شما . البته اگه می‌شد نمونه‌ای ایجاد کنید خیلی بهتر بود که باز هم ممنونم. اما فقط برای بحث خصوصیت هر

سطر چی؟ می‌دونید اگه بخواهیم به هر سطر یک Attr جدید بدهیم چطوری باید اونو به TR نسبت بدهیم. مثلا من می‌خواهم به ازای هر سطر <tr ProjectCode='12'><tr/> داشته باشم.

نویسنده: Sorosh
تاریخ: ۱۳۹۲/۱۱/۱۱ ۱۳:۱

با سلام مجدد؛ بالاخره پیداش کردم. برای ایجاد خصوصیت جدید به ازاء هر سطر و گرفتن مقدار از aoData بصورت زیر است .
مثلا برای اضافه کردن Attr - StudentCode

```
fnRowCallback": function (nRow, aData, iDisplayIndex)
{

  var StudentCode= aData["Code"];
  $(nRow).attr("StudentCode",StudentCode);
  return nRow;
}
```

با تشکر از شما

نویسنده: sorosh
تاریخ: ۱۳۹۲/۱۱/۱۲ ۱۳:۳۱

با سلام؛ من می‌خوام با قراردادن یک دکمه روی فرم به نام به‌روز رسانی، خاصیت اطلاعات داخل کوکی که با متد bStateSave کار می‌کنه از بین بره و جدول دوباره از نو رفرش بشه و حالت‌های جستجو و مرتب سازی قبلی از بین بره. متشکرم

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۱/۱۲ ۱۴:۲

- راه حل رسمی برای حذف کوکی ندارد . فقط مدت زمان آن قابل تنظیم است.
- به مطلب « کوکی در جاوا اسکریپت » در مورد حذف کوکی در همان سمت کلاینت مراجعه کنید. نام کوکی‌ها را هم با استفاده از افزونه Cookies manager می‌توانید مشاهده کنید.

نویسنده: رویا حیدری
تاریخ: ۱۳۹۳/۱۰/۱۶ ۱۶:۳۱

با سلام و تشکر فراوان. من میخواستم یک ستون با عنوان عملیات به جدول اضافه کنم که به صورت منو باشه و عملیات مختلف مثل ویرایش و حذف و تغییرات دیگه را از اون منو انتخاب و انجام بدم. برای اینکار باید چه کاری انجام بدم؟ و اینکه آیا برای اضافه کردن امکان ویرایش باید حتما از Editable DataTable استفاده کنم یا میشه تو همین نوع ساده هم امکان ویرایش را اضافه کرد؟ (Inline Edit نمی‌خوام انجام بدم)
ممنون میشم اگه راهنمایی کنید.

نویسنده: رویا حیدری
تاریخ: ۱۳۹۳/۱۰/۱۷ ۱۵:۰

در این صفحه تا حدودی به جوابم رسیدم.

عنوان: **ایجاد لینک با یک تصویر بوسیله Html Helper**
نویسنده: مهراد
تاریخ: ۱۳۹۲/۰۵/۰۸ ۸:۵
آدرس: www.dotnettips.info
گروه‌ها: Extension Method, ASP.Net MVC, MVC

با Html Helper ما میتوانیم لینک‌های متن دار را ایجاد نماییم. شاید گاهی پیش آید که بجای لینک‌های متنی، از تصویر بجای لینک بخواهید استفاده نمایید. می‌توان هر زمانی، لینکی را که حاوی یک تصویر باشد، ایجاد کنید. اما با Asp.net MVC یکی از راه‌های مناسب برای انجام این کار استفاده از Extension methods است که از تکرار کد نویسی نیز جلو گیری می‌نماید.
همان طور که در کد زیر مشاهده می‌نمایید، این کد تشکیل شده است از نوشته لینک و نام اکشن متد و نام کنترلر آن، همراه با آی دی و تعریف css. کاری که قرار است صورت گیرد همانند ActionLink می‌باشد، با این تفاوت که تصویری هم به لینک ما اضافه خواهد شد:

```
@Html.ActionLink("Text to display", "ActionNameHere", "ControllerNameHere", new { id = 123 },
new{@class="myClass"})
```

برای نوشتن این متد جهت تولید لینک‌هایی به همراه تصویر، از کدهای زیر استفاده می‌کنیم:

```
public MvcHtmlString ActionImage(this HtmlHelper htmlHelper,
   string controller,
   string action,
   object routeValues,
   string imagePath,
   string alternateText = "",
   object htmlAttributes = null)
{
  var anchorBuilder = new TagBuilder("a");
  var url = new UrlHelper(HtmlHelper.ViewContext.RequestContext);
  anchorBuilder.MergeAttribute("href",url.Action(action,controller,routeValues));
  var imgBuilder = new TagBuilder("img");
  imgBuilder.MergeAttribute("src", url.Content(imagePath));
  imgBuilder.MergeAttribute("alt", alternateText);
  var attributes = (IDictionary<string, object>)
HtmlHelper.AnonymousObjectToHtmlAttributes(htmlAttributes);
  imgBuilder.MergeAttributes(attributes);
  string imgHtml = imgBuilder.ToString(TagRenderMode.SelfClosing);
  anchorBuilder.InnerHtml = imgHtml;
  return MvcHtmlString.Create(anchorBuilder.ToString());
}
```

اولین پارامتر اجازه گسترش HtmlHelper را می‌دهد و توسط آن می‌توان متدهای جدید خود را با کلمه کلیدی @Html شروع کنید. 4 پارامتر بعدی، پارامترهای ضروری هستند. یکی controller، پس از آن Action routeValues و سپس آدرس تصویر است. در نهایت، ما می‌توانیم در صورت نیاز متن جایگزین و ویژگی‌های اضافی متنی را نیز مشخص نماییم.
در رابطه با توضیحات مربوط به بدنه متد: ابتدا لینک‌ها را ایجاد خواهد کرد با استفاده از RequestContext و UrlHelper. سپس بخش ساخت تصویر و اضافه کردن آن به لینک است. برای استفاده از میتوان از کد زیر استفاده نمود:

```
@Html.ActionImage("Workout", "Delete", new { id = @workout.Id }, "/images/bluetrash.png", "Delete", new
{ id="deletebutton"})
```

در شکل بالا خروجی HTML تولید شده را مشاهده می‌نمایید.

نظرات خوانندگان

نویسنده: Hamid NCH
تاریخ: ۱۳۹۲/۰۵/۲۵ ۱۰:۱

سلام و ممنون از مطلب ارسالی خیلی مفیدتون. از این خط خطا گرفته میشه:

```
var url = new UrlHelper(HtmlHelper.ViewContext.RequestContext);
```

و خطای گرفته شده:

Error 1 An object reference is required for the non-static field, method, or property ''System.Web.Mvc.HtmlHelper.ViewContext.get

نویسنده: Hamid NCH
تاریخ: ۱۳۹۲/۰۵/۲۵ ۱۰:۶

با عرض معذرت.مشکل از کد خودم بود و در این کد:

```
var url = new UrlHelper(htmlHelper.ViewContext.RequestContext);
```

htmlHelper را به صورت HtmlHelper نوشته بودم و اصلا متوجه این بخش نبودم.ممنون

نویسنده: داود
تاریخ: ۱۳۹۲/۱۰/۰۷ ۱۴:۱۸

سلام
مرسی از مطالب مفیدتون. میشه راهنمایی کنید که چطوری توی یه پروژه MVC از این کد باید استفاده کرد؟

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۱۰/۰۷ ۱۹:۳۶

یک کلاس استاتیک جدید درست کنید. بعد متد الحاقی ActionImage رو بهش اضافه کنید. حالا در کدهای Razor مطابق توضیحی که دادند Html.ActionImage قابل استفاده می‌شود. اگر لازم بود فضام نام کلاس را هم ابتدای فایل View اضافه کنید.

نویسنده: داود
تاریخ: ۱۳۹۲/۱۰/۰۸ ۹:۲۹

```
Compiler Error Message: CS0103: The name 'workout' does not exist in the current context
@using ActionImage.Extentions;
@Html.ActionImage("Workout", "Delete", new { id = @workout.Id }, "/images/bluetrash.png", "Delete", new
{ id="deletebutton"})
```

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۱۰/۰۸ ۹:۵۱

بجای اون id هرچی دوست داشتی بنویس. احتمالا یک حلقه بوده که workout یکی از اشیاء اون هست. مهم نیست. جزو کار نیست.

نویسنده: رشوند
تاریخ: ۱۳۹۳/۰۴/۲۹ ۱۸:۴۷

سپاس, بسیار مفید بود.
البته در کد آقا مهرداد هم به صورت H نوشته شده که لطفا اصلاح بفرمایید...

نویسنده: رشوند
تاریخ: ۱۳۹۳/۰۵/۱۱ ۱۶:۳۰

یه سوال برام پیش اومد
فرض کنیم کاربری لاگین کرده رولش هم مثلا خبرنگاره,
در پروفایلش لیست خبر هایی که ثبت کرده را می‌بینه با توجه به مثالی که شما هم زدید , اگر مثلا عدد "1" لینک ویرایش (یا حذف) که نشانگر id است رو به صورت دستی(تغییر کد html و ...) تغییر بده, می‌تونه به خبر دیگه ای دسترسی داشته باشه ؟
خبری که ثبت کنندش خودش نیست بتبع هم اجازه ویرایش و حذف ان را نباید داشته باشه...
راه حل چیست؟ آیا باید قبل از نشان دادن صفحه ویرایش یا قبل از عمل حذف ثبت کننده خبر هم چک بشه (که فردی که این خبر رو ثبت نکرده دسترسی به ان نداشته باشه)؟
لطفا راهنمایی بفرمایید.
سپاس.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۵/۱۱ ۱۸:۳۵

- وجود anti forgery token جهت مقابله با جعل صفحه و درخواست است.
- امکان رمزنگاری خودکار فیلدهای مخفی وجود دارد.
- بررسی‌های سمت سرور هم جای خودشان را دارند و لازم هستند.

عنوان: **بارگزاری PartialView با استفاده از jQuery در زمان اجرا**
نویسنده: پویا امینی
تاریخ: ۱۳۹۲/۰۵/۲۲ ۱۸:۲۰
آدرس: www.dotnettips.info
گروه‌ها: ASP.Net MVC, jQuery, jQuery Ajax, MVC

مزیت استفاده از PartialViewها، ماژولار کردن برنامه است. برای مثال اگر صفحه جاری شما قرار است از چهار قسمت اخبار، منوی پویا، سخن روز و آمار کاربران تشکیل شود، می‌توان هر کدام را توسط یک PartialView پیاده سازی کرد و سپس صفحه اصلی را از کنار هم قرار دادن این PartialViewها تهیه نمود.(منبع).
در این پست قصد دارم به نحوه بارگزاری یک PartialView با استفاده از ASP.NET MVC بپردازم.
ابتدا یک پروژه جدید ایجاد کنید. حال می‌خواهیم زمانیکه صفحه اصلی سایت بارگزاری می‌شود، لیست تمام محصولات را نمایش دهد. برای نمایش محصولات یک PartialView جدید را ایجاد می‌کنیم؛ همانند شکل ذیل:

حال برای ساده کردن مثال، متن ثابتی را درون این PartialView می‌نویسیم:

```
product  List Partial
```

در ادامه قصد داریم که درIndex، این View را بارگزاری کنیم. برای این کار از متد ajax مربوط به کتابخانه jQuery به صورت زیر استفاده می‌کنیم:

```
$(function () {
        $.ajax({
            //مشخص کردن  اکشنی که باید فراخوانی شود
            url: '/Home/Details',
            contentType: 'application/html; charset=utf-8',
            type: 'GET',
            //نوع نتیجه بازگشتی
            dataType: 'html'

        })
.success(function (result) {
    //زمانی که کدهای سمت سرور بدون خطا اجرا شده اند
    //این قسمت فراخوانی می‌شود و نتیجه اکشن درون متغیر
    //result
    //قرار می‌گیرد
    $('#sectionContents').html(result);
```

```
})
.error(function (xhr, status) {
    alert(xhr.responseText);
});
    });
```

و پس از آن محلی را که قرار است PartialView در آن بارگزاری شود، ایجاد می‌کنیم:

```
<div id="sectionContents"></div>
```

حال فقط باید اکشن مورد نظر را در HomeController پیاده سازی کنیم:

```
public ActionResult Details()
        {
            return PartialView("partial/_ProductList");
        }
```

با استفاده از این کد مشخص کردیم که این اکشن، یک PartialView را با نام _ProductList برگشت می‌دهد. البته در یک مثال واقعی باید لیست محصولات را هم به این PartialView پاس دهیم.
نتیجه اجرا به صورت زیر است:

نظرات خوانندگان

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۵/۲۲ ۱۹:۵۹

مطالب تکمیلی که از همین روش استفاده می‌کنند:

به روز رسانی غیرهمزمان قسمتی از صفحه به کمک jQuery در ASP.NET MVC

پیاده سازی دکمه «بیشتر» یا «اسکرول نامحدود» به کمک jQuery در ASP.NET MVC

آشنایی با تکنیک‌های Ajax در ASP.NET MVC

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۵/۲۳ ۰:۲۸

یک سؤال: «بارگزاری» درست هست یا « بارگذاری »؟ البته مطابق مطلبی که در ویکی‌پدیا نوشته باید از «بارگیری» استفاده کرد!

نویسنده: imo0
تاریخ: ۱۳۹۲/۰۶/۰۷ ۱۰:۶

یک سوال. من یک پارشال ویو دارم که خودش نیاز داره به یه سری کدهای جاوا اسکریپت و استایل .
من پارشال ویو هارو با اجکس لود میکنم . چطوری اون فایل‌های استایل و غیرشم لود کنم ؟. @RenderSection جواب نمیده.
چون اصلا صفحه ریفرش نمیشه.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۶/۰۷ ۱۱:۸

- راه اول: زمانیکه View اصلی در برگیرنده آن نمایش داده می‌شود، کلیه فایل‌های متناظر را هم الحاق کنید تا به صورت خودکار در جزئی از صفحه آن، که بعدا به روز خواهد شد، نیز اعمال شود.
- راه دوم: اصلا از RenderSection در یک partial view که قرار است Ajax ایی بارگذاری شود، استفاده نکنید. معمولی این‌ها را الصاق یا تعریف کنید. مثل تعاریف یک HTML ساده. یک نفر هم اینجا براش HtmlHelper نوشته ولی نکته اصلی یکی است؛ الصاق و تعریف معمولی فایل‌های مورد نیاز.
- همچنین خود jQuery امکان بارگذاری اسکریپت‌ها را به صورت پویا دارد . زمانیکه complete عملیات Ajax ایی رخداد، متد getScript عنوان شده را فراخوانی کنید. برای CSS هم به صورت زیر عمل کنید:

```
$("<style></style>").appendTo("head").html(data);
```

نویسنده: خوشقدم
تاریخ: ۱۳۹۲/۰۶/۱۳ ۸:۵۴

سلام

مشکلی که من دارم این است که در صفحه دوتا PartialView دارم که هر کدام با استفاده jquery همان طور که شما توضیح داده اید لود می‌شوند ، اما برای رفرش کردن اونها مشکل دارم و نمی‌تونم به هر کدام شماره Page و سایر اطلاعات WebGrid چطوری ارسال کنم؟

ممنون

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۶/۱۳ ۹:۰

نحوه صحیح ارسال پارامترها توسط jQuery Ajax
ارسال کوئری استرینگ‌ها به همراه عملیات Ajax در jQuery

ضمنا بحث وب گرید ، خارج از موضوع مطلب جاری است. از پارامتر ajaxUpdateContainerId آن افزونه خاص باید برای مدیریت مسایل paging و sorting استفاده کنید.

نویسنده: مجتبی شایان فر
تاریخ: ۱۳۹۲/۰۶/۲۸ ۱۳:۴

باسلام و خسته نباشید.
من مثال فوق را خط به خط اجرا کردم ولی partial view نمایش داده نمیشه. فکر کنم مکان قطعه کد Ajax را اشتباه جایگذاری کردم اگه ممکنه راهنمائی می‌کنید که قطعه:

```
$( function () {
$.ajax({
//مشخص کردن  اکشنی که باید فراخوانی شود
url: '/Home/Details' ,
contentType: 'application/html; charset=utf-8' ,
type: 'GET' ,
//نوع نتیجه بازگشتی
dataType: 'html'

})
.success( function (result) {
  //زمانی که کدهای سمت سرور بدون خطا اجرا شده اند
  //این قسمت فراخوانی می‌شود و نتیجه اکشن درون متغیر
  //result
  //قرار می‌گیرد
  $( '#sectionContents' ).html(result);
})
.error( function (xhr, status) {
  alert(xhr.responseText);
});
  });
```

دقیقا کجای Index باید قرارداده بشه؟ سورس پروژه را هم ارسال می‌کردید خیلی خوب می‌شد .
ممنون.

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۶/۲۸ ۲۰:۴

نحوه استفاده از افزونه Firebug برای دیباگ برنامه‌های ASP.NET مبتنی بر jQuery

نویسنده: منیژه محمدی
تاریخ: ۱۳۹۲/۰۸/۱۶ ۱۰:۴۴

سلام
من این مثال را قدم به قدم انجام دادم ولی خطای 404 میده و localhost.../home/Details را پیدا نمی‌کند.
با اینکه من پارشیال ویو را در این تابع تعریف کردم ....

نویسنده: وحید نصیری

تاریخ: ۱۳۹۲/۰۸/۱۶ ۱۰:۵۴

- برنامه را با فایرباگ دیباگ کنید تا مسیرهای درخواستی و خطاها را بهتر مشاهده کنید.
- نباید آدرس‌ها را مستقیما و به صورت رشته‌ای تعریف کنید. اطلاعات بیشتر
- در مثال عنوان شده «partial/_ProductList» این مورد به معنای وجود یک پوشه partial در اینجا است (تصویر اول). البته غیر الزامی است ولی برای باز تولید این مثال باید به آن دقت داشته باشید.

نویسنده: منیژه محمدی
تاریخ: ۱۳۹۲/۰۸/۱۷ ۱۱:۳۶

با سلام
من همه موارد را خیلی چک کردم از اینکه مسیر یا نام را درست داده باشم
علت این بود که من یک [httppost] بالای ان ActionResult که میگفت پیدا نمیکنم اضافه کرده بودم برای همین ان را پیدا نمی‌کرد
حالا من میخام یک سری اطلاعات به این اکشن پست کنم اینطوری مشکل پیدا میکنم چطور میتونم ان را حل کنم ؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۸/۱۷ ۱۱:۴۸

type در $.ajax مثال اصلی بحث جاری، مساوی Get است؛ تبدیلش کنید به Post. کامنت اول این مطلب سه مثال مشابه دیگر را عنوان کرده. اکثر این‌ها از حالت Post استفاده کرده‌اند.

نویسنده: آروین
تاریخ: ۱۳۹۲/۰۸/۲۱ ۱۱:۱۹

با عرض سلام و خسته نباشید
این روش فرقش با jquery.load('url') چیه؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۸/۲۱ ۱۴:۲۴

load در پشت صحنه از همین متد ajax استفاده می‌کند. تمام متدهای کمکی از این دست فقط محصور کننده تابع ajax اصلی جی‌کوئری هستند.

نویسنده: رشنو
تاریخ: ۱۳۹۲/۱۲/۱۵ ۱۰:۳۶

با سلام و تشکر
لیست محصولات رو چطور باید پاس داد؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۲/۱۵ ۱۰:۵۸

به روش‌های متداول « بررسی نحوه انتقال اطلاعات از یک کنترلر به Viewهای مرتبط با آن »

عنوان: **طراحی ValidationAttribute دلخواه و هماهنگ سازی آن با ASP.NET MVC**
نویسنده: صابر فتح الهی
تاریخ: ۱۳۹۲/۰۷/۲۱ ۱۵:۳۵
آدرس: www.dotnettips.info
گروه‌ها: Entity framework, MVC, ASP.Net MVC, jQuery

در سری پست‌های آقای مهندس یوسف نژاد با عنوان Globalization در ASP.NET MVC روشی برای پیاده سازی کار با Resourceها در ASP.NET با استفاده از دیتابیس شرح داده شده است. یکی از کمبودهایی که در این روش وجود داشت عدم استفاده از این نوع Resourceها از طریق Attributeها در ASP.NET MVC بود. برای استفاده از این روش در یک پروژه به این مشکل برخورد کردم و پس از تحقیق و بررسی چند پست سرانجام در این پست پاسخ خود را پیدا کرده و با ترکیب این روش با روش آقای یوسف نژاد موفق به پیاده سازی Attribute دلخواه شدم.
در این پست و با استفاده از سری پست‌های آقای مهندس یوسف نژاد در این زمینه، یک Attribute جهت هماهنگ سازی با سیستم اعتبار سنجی ASP.NET MVC در سمت سرور و سمت کلاینت (با استفاده از jQuery Validation) بررسی خواهد شد.

قبل از شروع مطالعه سری پست‌های MVC و Entity Framework الزامی است.

برای انجام این کار ابتدا مدل زیر را در برنامه خود ایجاد می‌کنیم.

```
using System;

public class SampleModel

{

    public DateTime StartDate { get; set; }

    public string Data { get; set; }

    public int Id { get; set; }

}
```

با استفاده از این مدل در برنامه در زمان ثبت داده‌ها هیچ گونه خطایی صادر نمی‌شود. برای اینکه بتوان از سیستم خطای پیش فرض ASP.NET MVC کمک گرفت می‌توان مدل را به صورت زیر تغییر داد.

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

public class SampleModel
{
    [Required(ErrorMessage = "Start date is required")]
    public DateTime StartDate { get; set; }

    [Required(ErrorMessage = "Data is required")]
    public string Data { get; set; }

    public int Id { get; set; }
}
```

حال ویو این مدل را طراحی می‌کنیم.

```
@model SampleModel
@{
    ViewBag.Title = "Index";
    Layout = "~/Views/Shared/_Layout.cshtml";
}

<section>
    <header>
        <h3>SampleModel</h3>
```

```
    </header>
    @Html.ValidationSummary(true, null, new { @class = "alert alert-error alert-block" })

    @using (Html.BeginForm("SaveData", "Sample", FormMethod.Post))
    {
        <p>
            @Html.LabelFor(x => x.StartDate)
            @Html.TextBoxFor(x => x.StartDate)
            @Html.ValidationMessageFor(x => x.StartDate)
        </p>
        <p>
            @Html.LabelFor(x => x.Data)
            @Html.TextBoxFor(x => x.Data)
            @Html.ValidationMessageFor(x => x.Data)
        </p>
        <input type="submit" value="Save"/>
    }
</section>
```

و بخش کنترلر آن را به صورت زیر پیاده سازی می‌کنیم.

```
public class SampleController : Controller
    {
        //
        // GET: /Sample/

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

        public ActionResult SaveData(SampleModel item)
        {
            if (ModelState.IsValid)
            {
                //save data
            }
            else
            {
                ModelState.AddModelError("","لطفا خطاهای زیر را برطرف نمایید");
                RedirectToAction("Index", item);
            }
            return View("Index");
        }
    }
```

حال با اجرای این کد و زدن دکمه Save صفحه مانند شکل پایین خطاها را نمایش خواهد داد.

تا اینجای کار روال عادی همیشگی است. اما برای طراحی Attribute دلخواه جهت اعتبار سنجی (مثلا برای مجبور کردن کاربر به وارد کردن یک فیلد با پیام دلخواه و زبان دلخواه) باید یک کلاس جدید تعریف کرده و از کلاس RequiredAttribute ارث ببرد. در پارامتر ورودی این کلاس جهت کار با Resourceهای ثابت در نظر گرفته شده است اما برای اینکه فیلد دلخواه را از دیتابیس بخواند این روش جوابگو نیست. برای انجام آن باید کلاس RequiredAttribute بازنویسی شود.

کلاس طراحی شده باید به صورت زیر باشد:

```
public class VegaRequiredAttribute : RequiredAttribute, IClientValidatable
    {
#region Fields (2)

        private readonly string _resourceId;
        private String _resourceString = String.Empty;

#endregion Fields

#region Constructors (1)

        public VegaRequiredAttribute(string resourceId)
        {
            _resourceId = resourceId;
            ErrorMessage = _resourceId;
            AllowEmptyStrings = true;
        }

#endregion Constructors

#region Properties (1)

        public new String ErrorMessage
        {
            get { return _resourceString; }
            set { _resourceString = GetMessageFromResource(value); }
        }

#endregion Properties

#region Methods (2)
```

```
// Public Methods (1)

        public IEnumerable<ModelClientValidationRule> GetClientValidationRules(ModelMetadata metadata,
ControllerContext context)
        {

            yield return new ModelClientValidationRule
            {
                ErrorMessage = GetMessageFromResource(_resourceId),
                ValidationType = "required"
            };
        }
// Private Methods (1)

        private string GetMessageFromResource(string resourceId)
        {
            var errorMessage = HttpContext.GetGlobalResourceObject(_resourceId, "Yes") as string;
            return errorMessage ?? ErrorMessage;
        }

#endregion Methods
    }
```

در این کلاس دو نکته وجود دارد.

1- ابتدا دستور

```
HttpContext.GetGlobalResourceObject(_resourceId, "Yes") as string;
```

که عنوان کلید Resource را از سازنده کلاس گرفته (کد اقای یوسف نژاد) رشته معادل آن را از دیتابیس بازیابی میکند.

2- ارث بری از اینترفیس IClientValidatable، در صورتی که از این اینترفیس ارث بری نداشته باشیم. Validator طراحی شده در طرف کلاینت کار نمی‌کند. بلکه کاربر با کلیک بروی دکمه مورد نظر داده‌ها را به سمت سرور ارسال می‌کند. در صورت وجود خطا در پست بک خطا نمایش داده خواهد شد. اما با ارث بری از این اینترفیس و پیاده سازی متد GetClientValidationRules می‌توان تعریف کرد که در طرف کلاینت با استفاده از Unobtrusive jQuery پیام خطای مورد نظر به کنترل ورودی مورد نظر (مانند تکست باکس) اعمال می‌شود. مثلا در این مثال خصوصیت data-val-required به input هایی که قبلا در مدل ما Reqired تعریف شده اند اعمال می‌شود.

حال در مدل تعریف شده می‌توان به جای Required می‌توان از VegaRequiredAttribute مانند زیر استفاده کرد. (همراه با نام کلید مورد نظر در دیتابیس)

```
public class SampleModel
{
    [VegaRequired("RequiredMessage")]
    public DateTime StartDate { get; set; }

    [VegaRequired("RequiredMessage")]
    public string Data { get; set; }

    public int Id { get; set; }
}
```

ورودی Validator مورد نظر نام کلیدی است به زبان دلخواه که عنوان آن RequiredMessage تعریف شده است و مقدار آن در دیتابیس مقداری مانند " **تکمیل این فیلد الزامی است** " است. با این کار در زمان اجرا با استفاده از این ولیدتور ابتدا کلید مورد نظر با توجه به زبان فعلی از دیتابیس بازیابی شده و در متادیتابی مدل ما قرار می‌گیرد. به جای استفاده از Resourceها می‌توان پیام‌های خطای دلخواه را در دیتابیس ذخیره کرد و در مواقع ضروری جهت جلوگیری از تکرار از آنها استفاده نمود.

با اجرای برنامه اینبار خروجی به شکل زیر خواهد بود.

جهت فعال ساری اعتبار سنجی سمت کلاینت ابتدا باید اسکریپت‌های زیر به صفحه اضافه شود.

```
<script src="@Url.Content("~/Scripts/jquery-1.9.1.min.js")" type="text/javascript"></script>
<script src="@Url.Content("~/Scripts/jquery.validate.min.js")" type="text/javascript"></script>
<script src="@Url.Content("~/Scripts/jquery.validate.unobtrusive.min.js")"
type="text/javascript"></script>
```

سپس در فایل web.config تنظیمات زیر باید اضافه شود

```
<appSettings>
    <add key="ClientValidationEnabled" value="true" />
    <add key="UnobtrusiveJavaScriptEnabled" value="true"/>
</appSettings>
```

سپس برای اعمال Validator طراحی شده باید توسط کدهای جاوا اسکریپت زیر داده‌های مورد نیاز سمت کلاینت رجیستر شوند.

```
<script type="text/javascript">
        jQuery.validator.addMethod('required', function (value, element, params) {
            if (value == null | value == "") {
                return false;
            } else {
                return true;
            }

        }, '');

        jQuery.validator.unobtrusive.adapters.add('required', {}, function (options) {
            options.rules['required'] = true;
            options.messages['required'] = options.message;
        });
    </script>
```

البته برای مثال ما قسمت بالا به صورت پیش فرض رجیستر شده است اما در صورتی که بخواهید یک ولیدتور دلخواه و غیر استاندارد بنویسید روال کار به همین شکل است.

موفق و موید باشید.

منابع ^ و ^ و ^ و ^

نظرات خوانندگان

نویسنده: امیر خلیلی
تاریخ: ۱۳۹۲/۰۸/۰۴ ۸:۵۹

یک مشکل اساسی وجود داره
هنگامی که با استفاده از جاواسکریپت اعتبارسنجی انجام میشه کاربر یا اون شخصی که قراره کدهای مخرب را وارد کنه میتونه استفاده از فایل‌های JS را در مرورگش غیر فعال کنه

و اینگونه اعتبار سنجی انجام نمیشه !

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۸/۰۴ ۹:۹

اعتبارسنجی در ASP.NET MVC دو مرحله‌ای است. مرحله اول آن فقط سمت کاربر است. مرحله دوم به تفسیر IClientValidatable در سمت سرور ختم می‌شود.

عنوان: CheckBoxList برای فیلد Enum Flags مدل در ASP.Net MVC
نویسنده: پژمان پارسائی
تاریخ: ۱۳۹۲/۰۷/۳۰ ۲۳:۰
آدرس: www.dotnettips.info
گروه‌ها: C#, ASP.Net MVC, Enum, ASP.Net MVC 4, MVC

قبلا مطالبی در سایت راجع به نوع داده شمارشی یا Enum و همچنین CheckBoxList و RadioButtonList وجود دارد. اما در این مطلب قصد دارم تا یک روش متفاوت را برای تولید و بهره گیری از CheckBoxList با استفاده از نوع داده‌های شمارشی برای شما ارائه کنم.
فرض کنید بخواهید به کاربر این امکان را بدهید تا بتواند چندین گزینه را برای یک فیلد انتخاب کند. به عنوان یک مثال ساده فرض کنید گزینه ای از مدل، پارچه‌های مورد علاقه یک نفر هست. کاربر می‌تواند چندین پارچه را انتخاب کند. و این فرض را هم بکنید که به لیست پارچه‌ها گزینه دیگری اضافه نخواهد شد. پارچه (Fabric) را مثلا می‌توانیم به صورت زیر تقسیم بندی کنیم :

پنبه (Cotton)
ابریشم (Silk)
پشم (Wool)
ابریشم مصنوعی (Rayon)
پارچه‌های دیگر (Other)

با توجه به اینکه دیگر قرار نیست به این لیست گزینه دیگری اضافه شود می‌توانیم آنرا به صورت یک نوع داده شمارشی (Enum) تعریف کنیم. مثلا بدین صورت:

```
public enum Fabric
{
    [Description("پنبه")]
    Cotton,

    [Description("ابریشم")]
    Silk,

    [Description("پشم")]
    Wool,

    [Description("ابریشم مصنوعی")]
    Rayon,

    [Description("پارچه‌های دیگر")]
    Other
}
```

حال فرض کنید View Model زیر فیلدی از نوع نوع داده شمارشی Fabric دارد:

```
public class MyViewModel
{
    public Fabric Fabric { get; set; }
}
```

توجه داشته باشید که فیلد Fabric از کلاس MyViewModel باید چند مقدار را در خود نگهداری کند. یعنی می‌تواند هر کدام از گزینه‌های Cotton، Silk، Wool، Rayon، Other به صورت جداگانه یا ترکیبی باشد. اما در حال حاضر با توجه به اینکه یک فیلد Enum معمولی فقط می‌تواند یک مقدار را در خودش ذخیره کند قابلیت ذخیره ترکیبی مقادیر در فیلد Fabric از View Model بالا وجود ندارد.

اما راه حل این مشکل استفاده از پرچم (Flags) در تعریف نوع داده شمارشی هست. با استفاده از پرچم نوع داده شمارشی بالا به صورت زیر باید تعریف شود:

```
[Flags]
public enum Fabric
```

```
{
    [Description("پنبه")]
    Cotton = 1,

    [Description("ابریشم")]
    Silk = 2,

    [Description("پشم")]
    Wool = 4,

    [Description("ابریشم مصنوعی")]
    Rayon = 8,

    [Description("پارچه‌های دیگر")]
    Other = 128
}
```

همان طور که می‌بینید از عبارت [Flags] قبل از تعریف enum استفاده کرده ایم. همچنین هر کدام از مقادیر ممکن این نوع داده شمارشی با توانهایی از 2 تنظیم شده اند. در این صورت یک نمونه از این نوع داده می‌تواند چندین مقدار را در خودش ذخیره کند.

برای آشنایی بیشتر با این موضوع به کدهای زیر نگاه کنید:

```
Fabric cotWool = Fabric.Cotton | Fabric.Wool;
int cotWoolValue = (int) cotWool;
```

به وسیله عملگر | می‌توان چندین مقدار را در یک نمونه از نوع Fabric ذخیره کرد. مثلا متغیر cotWool هم دارای مقدار Fabric.Cotton و هم دارای مقدار Fabric.Wool هست. مقدار عددی معادل متغیر cotWool برابر 5 هست که از جمع مقدار عددی Fabric.Cotton و Fabric.Wool به دست آمده است.

حال فرض کنید فیلد Fabric از View Model ذکر شده (کلاس MyViewModel) را به صورت لیستی از چک باکس‌ها نمایش دهیم. مثل زیر:

شکل (الف)

سپس بخواهیم تا کاربر بعد از انتخاب گزینه‌های مورد نظرش از لیست بالا و پست کردن فرم مورد نظر، بایندر وارد عمل شده و فیلد Fabric را بر اساس گزینه هایی که کاربر انتخاب کرده مقداردهی کند.

برای این کار از پروژه MVC Enum Flags کمک خواهیم گرفت. این پروژه شامل یک Html Helper برای تبدیل یه Enum به یک CheckBoxList و همچنین شامل Model Binder مربوطه هست. البته بعضی از کدهای Html Helper آن احتیاج به تغییر داشت که

آنرا انجام دادم ولی بایندر آن بسیار خوب کار می‌کند.

خوب html helper مربوط به آن به صورت زیر می‌باشد:

```
public static IHtmlString CheckBoxesForEnumFlagsFor<TModel, TEnum>(this HtmlHelper<TModel> htmlHelper,
Expression<Func<TModel, TEnum>> expression)
{
    ModelMetadata metadata = ModelMetadata.FromLambdaExpression(expression, htmlHelper.ViewData);
    Type enumModelType = metadata.ModelType;

    // Check to make sure this is an enum.
    if (!enumModelType.IsEnum)
    {
        throw new ArgumentException("This helper can only be used with enums. Type used was: " +
enumModelType.FullName.ToString() + ".");
    }

    // Create string for Element.
    var sb = new StringBuilder();

    foreach (Enum item in Enum.GetValues(enumModelType))
    {
        if (Convert.ToInt32(item) != 0)
        {
            var ti = htmlHelper.ViewData.TemplateInfo;
            var id = ti.GetFullHtmlFieldId(item.ToString());

            //Derive property name for checkbox name
            var body = expression.Body as MemberExpression;
            var propertyName = body.Member.Name;
            var name = ti.GetFullHtmlFieldName(propertyName);

            //Get currently select values from the ViewData model
            TEnum selectedValues = expression.Compile().Invoke(htmlHelper.ViewData.Model);

            var label = new TagBuilder("label");
            label.Attributes["for"] = id;
            label.Attributes["style"] = "display: inline-block;";
            var field = item.GetType().GetField(item.ToString());

            // Add checkbox.
            var checkbox = new TagBuilder("input");
            checkbox.Attributes["id"] = id;
            checkbox.Attributes["name"] = name;
            checkbox.Attributes["type"] = "checkbox";
            checkbox.Attributes["value"] = item.ToString();

            if ((selectedValues as Enum != null) && ((selectedValues as Enum).HasFlag(item)))
            {
                checkbox.Attributes["checked"] = "checked";
            }
            sb.AppendLine(checkbox.ToString());

            // Check to see if DisplayName attribute has been set for item.
            var displayName = field.GetCustomAttributes(typeof(DisplayNameAttribute), true)
                .FirstOrDefault() as DisplayNameAttribute;
            if (displayName != null)
            {
                // Display name specified.  Use it.
                label.SetInnerText(displayName.DisplayName);
            }
            else
            {
                // Check to see if Display attribute has been set for item.
                var display = field.GetCustomAttributes(typeof(DisplayAttribute), true)
                    .FirstOrDefault() as DisplayAttribute;
                if (display != null)
                {
                    label.SetInnerText(display.Name);
                }
                else
                {
                    label.SetInnerText(item.ToDescription());
                }
            }
            sb.AppendLine(label.ToString());

            // Add line break.
            sb.AppendLine("<br />");
```

```
        }
    }

    return new HtmlString(sb.ToString());
}
```

در کدهای بالا از متد الحاقی ToDescription نیز برای تبدیل معادل انگلیسی به فارسی یک مقدار از نوع داده شمارشی استفاده کرده ایم.

```
public static string ToDescription(this Enum value)
{
    var attributes =
(DescriptionAttribute[])value.GetType().GetField(value.ToString()).GetCustomAttributes(typeof(Descripti
onAttribute), false);
    return attributes.Length > 0 ? attributes[0].Description : value.ToString();
}
```

برای استفاده از این Html Helper در View کد زیر را می‌نویسیم:

```
@Html.CheckBoxesForEnumFlagsFor(x => x.Fabric)
```

که باعث تولید خروجی که در تصویر (الف) نشان داده شد می‌شود. و همچنین مدل بایندر مربوط به آن به صورت زیر هست:

```
public class FlagEnumerationModelBinder : DefaultModelBinder
{
    public override object BindModel(ControllerContext controllerContext, ModelBindingContext
bindingContext)
    {
        if (bindingContext == null) throw new ArgumentNullException("bindingContext");

        if (bindingContext.ValueProvider.ContainsPrefix(bindingContext.ModelName))
        {
            var values = GetValue<string[]>(bindingContext, bindingContext.ModelName);

            if (values.Length > 1 && (bindingContext.ModelType.IsEnum &&
bindingContext.ModelType.IsDefined(typeof(FlagsAttribute), false)))
            {
                long byteValue = 0;
                foreach (var value in values.Where(v => Enum.IsDefined(bindingContext.ModelType, v)))
                {
                    byteValue |= (int)Enum.Parse(bindingContext.ModelType, value);
                }

                return Enum.Parse(bindingContext.ModelType, byteValue.ToString());
            }
            else
            {
                return base.BindModel(controllerContext, bindingContext);
            }
        }

        return base.BindModel(controllerContext, bindingContext);
    }

    private static T GetValue<T>(ModelBindingContext bindingContext, string key)
    {
        if (bindingContext.ValueProvider.ContainsPrefix(key))
        {
            ValueProviderResult valueResult = bindingContext.ValueProvider.GetValue(key);
            if (valueResult != null)
            {
                bindingContext.ModelState.SetModelValue(key, valueResult);
                return (T)valueResult.ConvertTo(typeof(T));
            }
        }
        return default(T);
    }
}
```

این مدل بایندر را باید به این صورت در متد Application_Start فایل Global.asax فراخوانی کنیم:

```
ModelBinders.Binders.Add(typeof(Fabric), new FlagEnumerationModelBinder());
```

مشاهده می‌کنید که در اینجا دقیقا مشخص کرده ایم که این مدل بایندر برای نوع داده شمارشی Fabric هست. اگر نیاز دارید تا این بایندر برای نوع داده‌های شمارشی دیگری نیز به کار رود نیاز هست تا این خط کد را برای هر کدام از آنها تکرار کنید. اما راه حل بهتر این هست که کلاسی به صورت زیر تعریف کنیم و تمامی نوع داده‌های شمارشی که باید از بایندر بالا استفاده کنند را در یک پراپرتی آن برگشت دهیم. مثلا بدین صورت:

```
public class ModelEnums
{
    public static IEnumerable<Type> Types
    {
        get
        {
            var types = new List<Type> { typeof(Fabric) };
            return types;
        }
    }
}
```

سپس به متد Application_Start رفته و کد زیر را اضافه می‌کنیم:

```
foreach (var type in ModelEnums.Types)
{
   ModelBinders.Binders.Add(type, new FlagEnumerationModelBinder())
}
```

اگر گزینه‌های پشم و ابریشم مصنوعی را از CheckBoxList تولید شده انتخاب کنیم، بدین صورت:

شکل (ب)

و سپس فرم را پست کنید، موردی شبیه زیر مشاهده می‌کنید:

```
[HttpPost]
public ActionResult Index(MyViewModel model)
{
    int value = (int) model.Fabric;
    return View(model);
}
```

model {MvcEnumFlagsPr
Fabric Wool | Rayon

شکل (ج)

همچنین مقدار عددی معادل در این جا برابر 12 می‌باشد که از جمع دو مقدار Wool و Rayon به دست آمده است. بدین ترتیب در یک فیلد از مدل، گزینه‌های انتخابی توسط کاربر قرار گرفته شده اند.

پروژه مربوط به این مثال را از لینک زیر دریافت کنید:
MvcEnumFlagsProjectSample.zip

پی نوشت : پوشه‌های bin و obj و packages جهت کاهش حجم پروژه از آن حذف شده اند. برای بازسازی پوشه packages لطفا به مطلب بازسازی کامل پوشه packages بسته‌های NuGet به صورت خودکار مراجعه کنید.

نظرات خوانندگان

نویسنده: علی
تاریخ: ۱۳۹۲/۰۸/۰۱ ۱۸:۵۱

من یک گرید تلریک دارم که یک فیلد چکباکس کنارش هست و مثلا لیستی از یوزرها رو این گرید شامل میشه .
آیتم‌های انتخابی رو چطوری میتونم به کنترلرم ارسال کنم (ورودی کنترلرمو چی بگیرم)...

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۸/۰۱ ۱۸:۵۸

سؤال شما بیشتر به مطلب « CheckBoxList در ASP.NET MVC » مرتبط است تا enum ایی که ویژگی flag دارد. ضمنا گرید تلریک مستندات خوبی دارد که بهتر است به آن مراجعه کنید (مثال Ajax CheckBoxes هست که کدهای View و کنترلر آن نیز پیوست شدند).

نویسنده: علیرضا
تاریخ: ۱۳۹۲/۰۸/۰۲ ۱۲:۱۵

استفاده از Description برای Enumها جالب ولی به نظر من کمی مشکل سازه و مشکل اصلی اون Code Wired کردن شرح هر جزء Enum هست. اگر با یک پروژه دوزبانه طرف باشیم چی؟ اگر کلا استراتژی ترجمه رشته‌ها در پروژه ما این طور باشه که از منابع خارجی مثل DB یا اسمبلی‌های Resource استفاده کنیم چی؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۸/۰۲ ۱۲:۳۲

مراجعه کنید به پیشنیازهای این مباحث تکمیلی. مانند:
تهیه سایت‌های چند زبانه و بومی سازی نمایش اطلاعات در ASP.NET MVC
Globalization در ASP.NET MVC
اولی در مورد کار با ریسورس‌ها است و بومی سازی ویژگی‌ها نیز در آن لحاظ شده و دومی تهیه یک فریم ورک است برای کار با بانک اطلاعاتی و تامین منبع داده از این طریق

عنوان: **استفاده از DataAnnotations در EF DataBase First**
نویسنده: حسین حسینی 128
تاریخ: ۱۳۹۲/۰۸/۲۸ ۱۲:۰
آدرس: www.dotnettips.info
گروه‌ها: Entity framework, ASP.Net MVC, EF Database First

همانطور که میدانیم DataAnnotations برای فیلدهای مدل‌ها در MVC وقتی که از EF Code First استفاده کنیم کار ما را برای اعتبارسنجی بسیار ساده میکنند. اما وضع در EF Database First متفاوت است زیرا اگر مدلی را که برنامه برایمان میسازد را به روز کنیم (توسط Updata Model) تمام اعتبارسنجی هایی که نوشته بودیم پاک شده و مدل خام دوباره برای ما تولید میشود . برای رفع این مشل باید از PartialClass استفاده نماییم تا بتوانیم همیشه اعتبارسنجی‌ها را داشته باشیم :

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

namespace MvcApplication1.Models
{
    [MetadataType(typeof(MovieMetaData))]
    public partial class Movie
    {
    }


    public class MovieMetaData
    {
        [Required]
        public object Title { get; set; }

        [Required]
        [StringLength(5)]
        public object Director { get; set; }


        [DisplayName("Date Released")]
        [Required]
        public object DateReleased { get; set; }
    }

}
```

لینک مطلب اصلی : http://www.asp.net/mvc/tutorials/older-versions/models-(data)/validation-with-the-data-annotation-validators-cs

نظرات خوانندگان

نویسنده: لیلا
تاریخ: ۱۳۹۲/۰۸/۲۹ ۱۹:۲۳

با تشکر
اگر دیتابیس تغییر کند باید PartialClass تغییر را دستی تغییر بدهیم؟ اگر پاسخ بله می‌باشد راهی وجود دارد برای اعمال تغییر ات به صورت خودکار؟
ایا توصیه می‌شود از view model به جای PartialClass برای اعتبارسنجی استفاده شود؟

نویسنده: حسین حسینی 128
تاریخ: ۱۳۹۲/۰۸/۲۹ ۲۳:۹

بله باید دستی تغییر کند ، راه خودکاری نمیدانم متاسفانه
کلا بهتر است از viewmodel به جای استفاده از خود مدل استفاده کرد ولی به هرحال viewmodel نیز پس از تغییر دیتابیس باید دستی تغییر کند

نویسنده: bahman
تاریخ: ۱۳۹۲/۰۹/۱۲ ۱۸:۲۳

سلام.
به خاطر مطلب خوبتون تشکر میکنم.
من یه پروژه 2 لایه دارم که مدل در لایه DLL قرار گرفته.
و یک لایه که پروژه MVC داخل اون قرار گافته.
میخواستم بدنم که متادیتا باید داخل کدوم لایه نوشته بشه؟

نویسنده: حسین حسینی 128
تاریخ: ۱۳۹۲/۱۰/۱۷ ۱۷:۵۹

متادیتا داخل لایه DLL باشه بهتره

عنوان: CSS پویا در ASP.NET MVC
نویسنده: ناصر طاهری
تاریخ: ۱۳۹۲/۰۹/۱۲ ۲۲:۵
آدرس: www.dotnettips.info
برچسب‌ها: ASP.Net MVC, Dynamic Css

سناریو هایی وجود دارد که نیاز است مشتری ، خود شیوه نامه هایی (CSS) را برای قسمت‌های مختلف سایت انتخاب کند. برای مثال تنظیماتی را برای منوی سایت در نظر گرفته ایم که مشتری بتواند رنگ و قلم و ... را متناسب با سلیقه‌ی خود تغییر دهد و یا یک قسمت کلی برای اعمال شیوه نامه‌ها به سایت ایجاد کرده ایم که در همه‌ی قسمت‌های سایت اعمال شود. بدین شکل در صورتی که مشتری، اطلاعات اندکی هم در مورد CSS داشته باشد میتواند ظاهر سایت خود را به آسانی تغییر دهد و تا حدودی بار را از روی دوش پشتیبان سایت بر میدارد.
و برای همه‌ی این‌ها نیاز است تا فیلدی در دیتابیس برای ذخیره‌ی شیوه نامه‌های مشتری ایجاد شود و یا در یک فایل متنی ذخیره شود که بسته به سیاست برنامه نویس دارد.
در این مطلب تصمیم داریم این سناریو را به صورت ساده در یک پروژه ASP.NET MVC پیاده سازی کنیم.
ایتدا یک پروژه از نوع ASP.NET MVC 4 ایجاد میکنیم. سپس قطعه کد زیر را به فایل _Layout.cshtml موجود در مسیر Views/Shared به صورت زیر اضافه میکنیم :

```
<head>
    @*سایر شیوه نامه‌ها و اسکریپت ها*@
    @RenderSection("styles", required: false)
</head>
```

RenderSection این امکان را میدهد که ما بتوانیم شیوه نامه‌ی دیگری را در صفحات دیگر به MasterPage خود تزریق کنیم.
سپس کنترلری به نام Home ایجاد میکنیم :

```
namespace DynamicCssExample.Controllers
{
    public class HomeController : Controller
    {

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

        public string GetStyle()
        {
            Response.ContentType = "text/css";
            //در این قسمت میتوانیم به دیتابیس متصل شویم و شیوه نامه‌ی مورد نظر را واکشی کنیم و بازگشت
دهیم
            return "h2{color:yellow}";
        }

    }
}
```

در اینجا یک متد به نام GetStyle داریم که وظیفه‌ی بازگشت یک رشته را بر عهده دارد و نوع این مقدار بازگشتی از نوع text/css است و میتواند شیوه نامه هایی که در دیتابیس و یا یک فایل متنی ذخیره کرده ایم را واکشی و به استریم ارسال کند.
در انتها برای استفاده از این متد کافی است در View مربوطه بدین شکل عمل کنیم :

```
@{
    ViewBag.Title = "Index";
    Layout = "~/Views/Shared/_Layout.cshtml";
}

@section styles
{
    <link rel="stylesheet" href="@Url.Action("GetStyle", "Home")" type="text/css"/>
}

<h2>Index</h2>
```

در سکشن Style لینک شیوه نامه ای تعریف کرده ایم که ویژگی href آن به اکشن GetStyle در کنترلر Home اشاره میکند و این اکشن نیز محتوای شیوه نامه را برگشت میدهد.

مثال این مطلب : DynamicCssExample.zip به همراه بازسازی کامل پوشه packages بسته‌های NuGet به صورت خودکار

نظرات خوانندگان

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۹/۱۲ ۲۲:۵۴

این روش را می‌شود کمی بهبود داد؛ برای اینکه بشود داخل فایل CSS با کدهای Razor هم کار کرد (یعنی چیزی شبیه به LESS اما پیاده سازی شده با Razor و تفسیر شده توسط موتور آن؛ مانند یک View یا Partial View معمولی و کامل. حتی می‌شود داخل آن if و else یا حلقه نوشت):

```
public class CSS
    {
        public string Color { set; get; }
    }
```

```
public ActionResult GetDynamicStyle()
        {
            var color = "White";
            if (DateTime.Now.Hour > 18 || DateTime.Now.Hour < 8)
            {
                color = "Black";
            }

            this.Response.ContentType = "text/css";
            return PartialView(viewName: "~/Views/Home/_CSS.cshtml", model: new CSS { Color = color });
        }
```

با این محتوای Views/Home/_CSS.cshtml :

```
@model DynamicMvcCSS.Controllers.CSS

.foo {
    color: @Model.Color;
}
```

نویسنده: ناصر طاهری
تاریخ: ۱۳۹۲/۰۹/۱۲ ۲۳:۴۲

ممنون خیلی بهتر شد. برای موارد ثابت این روش خیلی شکیلتر و قانونمندتره مثل تنظیمات منوها و بعضی قسمت‌ها که فقط یک سری موارد خاص رو اجازه‌ی تغییر داره. اما اگر بخواهیم به مشتری اجازه بدهیم که هر قسمتی که دوست دارد را تغییر دهد، یعنی شیوه نامه ای که ایجاد میکنه بر کل سایت تاثیر بزاره ،دیگه فکر نکنم بتونیم اینطور قانونمند عمل کنیم درسته؟ راهی هم برای این مسئله وجود داره؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۹/۱۲ ۲۳:۴۶

- روش شما برای بازگشت یک رشته متغیر از پیش تعریف شده خوب است.
+ نیازی نیست کل CSS را در اختیار کاربر برای ویرایش قرار داد. قسمت‌هایی را که قرار است تغییر کنند به صورت فیلد دربیارید تا کاربر بتواند مقدار دهی کند. بعد با روش بالا (که model آن به صورت پویا قابل مقدار دهی است) می‌شود در فایل razor نهایی مثل یک view عناصر را مقدار دهی و استفاده کرد.
- ضمنا استفاده از Output cache هم توصیه می‌شود.

عنوان: **افزودن ASP.NET Identity به یک پروژه Web Forms**
نویسنده: آرمین ضیاء
تاریخ: ۱۳۹۲/۱۰/۱۸ ۱۳:۴۵
آدرس: www.dotnettips.info
گروه‌ها: ASP.Net, Entity framework, Security, ASP.Net MVC, ASP.NET Identity, Forms Authentication

با نصب و اجرای Visual Studio 2013 Express for Web یا Visual Studio 2013 شروع کنید.

یک پروژه جدید بسازید (از صفحه شروع یا منوی فایل)

گزینه Visual C# و سپس **ASP.NET Web Application** را انتخاب کنید. نام پروژه را به "WebFormsIdentity" تغییر داده و OK کنید.

در دیالوگ جدید ASP.NET گزینه **Empty** را انتخاب کنید.

دقت کنید که دکمه **Change Authentication** غیرفعال است و هیچ پشتیبانی ای برای احراز هویت در این قالب پروژه وجود ندارد.

افزودن پکیج‌های ASP.NET Identity به پروژه

روی نام پروژه کلیک راست کنید و گزینه **Manage NuGet Packages** را انتخاب کنید. در قسمت جستجوی دیالوگ باز شده عبارت "Identity.E" را وارد کرده و این پکیج را نصب کنید.

دقت کنید که نصب کردن این پکیج وابستگی‌ها را نیز بصورت خودکار نصب می‌کند: Entity Framework و ASP.NET Idenity Core.

افزودن فرم‌های وب لازم برای ثبت نام کاربران

یک فرم وب جدید بسازید.

در دیالوگ باز شده نام فرم را به **Register** تغییر داده و تایید کنید.

فایل ایجاد شده جدید را باز کرده و کد Markup آن را با قطعه کد زیر جایگزین کنید.

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Register.aspx.cs"
Inherits="WebFormsIdentity.Register" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
```

```
</head>
<body style="
    <form id="form1" runat="server">
    <div>
        <h4 style="Register a new user</h4>
        <hr />
        <p>
            <asp:Literal runat="server" ID="StatusMessage" />
        </p>
        <div style="margin-bottom:10px">
            <asp:Label runat="server" AssociatedControlID="UserName">User name</asp:Label>
            <div>
                <asp:TextBox runat="server" ID="UserName" />
            </div>
        </div>
        <div style="margin-bottom:10px">
            <asp:Label runat="server" AssociatedControlID="Password">Password</asp:Label>
            <div>
                <asp:TextBox runat="server" ID="Password" TextMode="Password" />
            </div>
        </div>
        <div style="margin-bottom:10px">
            <asp:Label runat="server" AssociatedControlID="ConfirmPassword">Confirm
password</asp:Label>
            <div>
                <asp:TextBox runat="server" ID="ConfirmPassword" TextMode="Password" />
            </div>
        </div>
        <div>
            <div>
                <asp:Button runat="server" OnClick="CreateUser_Click" Text="Register" />
            </div>
        </div>
    </div>
    </form>
</body>
</html>
```

این تنها یک نسخه ساده شده *Register.aspx* است که از چند فیلد فرم و دکمه ای برای ارسال آنها به سرور استفاده می‌کند.

فایل کد این فرم را باز کرده و محتویات آن را با قطعه کد زیر جایگزین کنید.

```
using Microsoft.AspNet.Identity;
using Microsoft.AspNet.Identity.EntityFramework;
using System;
using System.Linq;

namespace WebFormsIdentity
{
   public partial class Register : System.Web.UI.Page
   {
      protected void CreateUser_Click(object sender, EventArgs e)
      {
         // Default UserStore constructor uses the default connection string named: DefaultConnection
         var userStore = new UserStore<IdentityUser>();
         var manager = new UserManager<IdentityUser>(userStore);

         var user = new IdentityUser() { UserName = UserName.Text };
         IdentityResult result = manager.Create(user, Password.Text);

         if (result.Succeeded)
         {
            StatusMessage.Text = string.Format("User {0} was created successfully!", user.UserName);
         }
         else
         {
            StatusMessage.Text = result.Errors.FirstOrDefault();
         }
      }
   }
}
```

کد این فرم نیز نسخه ای ساده شده است. فایلی که بصورت خودکار توسط VS برای شما ایجاد می‌شود متفاوت است.

کلاس *IdentityUser* پیاده سازی پیش فرض EntityFramework از قرارداد IUser است. قرارداد IUser تعریفات حداقلی یک کاربر در ASP.NET Identity Core را در بر می‌گیرد.

کلاس *UserStore* پیاده سازی پیش فرض EF از یک فروشگاه کاربر (user store) است. این کلاس چند قرارداد اساسی ASP.NET Identity Core را پیاده سازی می‌کند: *IUserClaimStore*, *IUserLoginStore*, *IUserStore* و *IUserRoleStore*.

کلاس *UserManager* دسترسی به APIهای مربوط به کاربران را فراهم می‌کند. این کلاس تمامی تغییرات را بصورت خودکار در *UserStore* ذخیره می‌کند.

کلاس *IdentityResult* نتیجه یک عملیات هویتی را معرفی می‌کند (identity operations).

پوشه **App_Data** را به پروژه خود اضافه کنید.

فایل *Web.config* پروژه را باز کنید و رشته اتصال جدیدی برای دیتابیس اطلاعات کاربران اضافه کنید. این دیتابیس در زمان اجرا (runtime) بصورت خودکار توسط EF ساخته می‌شود. این رشته اتصال شبیه به رشته اتصالی است که هنگام ایجاد پروژه بصورت

خودکار برای شما تنظیم می‌شود.

```
<?xml version="1.0" encoding="utf-8"?>
<!--
  For more information on how to configure your ASP.NET application, please visit
  http://go.microsoft.com/fwlink/?LinkId=169433
  -->
<configuration>
  <configSections>
    <!-- For more information on Entity Framework configuration, visit
http://go.microsoft.com/fwlink/?LinkID=237468 -->
    <section name="entityFramework"
type="System.Data.Entity.Internal.ConfigFile.EntityFrameworkSection, EntityFramework, Version=6.0.0.0,
Culture=neutral, PublicKeyToken=b77a5c561934e089" requirePermission="false" />
  </configSections>
   <connectionStrings>
      <add name="DefaultConnection" connectionString="Data
Source=(LocalDb)\v11.0;AttachDbFilename=|DataDirectory|\WebFormsIdentity.mdf;Initial
Catalog=WebFormsIdentity;Integrated Security=True"
            providerName="System.Data.SqlClient" />
   </connectionStrings>
  <system.web>
    <compilation debug="true" targetFramework="4.5" />
    <httpRuntime targetFramework="4.5" />
  </system.web>
  <entityFramework>
    <defaultConnectionFactory type="System.Data.Entity.Infrastructure.LocalDbConnectionFactory,
EntityFramework">
      <parameters>
        <parameter value="v11.0" />
      </parameters>
    </defaultConnectionFactory>
    <providers>
      <provider invariantName="System.Data.SqlClient"
type="System.Data.Entity.SqlServer.SqlProviderServices, EntityFramework.SqlServer" />
    </providers>
  </entityFramework>
</configuration>
```

همانطور که مشاهده می‌کنید نام این رشته اتصال *DefaultConnection* است.

روی فایل *Register.aspx* کلیک راست کنید و گزینه **Set As Start Page** را انتخاب کنید. اپلیکیشن خود را با کلیدهای ترکیبی Ctrl + F5 اجرا کنید که تمام پروژه را کامپایل نیز خواهد کرد. یک نام کاربری و کلمه عبور وارد کنید و روی **Register** کلیک کنید.

ASP.NET Identity از اعتبارسنجی نیز پشتیبانی می‌کند، مثلا در این مرحله می‌توانید از اعتبارسنج هایی که توسط ASP.NET Identity Core عرضه می‌شوند برای کنترل رفتار فیلدهای نام کاربری و کلمه عبور استفاده کنید. اعتبارسنج پیش فرض کاربران (User) که UserValidator نام دارد خاصیتی با نام AllowOnlyAlphanumericUserNames دارد که مقدار پیش فرضش هم true است. اعتبارسنج پیش فرض کلمه عبور (MinimumLengthValidator) اطمینان حاصل می‌کند که کلمه عبور حداقل 6 کاراکتر باشد. این اعتبارسنج‌ها بصورت propertyها در کلاس UserManager تعریف شده اند و می‌توانید آنها را overwrite کنید و اعتبارسنجی سفارشی خود را پیاده کنید. از آنجا که الگوی دیتابیس سیستم عضویت توسط Entity Framework مدیریت می‌شود، روی الگوی دیتابیس کنترل کامل دارید، پس از Data Annotations نیز می‌توانید استفاده کنید.

تایید دیتابیس LocalDbIdentity که توسط EF ساخته می‌شود

از منوی **View** گزینه **Server Explorer** را انتخاب کنید.

MvcAuth - Microsoft Visual Studio
FILE
EDIT
VIEW
PROJECT
BUILD
DEBUG
TEAM
TOOLS
Solution Explorer Ctrl+Alt+L
Team Explorer Ctrl+\, Ctrl+M
Server Explorer Ctrl+Alt+S
Architecture Explorer Ctrl+\, Ctrl+R
SQL Server Object Explorer Ctrl+\, Ctrl+S
Bookmark Window Ctrl+K, Ctrl+W
Call Hierarchy Ctrl+Alt+K
Class View Ctrl+Shift+C
Code Definition Window Ctrl+\, D
Document Outline Ctrl+Alt+T
Object Browser Ctrl+Alt+J
Error List Ctrl+\, E
Output Ctrl+Alt+O
Start Page
Task List Ctrl+\, T
Toolbox Ctrl+Alt+X
Notifications Ctrl+W, N
Find Results
Other Windows
Toolbars
Full Screen Shift+Alt+Enter
All Windows Shift+Alt+M
Properties Window F4
Property Pages Shift+F4

گره **DefaultConnection (WebFormsIdentity)** و سپس **Tables** را باز کنید. روی جدول **AspNetUsers** کلیک راست کرده و **Show Table Data** را انتخاب کنید.

پیکربندی اپلیکیشن برای استفاده از احراز هویت OWIN

تا این مرحله ما تنها امکان ایجاد حساب‌های کاربری را فراهم کرده ایم. حال نیاز داریم امکان احراز هویت کاربران برای ورود آنها به سایت را فراهم کنیم. ASP.NET Identity برای احراز هویت مبتنی بر فرم (forms authentication) از OWIN Authentication استفاده می‌کند. OWIN Cookie Authentication مکانیزمی برای احراز هویت کاربران بر اساس cookieها و claimها است (claims-based). این مکانیزم می‌تواند توسط Entity Framework روی OWIN یا IIS استفاده شود.

با چنین مدلی، می‌توانیم از پکیج‌های احراز هویت خود در فریم ورک‌های مختلفی استفاده کنیم، مانند ASP.NET MVC و ASP.NET Web Forms. برای اطلاعات بیشتر درباره پروژه Katana و نحوه اجرای آن بصورت Host Agnostic به لینک Getting Started with the Katana Project مراجعه کنید.

نصب پکیج‌های احراز هویت روی پروژه

روی نام پروژه خود کلیک راست کرده و **Manage NuGet Packages** را انتخاب کنید. در قسمت جستجوی دیالوگ باز شده عبارت "*Identity.Owin*" را وارد کنید و این پکیج را نصب کنید.

به دنبال پکیجی با نام *Microsoft.Owin.Host.SystemWeb* بگردید و آن را نیز نصب کنید.

پکیج **Microsoft.Aspnet.Identity.Owin** حاوی یک سری کلاس Owin Extension است و امکان مدیریت و پیکربندی OWIN Authentication در پکیج‌های ASP.NET Identity Core را فراهم می‌کند.

پکیج **Microsoft.Owin.Host.SystemWeb** حاوی یک سرور OWIN است که اجرای اپلیکیشن‌های مبتنی بر OWIN را روی IIS و با استفاده از ASP.NET Request Pipeline ممکن می‌سازد. برای اطلاعات بیشتر به OWIN Middleware in the IIS integrated pipeline مراجعه کنید.

افزودن کلاس‌های پیکربندی Startup و Authentication

روی پروژه خود کلیک راست کرده و گزینه **Add** و سپس **Add New Item** را انتخاب کنید. در قسمت جستجوی دیالوگ باز شده عبارت "owin" را وارد کنید. نام کلاس را "*Startup*" تعیین کرده و تایید کنید.

فایل Startup.cs را باز کنید و قطعه کد زیر را با محتویات آن جایگزین کنید تا احراز هویت OWIN Cookie Authentication پیکربندی شود.

```
using Microsoft.AspNet.Identity;
using Microsoft.Owin;
using Microsoft.Owin.Security.Cookies;
using Owin;

[assembly: OwinStartup(typeof(WebFormsIdentity.Startup))]

namespace WebFormsIdentity
{
   public class Startup
   {
      public void Configuration(IAppBuilder app)
      {
         // For more information on how to configure your application, visit
http://go.microsoft.com/fwlink/?LinkID=316888
         app.UseCookieAuthentication(new CookieAuthenticationOptions
         {
            AuthenticationType = DefaultAuthenticationTypes.ApplicationCookie,
            LoginPath = new PathString("/Login")
         });
      }
   }
}
```

این کلاس حاوی خاصیت OwinAttribute است که کلاس راه انداز OWIN را نشانه گذاری می‌کند. هر اپلیکیشن OWIN یک کلاس راه انداز (startup) دارد که توسط آن می‌توانید کامپوننت‌های application pipeline را مشخص کنید. برای اطلاعات بیشتر درباره این مدل، به OWIN Startup Class Detection مراجعه فرمایید.

افزودن فرم‌های وب برای ثبت نام و ورود کاربران

فایل *Register.cs* را باز کنید و قطعه کد زیر را وارد کنید. این قسمت پس از ثبت نام موفقیت آمیز کاربر را به سایت وارد می‌کند.

```
using Microsoft.AspNet.Identity;
using Microsoft.AspNet.Identity.EntityFramework;
using Microsoft.Owin.Security;
using System;
using System.Linq;
using System.Web;

namespace WebFormsIdentity
{
   public partial class Register : System.Web.UI.Page
   {
      protected void CreateUser_Click(object sender, EventArgs e)
      {
         // Default UserStore constructor uses the default connection string named: DefaultConnection
         var userStore = new UserStore<IdentityUser>();
         var manager = new UserManager<IdentityUser>(userStore);
         var user = new IdentityUser() { UserName = UserName.Text };

         IdentityResult result = manager.Create(user, Password.Text);

         if (result.Succeeded)
         {
            var authenticationManager = HttpContext.Current.GetOwinContext().Authentication;
            var userIdentity = manager.CreateIdentity(user,
DefaultAuthenticationTypes.ApplicationCookie);
            authenticationManager.SignIn(new AuthenticationProperties() { }, userIdentity);
            Response.Redirect("~/Login.aspx");
         }
         else
         {
            StatusMessage.Text = result.Errors.FirstOrDefault();
         }
      }
   }
}
```

از آنجا که ASP.NET Identity و OWIN Cookie Authentication هر دو مبتنی بر Claims هستند، فریم ورک از برنامه نویس اپلیکیشن انتظار دارد تا برای کاربر یک آبجکت از نوع ClaimsIdentity تولید کند. این آبجکت تمام اطلاعات اختیارات کاربر را در بر می‌گیرد، مثلا اینکه کاربر به چه نقش هایی تعلق دارد. همچنین در این مرحله می‌توانید اختیارات (Claims) جدیدی به کاربر اضافه کنید.

شما با استفاده از AuthenticationManager که متعلق به OWIN است می‌توانید کاربر را به سایت وارد کنید. برای این کار شما متد SignIn را فراخوانی می‌کنید و آبجکتی از نوع ClaimsIdentity را به آن پاس می‌دهید. این کد کاربر را به سایت وارد می‌کند و یک کوکی برای او می‌سازد. این فراخوانی معادل همان FormAuthentication.SetAuthCookie است که توسط ماژول FormsAuthentication استفاده می‌شود.

روی پروژه خود کلیک راست کرده، فرم وب جدیدی با نام **Login** بسازید.

فایل *Login.aspx* را باز کنید و کد Markup آن را مانند قطعه کد زیر تغییر دهید.

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Login.aspx.cs"
Inherits="WebFormsIdentity.Login" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
   <title></title>
</head>
<body style="font-family: Arial, Helvetica, sans-serif; font-size: small">
   <form id="form1" runat="server">
      <div>
```

```
         <h4 style="font-size: medium">Log In</h4>
         <hr />
         <asp:PlaceHolder runat="server" ID="LoginStatus" Visible="false">
            <p>
               <asp:Literal runat="server" ID="StatusText" />
            </p>
         </asp:PlaceHolder>
         <asp:PlaceHolder runat="server" ID="LoginForm" Visible="false">
            <div style="margin-bottom: 10px">
               <asp:Label runat="server" AssociatedControlID="UserName">User name</asp:Label>
               <div>
                  <asp:TextBox runat="server" ID="UserName" />
               </div>
            </div>
            <div style="margin-bottom: 10px">
               <asp:Label runat="server" AssociatedControlID="Password">Password</asp:Label>
               <div>
                  <asp:TextBox runat="server" ID="Password" TextMode="Password" />
               </div>
            </div>
            <div style="margin-bottom: 10px">
               <div>
                  <asp:Button runat="server" OnClick="SignIn" Text="Log in" />
               </div>
            </div>
         </asp:PlaceHolder>
         <asp:PlaceHolder runat="server" ID="LogoutButton" Visible="false">
            <div>
               <div>
                  <asp:Button runat="server" OnClick="SignOut" Text="Log out" />
               </div>
            </div>
         </asp:PlaceHolder>
      </div>
   </form>
</body>
</html>
```

محتوای فایل *Login.aspx.cs* را نیز مانند لیست زیر تغییر دهید.

```
using Microsoft.AspNet.Identity;
using Microsoft.AspNet.Identity.EntityFramework;
using Microsoft.Owin.Security;
using System;
using System.Web;
using System.Web.UI.WebControls;

namespace WebFormsIdentity
{
   public partial class Login : System.Web.UI.Page
   {
      protected void Page_Load(object sender, EventArgs e)
      {
         if (!IsPostBack)
         {
            if (User.Identity.IsAuthenticated)
            {
               StatusText.Text = string.Format("Hello {0}!", User.Identity.GetUserName());
               LoginStatus.Visible = true;
               LogoutButton.Visible = true;
            }
            else
            {
               LoginForm.Visible = true;
            }
         }
      }

      protected void SignIn(object sender, EventArgs e)
      {
         var userStore = new UserStore<IdentityUser>();
         var userManager = new UserManager<IdentityUser>(userStore);
         var user = userManager.Find(UserName.Text, Password.Text);

         if (user != null)
         {
            var authenticationManager = HttpContext.Current.GetOwinContext().Authentication;
            var userIdentity = userManager.CreateIdentity(user,
```

```
DefaultAuthenticationTypes.ApplicationCookie);

            authenticationManager.SignIn(new AuthenticationProperties() { IsPersistent = false },
userIdentity);
            Response.Redirect("~/Login.aspx");
         }
         else
         {
            StatusText.Text = "Invalid username or password.";
            LoginStatus.Visible = true;
         }
      }

      protected void SignOut(object sender, EventArgs e)
      {
         var authenticationManager = HttpContext.Current.GetOwinContext().Authentication;
         authenticationManager.SignOut();
         Response.Redirect("~/Login.aspx");
      }
   }
}
```

متد Page_Load حالا وضعیت کاربر جاری را بررسی می‌کند و بر اساس وضعیت Context.User.Identity.IsAuthenticated تصمیم گیری می‌کند.

نمایش نام کاربر جاری: فریم ورک ASP.NET Identity روی System.Security.Principal.Identity متدهایی نوشته است که به شما امکان دریافت نام و شناسه کاربر جاری را می‌دهد. این متدها در اسمبلی Microsoft.AspNet.Identity.Core وجود دارند. این متدها جایگزین HttpContext.User.Identity.Name هستند.

متد SignIn

این متد، متد CreateUser_Click را که پیشتر بصورت خودکار ایجاد شده جایگزین می‌کند و پس از ایجاد موفقیت آمیز حساب کاربری، کاربر جاری را به سایت وارد می‌کند. فریم ورک OWIN متدهایی روی System.Web.HttpContext افزوده است که به شما این امکان را می‌دهند که یک ارجاع از نوع IOwinContext بگیرید. این متدها در اسمبلی Microsoft.Owin.Host.SystemWeb وجود دارند. کلاس OwinContext خاصیتی از نوع IAuthenticationManager دارد که امکانات احراز هویت موجود برای درخواست جاری را معرفی می‌کند.

پروژه را با Ctrl + F5 اجرا کنید و کاربر جدیدی بسازید. پس از وارد کردن نام کاربری و کلمه عبور و کلیک کردن دکمه Register باید بصورت خودکار به سایت وارد شوید و نام خود را مشاهده کنید.

همانطور که مشاهده می‌کنید در این مرحله حساب کاربری جدید ایجاد شده و به سایت وارد شده اید. روی Log out کلیک کنید تا از سایت خارج شوید. پس از آن باید به صفحه ورود هدایت شوید.
حالا یک نام کاربری یا کلمه عبور نامعتبر وارد کنید و روی **Log in** کلیک کنید.

متد UserManager.Find مقدار null بر می‌گرداند، بنابراین پیام خطای " *Invalid username or password* " نمایش داده خواهد شد.

http://localhost:41969/Login.a
Log In
Invalid username or password.
User name
RaquelS
Password
Log in

نظرات خوانندگان

نویسنده: Programmer
تاریخ: ۱۳۹۲/۱۰/۱۹ ۱۴:۴۱

با عرض سلام و تشکر بابت ترجمه روانتون. خیلی وقت بود که منتظر همچین پستی بودم. اینکه بشه با EF عملیات احراز هویت رو با مکانیسمی قویتر از Memberdship سابق انجام داد. اگر ممکنه همین مثال رو در قالب پروژه MVC انجام بدید. ممنون

نویسنده: آرمین ضیاء
تاریخ: ۱۳۹۲/۱۰/۱۹ ۱۸:۹

سلام، متشکرم.
ASP.NET Identity بصورت پیش فرض در قالب پروژه‌های VS 2013 استفاده میشه. در پست‌های قبلی بیشتر درباره این فریم بحث شده که می‌تونید مراجعه کنید. برای اطلاعات بیشتر به ASP.NET Identity سر بزنید.

نویسنده: کامران
تاریخ: ۱۳۹۲/۱۲/۲۶ ۱۶:۰

سلام.
من اگر بخوام لیست کاربرایی که یک نقش خاص مثلا "Admin" رو دارن لیست کنم چطور میتونم دسترسی داشته باشم؟

نویسنده: غلامرضا
تاریخ: ۱۳۹۳/۰۵/۳۰ ۳:۴۹

سلام و میبخشید آیا به غیر از روش ایداعی این سایت روش دیگه ای برای استفاده از asp.net identity با EF db first وجود داره؟ اگه هست لطفا راهنمایی کنید.

نویسنده: غلامرضا
تاریخ: ۱۳۹۳/۰۵/۳۱ ۱۶:۳۲

جواب رو پیدا کردم اینم لینکش. فیلم

عنوان: **معرفی کتابخانه Postal برای ASP.NET MVC**
نویسنده: آرمین ضیاء
تاریخ: ۱۳۹۲/۱۰/۲۰ ۰:۱۵
آدرس: www.dotnettips.info
گروه‌ها: Tips, Razor, ASP.Net MVC, MVC, Email, SMTP

Postal کتابخانه ای برای تولید و ارسال ایمیل توسط نماهای ASP.NET MVC است. برای شروع این کتابخانه را به پروژه خود اضافه کنید. پنجره **Package Manager Console** را باز کرده و فرمان زیر را اجرا کنید.

```
PM> Install-Package Postal
```

شروع به کار با Postal

نحوه استفاده از Postal در کنترلرهای خود را در کد زیر مشاهده می‌کنید.

```
using Postal;

public class HomeController : Controller
{
  public ActionResult Index()
  {
      dynamic email = new Email("Example");
      email.To = "webninja@example.com";
      email.FunnyLink = DB.GetRandomLolcatLink();
      email.Send();
      return View();
  }
}
```

Postal نمای ایمیل را در مسیر Views\Emails\Example.cshtml جستجو می‌کند.

```
To: @ViewBag.To
From: lolcats@website.com
Subject: Important Message

Hello,
You wanted important web links right?
Check out this: @ViewBag.FunnyLink

<3
```

پیکربندی SMTP

Postal ایمیل‌ها را توسط **SmtpClient** ارسال می‌کند که در فریم ورک دات نت موجود است. تنظیمات SMTP را می‌توانید در فایل web.config خود پیکربندی کنید. برای اطلاعات بیشتر به MSDN Documentation مراجعه کنید.

```
<configuration>
  ...
  <system.net>
    <mailSettings>
      <smtp deliveryMethod="network">
        <network host="example.org" port="25" defaultCredentials="true"/>
      </smtp>
    </mailSettings>
  </system.net>
  ...
</configuration>
```

ایمیل‌های Strongly-typed

همه خوششان نمی‌آید از آبجکت‌های دینامیک استفاده کنند. علاوه بر آن آبجکت‌های دینامیک مشکلاتی هم دارند. مثلا قابلیت

IntelliSense و یا Compile-time error را نخواهید داشت.

قدم اول - کلاسی تعریف کنید که از Email ارث بری می‌کند.

```
namespace App.Models
{
  public class ExampleEmail : Email
  {
    public string To { get; set; }
    public string Message { get; set; }
  }
}
```

قدم دوم - از این کلاس استفاده کنید!

```
public void Send()
{
  var email = new ExampleEmail
  {
    To = "hello@world.com",
    Message = "Strong typed message"
  };
  email.Send();
}
```

قدم سوم - نمایی ایجاد کنید که از مدل شما استفاده می‌کند. نام نما، بر اساس نام کلاس مدل انتخاب شده است. بنابراین مثلا *ExampleEmail* نمایی با نام *Example.cshtml* لازم دارد.

```
@model App.Models.ExampleEmail
To: @Model.To
From: postal@example.com
Subject: Example

Hello,
@Model.Message
Thanks!
```

آزمون‌های واحد (Unit Testing)

هنگام تست کردن کدهایی که با Postal کار می‌کنند، یکی از کارهایی که می‌خواهید انجام دهید حصول اطمینان از ارسال شدن ایمیل‌ها است. البته در بدنه تست‌ها نمی‌خواهیم هیچ ایمیلی ارسال شود.

Postal یک قرارداد بنام **IEmailService** و یک پیاده سازی پیش فرض از آن بنام **EmailService** ارائه می‌کند، که در واقع ایمیل‌ها را ارسال هم می‌کند. با در نظر گرفتن این پیش فرض که شما از یک IoC Container استفاده می‌کنید (مانند StructureMap, Ninject)، آن را طوری پیکربندی کنید تا یک نمونه از IEmailService به کنترلرها تزریق کند. سپس از این سرویس برای ارسال آبجکت‌های ایمیل‌ها استفاده کنید (بجای فراخوانی متد Email.Send()).

```
public class ExampleController : Controller
{
    public ExampleController(IEmailService emailService)
    {
        this.emailService = emailService;
    }

    readonly IEmailService emailService;

    public ActionResult Index()
    {
        dynamic email = new Email("Example");
        // ...
        emailService.Send(email);
        return View();
    }
}
```

این کنترلر را با ساختن یک Mock از اینترفیس IEmailService تست کنید. یک مثال با استفاده از FakeItEasy را در زیر مشاهده می‌کنید.

```
[Test]
public void ItSendsEmail()
{
    var emailService = A.Fake<IEmailService>();
    var controller = new ExampleController(emailService);
    controller.Index();
    A.CallTo(() => emailService.Send(A<Email>._))
     .MustHaveHappened();
}
```

ایمیل‌های ساده و HTML

Postal ارسال ایمیل‌های ساده (plain text) و HTML را بسیار ساده می‌کند.

قدم اول - نمای اصلی را بسازید. این نما headerها را خواهد داشت و نماهای مورد نیاز را هم رفرنس می‌کند. مسیر نما ~\Views\Emails\Example.cshtml است.

```
To: test@test.com
From: example@test.com
Subject: Fancy email
Views: Text, Html
```

قدوم دوم - نمای تکست را ایجاد کنید. به قوانین نامگذاری دقت کنید، Example.cshtml به Example.Text.cshtml تغییر یافته. مسیر فایل Views\Emails\Example.Text.cshtml است.

```
Content-Type: text/plain; charset=utf-8

Hello @ViewBag.PersonName,
This is a message
```

دقت داشته باشید که تنها یک Content-Type باید تعریف کنید.

قدم سوم - نمای HTML را ایجاد کنید (باز هم فقط با یک Content-Type). مسیر فایل ~\Views\Emails\Example.Html.cshtml است.

```
Content-Type: text/html; charset=utf-8

<html>
  <body>
    <p>Hello @ViewBag.PersonName,</p>
    <p>This is a message</p>
  </body>
</html>
```

ضمیمه ها

برای افزودن ضمائم خود به ایمیل ها، متد Attach را فراخوانی کنید.

```
dynamic email = new Email("Example");
email.Attach(new Attachment("c:\\attachment.txt"));
email.Send();
```

جاسازی تصاویر در ایمیل ها

Postal یک HTML Helper دارد که امکان جاسازی (embedding) تصاویر در ایمیل‌ها را فراهم می‌کند. دیگر نیازی نیست به یک URL خارجی اشاره کنید.

ابتدا مطمئن شوید که فایل web.config شما فضای نام Postal را اضافه کرده است. این کار دسترسی به HTML Helper مذکور در نمای‌های ایمیل را ممکن می‌سازد.

```
<configuration>
  <system.web.webPages.razor>
    <pages pageBaseType="System.Web.Mvc.WebViewPage">
      <namespaces>
        <add namespace="Postal" />
      </namespaces>
    </pages>
  </system.web.webPages.razor>
</configuration>
```

متد **EmbedImage** تصویر مورد نظر را در ایمیل شما جاسازی می‌کند و توسط یک تگ <img/> آن را رفرنس می‌کند.

```
To: john@example.org
From: app@example.org
Subject: Image

@Html.EmbedImage("~/content/postal.jpg")
```

Postal سعی می‌کند تا نام فایل تصویر را، بر اساس مسیر تقریبی ریشه اپلیکیشن شما تعیین کند.

Postal بیرون از ASP.NET

Postal می‌تواند نماهای ایمیل‌ها را بیرون از فضای ASP.NET رندر کند. مثلا در یک اپلیکیشن کنسول یا یک سرویس ویندوز. این امر توسط یک View Engine سفارشی میسر می‌شود. تنها نماهای Razor پشتیبانی می‌شوند. نمونه کدی را در زیر مشاهده می‌کنید.

```
using Postal;

class Program
{
    static void Main(string[] args)
    {
        // Get the path to the directory containing views
        var viewsPath = Path.GetFullPath(@"..\..\Views");

        var engines = new ViewEngineCollection();
        engines.Add(new FileSystemRazorViewEngine(viewsPath));

        var service = new EmailService(engines);

        dynamic email = new Email("Test");
        // Will look for Test.cshtml or Test.vbhtml in Views directory.
        email.Message = "Hello, non-asp.net world!";
        service.Send(email);
    }
}
```

محدودیت ها: نمی توانید برای نمای ایمیل هایتان از Layoutها استفاده کنید. همچنین در نماهای خود تنها از مدل‌ها (Models) می‌توانید استفاده کنید، و نه ViewBag.

Email Headers: برای در بر داشتن نام، در آدرس ایمیل از فرمت زیر استفاده کنید.

```
To: John Smith <john@example.org>
```

Multiple Values: برخی از headerها می‌توانند چند مقدار داشته باشند. مثلا Bcc و CC. اینگونه مقادیر را می‌توانید به دو روش در نمای خود تعریف کنید:

جدا کردن مقادیر با کاما:

```
Bcc: john@smith.com, harry@green.com
Subject: Example
```

```
etc
```

و یا تکرار header:

```
Bcc: john@smith.com
Bcc: harry@green.com
Subject: Example

etc
```

ساختن ایمیل بدون ارسال آن

لازم نیست برای ارسال ایمیل هایتان به Postal تکیه کنید. در عوض می‌توانید یک آبجکت از نوع System.Net.Mail.MailMessage تولید کنید و به هر نحوی که می‌خواهید آن را پردازش کنید. مثلا شاید بخواهید بجای ارسال ایمیل ها، آنها را به یک صف پیام مثل MSMQ انتقال دهید یا بعدا توسط سرویس دیگری ارسال شوند. این آبجکت MailMessage تمامی Header ها، محتوای اصلی ایمیل و ضمائم را در بر خواهد گرفت.

کلاس EmailService در Postal متدی با نام CreateMailMessage فراهم می‌کند.

```
public class ExampleController : Controller
{
    public ExampleController(IEmailService emailService)
    {
        this.emailService = emailService;
    }

    readonly IEmailService emailService;

    public ActionResult Index()
    {
        dynamic email = new Email("Example");
        // ...

        var message = emailService.CreateMailMessage(email);
        CustomProcessMailMessage(message);

        return View();
    }
}
```

در این پست با امکانات اصلی کتابخانه Postal آشنا شدید و دیدید که به سادگی می‌توانید ایمیل‌های Razor بسازید. برای اطلاعات بیشتر لطفا به سایت پروژه Postal مراجعه کنید.

عنوان: Attribute Routing در ASP.NET MVC 5
نویسنده: آرمین ضیاء
تاریخ: ۱۳۹۲/۱۰/۲۵ ۲:۴۵
آدرس: www.dotnettips.info
گروه‌ها: MVC, ASP.Net MVC, ASP.NET MVC 5

Routing مکانیزم مسیریابی ASP.NET MVC است، که یک URI را به یک اکشن متد نگاشت می‌کند. MVC 5 نوع جدیدی از مسیر یابی را پشتیبانی میکند که Attribute Routing یا مسیریابی نشانه ای نام دارد. همانطور که از نامش پیداست، مسیریابی نشانه ای از Attributeها برای این امر استفاده میکند. این روش به شما کنترل بیشتری روی URIهای اپلیکیشن تان می‌دهد.
مدل قبلی مسیریابی (conventional-routing) هنوز کاملا پشتیبانی می‌شود. در واقع می‌توانید هر دو تکنیک را بعنوان مکمل یکدیگر در یک پروژه استفاده کنید.
در این پست قابلیت‌ها و گزینه‌های اساسی مسیریابی نشانه ای را بررسی میکنیم.
چرا مسیریابی نشانه ای؟
فعال سازی مسیریابی نشانه ای
پارامترهای اختیاری URI و مقادیر پیش فرض
پیشوند مسیر ها
مسیر پیش فرض
محدودیت‌های مسیر ها محدودیت‌های سفارشی

نام مسیر ها
ناحیه‌ها (Areas)

چرا مسیریابی نشانه ای

برای مثال یک وب سایت تجارت آنلاین بهینه شده اجتماعی، می‌تواند مسیرهایی مانند لیست زیر داشته باشد:
{productId:int}/{productTitle}
نگاشت می‌شود به: ProductsController.Show(int id)
{username}
نگاشت می‌شود به: ProfilesController.Show(string username)
{username}/catalogs/{catalogId:int}/{catalogTitle}

نگاشت می‌شود به: CatalogsController.Show(string username, int catalogId)
در نسخه قبلی ASP.NET MVC، قوانین مسیریابی در فایل RouteConfig.cs تعریف می‌شدند، و اشاره به اکشن‌های کنترلرها به نحو زیر انجام می‌شد:

```
routes.MapRoute(
    name: "ProductPage",
    url: "{productId}/{productTitle}",
    defaults: new { controller = "Products", action = "Show" },
    constraints: new { productId = "\\d+" }
);
```

هنگامی که قوانین مسیریابی در کنار اکشن متدها تعریف می‌شوند، یعنی در یک فایل سورس و نه در یک کلاس پیکربندی خارجی، درک و فهم نگاشت URIها به اکشن‌ها واضح‌تر و راحت می‌شود. تعریف مسیر قبلی، می‌تواند توسط یک attribute ساده بدین صورت نگاشت شود:

```
[Route("{productId:int}/{productTitle}")]
public ActionResult Show(int productId) { ... }
```

فعال سازی Attribute Routing

برای فعال سازی مسیریابی نشانه ای، متد MapMvcAttributeRoutes را هنگام پیکربندی فراخوانی کنید.

```
public class RouteConfig
{
    public static void RegisterRoutes(RouteCollection routes)
    {
        routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

        routes.MapMvcAttributeRoutes();
    }
}
```

همچنین می‌توانید مدل قبلی مسیریابی را با تکنیک جدید تلفیق کنید.

```
public static void RegisterRoutes(RouteCollection routes)
{
    routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

    routes.MapMvcAttributeRoutes();

    routes.MapRoute(
        name: "Default",
        url: "{controller}/{action}/{id}",
        defaults: new { controller = "Home", action = "Index", id = UrlParameter.Optional }
    );
}
```

پارامترهای اختیاری URI و مقادیر پیش فرض

می توانید با اضافه کردن یک علامت سوال به پارامترهای مسیریابی، آنها را optional یا اختیاری کنید. برای تعیین مقدار پیش فرض هم از فرمت parameter=value استفاده می‌کنید.

```
public class BooksController : Controller
{
    // eg: /books
    // eg: /books/1430210079
    [Route("books/{isbn?}")]
    public ActionResult View(string isbn)
    {
        if (!String.IsNullOrEmpty(isbn))
        {
            return View("OneBook", GetBook(isbn));
        }
        return View("AllBooks", GetBooks());
    }

    // eg: /books/lang
    // eg: /books/lang/en
    // eg: /books/lang/he
    [Route("books/lang/{lang=en}")]
    public ActionResult ViewByLanguage(string lang)
    {
        return View("OneBook", GetBooksByLanguage(lang));
    }
}
```

در این مثال، هر دو مسیر books/ و books/1430210079/ به اکشن متد "View" نگاشت می‌شوند، مسیر اول تمام کتاب‌ها را لیست میکند، و مسیر دوم جزئیات کتابی مشخص را لیست می‌کند. هر دو مسیر books/lang/ و books/lang/en/ به یک شکل نگاشت می‌شوند، چرا که مقدار پیش فرض این پارامتر en تعریف شده.

پیشوند مسیرها (Route Prefixes)

برخی اوقات، تمام مسیرها در یک کنترلر با یک پیشوند شروع می‌شوند. بعنوان مثال:

```
public class ReviewsController : Controller
{
    // eg: /reviews
```

```
    [Route("reviews")]
    public ActionResult Index() { ... }
    // eg: /reviews/5
    [Route("reviews/{reviewId}")]
    public ActionResult Show(int reviewId) { ... }
    // eg: /reviews/5/edit
    [Route("reviews/{reviewId}/edit")]
    public ActionResult Edit(int reviewId) { ... }
}
```

همچنین می‌توانید با استفاده از خاصیت [RoutePrefix] یک پیشوند عمومی برای کل کنترلر تعریف کنید:

```
[RoutePrefix("reviews")]
public class ReviewsController : Controller
{
    // eg.: /reviews
    [Route]
    public ActionResult Index() { ... }
    // eg.: /reviews/5
    [Route("{reviewId}")]
    public ActionResult Show(int reviewId) { ... }
    // eg.: /reviews/5/edit
    [Route("{reviewId}/edit")]
    public ActionResult Edit(int reviewId) { ... }
}
```

در صورت لزوم، می‌توانید برای بازنویسی (override) پیشوند مسیرها از کاراکتر ~ استفاده کنید:

```
[RoutePrefix("reviews")]
public class ReviewsController : Controller
{
    // eg.: /spotlight-review
    [Route("~/spotlight-review")]
    public ActionResult ShowSpotlight() { ... }

    ...
}
```

مسیر پیش فرض

می توانید خاصیت [Route] را روی کنترلر اعمال کنید، تا اکشن متد را بعنوان یک پارامتر بگیرید. این مسیر سپس روی تمام اکشن متدهای این کنترلر اعمال می‌شود، مگر آنکه یک [Route] بخصوص روی اکشن‌ها تعریف شده باشد.

```
[RoutePrefix("promotions")]
[Route("{action=index}")]
public class ReviewsController : Controller
{
    // eg.: /promotions
    public ActionResult Index() { ... }

    // eg.: /promotions/archive
    public ActionResult Archive() { ... }

    // eg.: /promotions/new
    public ActionResult New() { ... }

    // eg.: /promotions/edit/5
    [Route("edit/{promoId:int}")]
    public ActionResult Edit(int promoId) { ... }
}
```

محدودیت‌های مسیر ها

با استفاده از Route Constraints می‌توانید نحوه جفت شدن پارامترها در قالب مسیریابی را محدود و کنترل کنید. فرمت کلی {parameter:constraint} است. بعنوان مثال:

```
// eg: /users/5
[Route("users/{id:int}"]
public ActionResult GetUserById(int id) { ... }

// eg: users/ken
[Route("users/{name}"]
public ActionResult GetUserByName(string name) { ... }
```

در اینجا، مسیر اول تنها در صورتی انتخاب می‌شود که قسمت id در URI یک مقدار integer باشد. در غیر اینصورت مسیر دوم انتخاب خواهد شد.

جدول زیر constraintها یا محدودیت هایی که پشتیبانی می‌شوند را لیست می‌کند.

مثال	توضیحات	محدودیت
{x:alpha}	کاراکترهای الفبای لاتین را تطبیق (match) می‌دهد (a-z, A-Z).	alpha
{x:bool}	یک مقدار منطقی را تطبیق می‌دهد.	bool
{x:datetime}	یک مقدار **DateTime** را تطبیق می‌دهد.	datetime
{x:decimal}	یک مقدار پولی را تطبیق می‌دهد.	decimal
{x:double}	یک مقدار اعشاری 64 بیتی را تطبیق می‌دهد.	double
{x:float}	یک مقدار اعشاری 32 بیتی را تطبیق می‌دهد.	float
{x:guid}	یک مقدار GUID را تطبیق می‌دهد.	guid
{x:int}	یک مقدار 32 بیتی integer را تطبیق می‌دهد.	int
{x:length(6)} {x:length(1,20)}	رشته ای با طول تعیین شده را تطبیق می‌دهد.	length
{x:long}	یک مقدار 64 بیتی integer را تطبیق می‌دهد.	long
{(x:max(10}	یک مقدار integer با حداکثر مجاز را تطبیق می‌دهد.	max
{(x:maxlength(10}	رشته ای با حداکثر طول تعیین شده را تطبیق می‌دهد.	maxlength
{(x:min(10}	مقداری integer با حداقل مقدار تعیین شده را تطبیق می‌دهد.	min
{(x:minlength(10}	رشته ای با حداقل طول تعیین شده را تطبیق می‌دهد.	minlength
{x:range(10,50)}	مقداری integer در بازه تعریف شده را تطبیق می‌دهد.	range
{($${x:regex(^\d{3}-\d{3}-\d{4}	یک عبارت با قاعده را تطبیق می‌دهد.	regex

توجه کنید که بعضی از constraint ها، مانند "min" آرگومان‌ها را در پرانتز دریافت می‌کنند.

می توانید محدودیت‌های متعددی روی یک پارامتر تعریف کنید، که باید با دونقطه جدا شوند. بعنوان مثال:

```
// eg: /users/5
// but not /users/10000000000 because it is larger than int.MaxValue,
// and not /users/0 because of the min(1) constraint.
[Route("users/{id:int:min(1)}")]
public ActionResult GetUserById(int id) { ... }
```

مشخص کردن اختیاری بودن پارامتر ها، باید در آخر لیست constraints تعریف شود:

```
// eg: /greetings/bye
// and /greetings because of the Optional modifier,
// but not /greetings/see-you-tomorrow because of the maxlength(3) constraint.
[Route("greetings/{message:maxlength(3)?}")]
public ActionResult Greet(string message) { ... }
```

محدودیت‌های سفارشی

با پیاده سازی قرارداد IRouteConstraint می‌توانید محدودیت‌های سفارشی بسازید. بعنوان مثال، constraint زیر یک پارامتر را به لیستی از مقادیر قابل قبول محدود می‌کند:

```
public class ValuesConstraint : IRouteConstraint
{
    private readonly string[] validOptions;
    public ValuesConstraint(string options)
    {
        validOptions = options.Split('|');
    }

    public bool Match(HttpContextBase httpContext, Route route, string parameterName,
RouteValueDictionary values, RouteDirection routeDirection)
    {
        object value;
        if (values.TryGetValue(parameterName, out value) && value != null)
        {
            return validOptions.Contains(value.ToString(), StringComparer.OrdinalIgnoreCase);
        }
        return false;
    }
}
```

قطعه کد زیر نحوه رجیستر کردن این constraint را نشان می‌دهد:

```
public class RouteConfig
{
    public static void RegisterRoutes(RouteCollection routes)
    {
        routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

        var constraintsResolver = new DefaultInlineConstraintResolver();

        constraintsResolver.ConstraintMap.Add("values", typeof(ValuesConstraint));

        routes.MapMvcAttributeRoutes(constraintsResolver);
    }
}
```

حالا می‌توانید این محدودیت سفارشی را روی مسیرها اعمال کنید:

```
public class TemperatureController : Controller
{
    // eg: temp/celsius and /temp/fahrenheit but not /temp/kelvin
    [Route("temp/{scale:values(celsius|fahrenheit)}")]
    public ActionResult Show(string scale)
    {
```

```
        return Content("scale is " + scale);
    }
}
```

نام مسیر ها

می توانید به مسیرها یک نام اختصاص دهید، با این کار تولید URIها هم راحت‌تر می‌شوند. بعنوان مثال برای مسیر زیر:

```
[Route("menu", Name = "mainmenu")]
public ActionResult MainMenu() { ... }
```

می‌توانید لینکی با استفاده از Url.RouteUrl تولید کنید:

```
<a href="@Url.RouteUrl("mainmenu")">Main menu</a>
```

ناحیه‌ها (Areas)

برای مشخص کردن ناحیه ای که کنترلر به آن تعلق دارد می‌توانید از خاصیت [RouteArea] استفاده کنید. هنگام استفاده از این خاصیت، می‌توانید با خیال راحت کلاس AreaRegistration را از ناحیه مورد نظر حذف کنید.

```
[RouteArea("Admin")]
[RoutePrefix("menu")]
[Route("{action}")]
public class MenuController : Controller
{
    // eg: /admin/menu/login
    public ActionResult Login() { ... }

    // eg: /admin/menu/show-options
    [Route("show-options")]
    public ActionResult Options() { ... }

    // eg: /stats
    [Route("~/stats")]
    public ActionResult Stats() { ... }
}
```

با این کنترلر، فراخوانی تولید لینک زیر، رشته " */Admin/menu/show-options* " را بدست میدهد:

```
Url.Action("Options", "Menu", new { Area = "Admin" })
```

به منظور تعریف یک پیشوند سفارشی برای یک ناحیه، که با نام خود ناحیه مورد نظر متفاوت است می‌توانید از پارامتر AreaPrefix استفاده کنید. بعنوان مثال:

```
[RouteArea("BackOffice", AreaPrefix = "back-office")]
```

اگر از ناحیه‌ها هم بصورت مسیریابی نشانه ای، و هم بصورت متداول (که با کلاس‌های AreaRegistration پیکربندی می‌شوند) استفاده می‌کنید باید مطمئن شوید که رجیستر کردن نواحی اپلیکیشن پس از مسیریابی نشانه ای پیکربندی می‌شود. به هر حال رجیستر کردن ناحیه‌ها پیش از تنظیم مسیرها بصورت متداول باید صورت گیرد. دلیل آن هم مشخص است، برای اینکه درخواست‌های ورودی بدرستی با مسیرهای تعریف شده تطبیق داده شوند، باید ابتدا attribute routes، سپس area registration و در آخر default route رجیستر شوند. بعنوان مثال:

```
public static void RegisterRoutes(RouteCollection routes)
{
    routes.IgnoreRoute("{resource}.axd/{*pathInfo}");
```

```
    routes.MapMvcAttributeRoutes();

    AreaRegistration.RegisterAllAreas();

    routes.MapRoute(
        name: "Default",
        url: "{controller}/{action}/{id}",
        defaults: new { controller = "Home", action = "Index", id = UrlParameter.Optional }
    );
}
```

نظرات خوانندگان

نویسنده: منصور جعفری
تاریخ: ۱۳۹۲/۱۲/۲۹ ۰:۲۷

سلام
الان من در قسمت route.config به این صورت کدهام تعریف شده

```
public static void RegisterRoutes(RouteCollection routes)
        {
            routes.IgnoreRoute("{resource}.axd/{*pathInfo}");
            routes.MapMvcAttributeRoutes();
            routes.MapRoute(
                name: "Default",
                url: "{controller}/{action}/{id}",
                defaults: new { controller = "Home", action = "List", id = UrlParameter.Optional }
            );
        }
```

و اومدم توی کنترل Home و اکشن موبوط به اون که دارای یک پارامتر دریافتی برای پیج هست این کد رو تعریف کردم:

```
[Route("Page/{page?}")]
        public ActionResult List(int page=1)
        {
```

اما وقتی برنامه رو اجرا میکنم خطای 404 رو میده. ممنون میشم راهنمایی کنید مشکل از کجاست.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۲/۲۹ ۱:۱۱

برای ریشه سایت ویژگی زیر را هم باید اضافه کنید:

```
[Route("~/")]
```

نویسنده: منصور جعفری
تاریخ: ۱۳۹۲/۱۲/۲۹ ۱۲:۳۳

ممنونم که پاسخ دادید یعنی باید به این شکل بنویسم

```
[Route("~/Page/{page?}")]
        public ActionResult List(int page=1)
        {
```

نیازی به تعریف اکشن یا RoutePrifix نیست؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۲/۲۹ ۱۲:۴۵

قابلیت ترکیب دارند:

```
[Route("~/")]
[Route("Page/{page?}")]
public ActionResult List(int page=1)
{
```

نویسنده: یاشار راشدی
تاریخ: ۱۳۹۳/۰۳/۲۹ ۱۳:۲۲

در صورتی که از Areaها استفاده کنید باید بالای هر کنترلر داخل Area حتما Prefix مربوط به Area را اضافه کنید وگرنه Exception دریافت می‌کنید.

```
[RouteArea("Admin")]
[RoutePrefix("menu")]
[Route("{action}")]
public class MenuController : Controller
{
    // eg: /admin/menu/login
    public ActionResult Login() { ... }

    // eg: /admin/menu/show-options
    [Route("show-options")]
    public ActionResult Options() { ... }

    // eg: /stats
    [Route("~/stats")]
    public ActionResult Stats() { ... }
```

توضیحات بیشتر

http://blogs.msdn.com/b/webdev/archive/2013/10/17/attribute-routing-in-asp-net-mvc-5.aspx#route-areas

عنوان: **ایجاد سیستم وضعیت آب و هوا مانند گوگل (بخش اول)**
نویسنده: شقایق اشتری
تاریخ: ۱۳۹۲/۱۲/۱۰ ۱۱:۳۰
آدرس: www.dotnettips.info
گروه‌ها: ASP.Net, C#, ASP.Net MVC, MVC, xml

در این آموزش قصد دارم چگونگی ایجاد یک سیستم اعلام وضعیت آب و هوا را مشابه آنچه که در سایت گوگل می‌بینید برای شما توضیح دهم. باید توجه داشت من این آموزش را با ASP.NET MVC نوشتم ولی شما می‌توانید با اندک تغییراتی در کدها، آنرا در ASP.NET وب فرمز نیز استفاده کنید. برای گرفتن آب و هوای هر شهر از Rssهای اعلام وضعیت آب و هوای یاهو استفاده می‌کنم و توضیح خواهم داد که چگونه با Rss آن کار کنید.
Rss آب و هوای هر شهر در یاهو به صورت یک لینک یکتا می‌باشد؛ به شکل زیر :

```
http://weather.yahooapis.com/forecastrss?w=WOEID&u=c
```

حال می‌خواهم کوئری استرینگ‌های این لینک را برای شما توضیح دهم. هر شهری بر روی کره‌ی زمین یک WOEID یکتا و منحصر بفرد دارد که شما به پارامتر w عدد WOEID شهر موردنظر خود را می‌دهید. بعد از مقداردهی پارامتر w، وقتی این لینک را در آدرس بار مرورگر خود می‌زنید، RSS مربوط به آب و هوای آن شهر را به شما می‌دهد. مثلا WOEID تهران عدد 28350859 می‌باشد.
و این لینک http://weather.yahooapis.com/forecastrss?w=28350859&u=c اطلاعات آب و هوای تهران را در قالب یک RSS به شما نمایش خواهد داد.

خوب، حالا پارامتر دوم یعنی پارامتر u چکاری را انجام می‌دهد؟
* چنانچه مقدار پارامتر u برابر c باشد، یعنی شما دمای آب و هوای شهر مد نظر را بر اساس سانتیگراد می‌خواهید.
* اگر مقدار پارامتر u برابر f باشد، یعنی شما دمای آب و هوای آن شهر مورد نظر را بر اساس فارنهایت می‌خواهید.

برای گرفتن WOEID شهرها هم به این سایت بروید http://woeid.rosselliot.co.nz و اسم هر شهری که می‌خواهید بزنید تا WOEID را به شما نمایش دهد.

در این مثال من از یک DropDown استفاده کردم که کاربر با انتخاب هر شهر از DropDown، آب و هوای آن شهر را مشاهده می‌کند.
Action مربوط به صفحه‌ی Index به صورت زیر می‌باشد :

```
[HttpGet]
        public ActionResult Index()
        {
           ViewBag.ProvinceList = _RPosition.Positions;
            ShowWeatherProvince(8);
            return View();
        }
```

در اینجا من لیست شهرها را از جدول می‌خوانم که البته این جدول را چون بخش مهمی نبود و فقط شامل ID و نام شهرها بود در فایل ضمیمه قرار ندادم و نام شهرها و ID آنها را بر عهده‌ی خودتان گذاشتم.
حال تابعی را که آب و هوای مربوط به هر شهر را نمایش می‌دهد، به شرح زیر است:

```
public ActionResult ShowWeatherProvince(int dpProvince)
        {
            XDocument rssXml=null;
            CountryName CountryName = new CountryName();
            if (dpProvince != 0)
            {
                switch (dpProvince)
                {
                    case 1:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345768&u=c");
```

```
                            CountryName = new CountryName() { Country = "Iran", City = "Azarbayejan-e
Sharqhi" };
                            break;
                        }
                    case 2:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345767&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Azarbayejan-e
Qarbi" };
                            break;
                        }
                    case 3:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2254335&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Ardebil" };
                            break;
                        }
                    case 4:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=28350859&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Alborz" };
                            break;
                        }
                    case 5:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345787&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Esfahan" };
                            break;
                        }
                    case 6:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345775&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Ilam" };
                            break;
                        }
                    case 7:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2254463&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Bushehr" };
                            break;
                        }
                    case 8:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=28350859&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Tehran" };
                            break;
                        }
                    case 9:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345769&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Chahar Mahall
va Bakhtiari" };
                            break;
                        }
                    case 10:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=56189824&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Razavi
Khorasan" };
                            break;
                        }
                    case 11:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345789&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Shomali
Khorasan" };
                            break;
                        }
                    case 12:
                        {
                            rssXml =
```

```
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345789&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Jonubi
Khorasan" };
                            break;
                        }
                    case 13:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345778&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Khuzestan" };
                            break;
                        }
                    case 14:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2255311&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Zanjan" };
                            break;
                        }
                    case 15:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345784&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Semnan" };
                            break;
                        }
                    case 16:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345770&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Sistan va
Baluchestan" };
                            break;
                        }
                    case 17:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345772&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Fars" };
                            break;
                        }
                    case 18:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=20070200&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Qazvin" };
                            break;
                        }
                    case 19:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2255062&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Qom" };
                            break;
                        }
                    case 20:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345779&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Kordestan" };
                            break;
                        }
                    case 21:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2254796&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Kerman" };
                            break;
                        }
                    case 22:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2254797&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Kermanshah" };
                            break;
                        }
                    case 23:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345771&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Kohgiluyeh va
```

```
Buyer Ahmad" };
                            break;
                        }
                    case 24:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=20070201&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Golestan" };
                            break;
                        }
                    case 25:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345773&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Gilan" };
                            break;
                        }
                    case 26:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345782&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Lorestan" };
                            break;
                        }
                    case 27:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345783&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Markazi" };
                            break;
                        }
                    case 28:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345780&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Mazandaran" };
                            break;
                        }
                    case 29:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2254664&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Hamedan" };
                            break;
                        }
                    case 30:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2345776&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Hormozgan" };
                            break;
                        }
                    case 31:
                        {
                            rssXml =
XDocument.Load("http://weather.yahooapis.com/forecastrss?w=2253355&u=c");
                            CountryName = new CountryName() { Country = "Iran", City = "Yazd" };
                            break;
                        }
                }
                ViewBag.Location = CountryName;
                XNamespace yWeatherNS = "http://xml.weather.yahoo.com/ns/rss/1.0";
                List<YahooWeatherRssItem> WeatherList = new List<YahooWeatherRssItem>();
                for (int i = 0; i < 4; i++)
                {
                    YahooWeatherRssItem YahooWeatherRssItem = new YahooWeatherRssItem()
                    {
                        Code = Convert.ToInt32(rssXml.Descendants("item").Elements(yWeatherNS +
"forecast").ElementAt(i).Attribute("code").Value),
                        Day = rssXml.Descendants("item").Elements(yWeatherNS +
"forecast").ElementAt(i).Attribute("day").Value,
                        Low = rssXml.Descendants("item").Elements(yWeatherNS +
"forecast").ElementAt(i).Attribute("low").Value,
                        High = rssXml.Descendants("item").Elements(yWeatherNS +
"forecast").ElementAt(i).Attribute("high").Value,
                        Text = rssXml.Descendants("item").Elements(yWeatherNS +
"forecast").ElementAt(i).Attribute("text").Value,
                    };

                    WeatherList.Add(YahooWeatherRssItem);
                }
```

```
                ViewBag.FeedList = WeatherList;
            }

                return PartialView("_Weather");
        }
```

قسمت SwitchCase، مقدار و Value مربوط به هر آیتم DropDown را که شامل یک اسم شهر است، میگیرد و RSS مربوط به آن شهر را بر می‌گرداند.
حالا کد مربوط به خواندن فایل Rss را برایتان توضیح می‌دهم : حلقه‌ی for 0 تا 4 (که در کد بالا مشاهده می‌کنید)یعنی اطلاعات 4 روز آینده را برایم برگردان.
من تگ‌های High ، Low ، Day ، Code و text فایل RSS را در این حلقه For می‌خوانم که البته مقادیر این 4 روز را در لیستی اضافه می‌کنم که نوع این لیست هم از نوع YahooWeatherRssItem می‌باشد. من این کلاس را در فایل ضمیمه قرار دادم. اکنون هر کدام از این تگ‌ها را برایتان توضیح می‌دهم:

code : هر آب و هوا کدی دارد .مثلا آب و هوای نیمه ابری یک کد ، آب و هوای آفتابی کدی دیگر و ...
Low: حداقل دمای آن روز را به ما می‌دهد .
High: حداکثر دمای آن روز را به می‌دهد .
day: نام روز از هفته را بر می‌گرداند مثلا شنبه ، یکشنبه و ....
text: که توضیحاتی می‌دهد مثلا اگر هوا آفتابی باشد مقدار sunny را بر می‌گرداند و ...

خوب، تا اینجا ما Rss مربوط به هر شهر را خواندیم حالا در قسمت Design باید چکار کنیم .
کدهای html صفحه‌ی Index ما شامل کدهای زیر است :

```
@{

    ViewBag.Title = "Weather";
}

<link href="~/Content/User/Weather/Weather.css" rel="stylesheet" />
@section scripts{
    <script src="@Url.Content("~/Scripts/jquery-1.6.2.min.js")" type="text/javascript"></script>
        <script src="@Url.Content("~/Scripts/jquery.unobtrusive-ajax.min.js")"
type="text/javascript"></script>
       <script type="text/javascript">
           $("#dpProvince").change(function () {
               $(this).parents("form").submit();
           });
    </script>
}
<h2>Weather</h2>
<div id="Progress">
    <img src="~/Images/User/Other/ajax-loader.gif" />
</div>
<div id="BoxContent"> @Html.Partial("_Weather")</div>

  @using (Ajax.BeginForm(actionName: "ShowWeatherProvince", ajaxOptions: new AjaxOptions {
UpdateTargetId = "BoxContent", LoadingElementId = "Progress", InsertionMode = InsertionMode.Replace }))
                {
<div style="padding-top:15px;">
        <div style="float:left; width:133px; ">Select Your Province</div>
        <div style="float:left">   @Html.DropDownList("dpProvince", new
SelectList(ViewBag.ProvinceList, "Id", "Name"),"Select Your Province", new { @class =
"webUserDropDown", @style = "width:172px" })</div>
    </div>
  }
```

و کدهای _Weather که Partial است به صورت زیر است:

```
@{
    List<Weather.YahooWeatherRssItem> Feeds = ViewBag.FeedList;
}
<div>
```

```
    @{
        HtmlString StartTable = new HtmlString("<table class='WeatherTable' cellspacing='0'
cellpadding='0'><tbody><tr>");
        HtmlString EndTable = new HtmlString("</tr></tbody></table>");
        HtmlString StartTD = new HtmlString("<td>");
        HtmlString EndTD = new HtmlString("</td>");
    }
    <div style="width: 300px;">
        @{
            @StartTable
            foreach (var item in Feeds)
            {
            @StartTD
            <div>@item.Day</div>
            <div>
                @{
                string FileName = "";
                switch (item.Code)
                {
                    case 0: { FileName = "/Images/User/Weather/Tornado.png"; break; }
                    case 1: { FileName = "/Images/User/Weather/storm2.gif"; break; }
                    case 2: { FileName = "/Images/User/Weather/storm2.gif"; break; }
                    case 3: { FileName = "/Images/User/Weather/storm2.gif"; break; }
                    case 4: { FileName = "/Images/User/Weather/15.gif"; break; }
                    case 5: { FileName = "/Images/User/Weather/29.gif"; break; }
                    case 6: { FileName = "/Images/User/Weather/29.gif"; break; }
                    case 7: { FileName = "/Images/User/Weather/29.gif"; break; }
                    case 8: { FileName = "/Images/User/Weather/26.gif"; break; }
                    case 9: { FileName = "/Images/User/Weather/drizzle.png"; break; }
                    case 10: { FileName = "/Images/User/Weather/26.gif"; break; }
                    case 11: { FileName = "/Images/User/Weather/18.gif"; break; }
                    case 12: { FileName = "/Images/User/Weather/18.gif"; break; }
                    case 13: { FileName = "/Images/User/Weather/19.gif"; break; }
                    case 14: { FileName = "/Images/User/Weather/19.gif"; break; }
                    case 15: { FileName = "/Images/User/Weather/19.gif"; break; }
                    case 16: { FileName = "/Images/User/Weather/22.gif"; break; }
                    case 17: { FileName = "/Images/User/Weather/Hail.png"; break; }
                    case 18: { FileName = "/Images/User/Weather/25.gif"; break; }
                    case 19: { FileName = "/Images/User/Weather/dust.png"; break; }
                    case 20: { FileName = "/Images/User/Weather/fog_icon.png"; break; }
                    case 21: { FileName = "/Images/User/Weather/hazy_icon.png"; break; }
                    case 22: { FileName = "/Images/User/Weather/2017737395.png"; break; }
                    case 23: { FileName = "/Images/User/Weather/32.gif"; break; }
                    case 24: { FileName = "/Images/User/Weather/32.gif"; break; }
                    case 25: { FileName = "/Images/User/Weather/31.gif"; break; }
                    case 26: { FileName = "/Images/User/Weather/7.gif"; break; }
                    case 27: { FileName = "/Images/User/Weather/38.gif"; break; }
                    case 28: { FileName = "/Images/User/Weather/6.gif"; break; }
                    case 29: { FileName = "/Images/User/Weather/35.gif"; break; }
                    case 30: { FileName = "/Images/User/Weather/7.gif"; break; }
                    case 31: { FileName = "/Images/User/Weather/33.gif"; break; }
                    case 32: { FileName = "/Images/User/Weather/1.gif"; break; }
                    case 33: { FileName = "/Images/User/Weather/34.gif"; break; }
                    case 34: { FileName = "/Images/User/Weather/2.gif"; break; }
                    case 35: { FileName = "/Images/User/Weather/freezing_rain.png"; break; }
                    case 36: { FileName = "/Images/User/Weather/30.gif"; break; }
                    case 37: { FileName = "/Images/User/Weather/15.gif"; break; }
                    case 38: { FileName = "/Images/User/Weather/15.gif"; break; }
                    case 39: { FileName = "/Images/User/Weather/15.gif"; break; }
                    case 40: { FileName = "/Images/User/Weather/12.gif"; break; }
                    case 41: { FileName = "/Images/User/Weather/22.gif"; break; }
                    case 42: { FileName = "/Images/User/Weather/22.gif"; break; }
                    case 43: { FileName = "/Images/User/Weather/22.gif"; break; }
                    case 44: { FileName = "/Images/User/Weather/39.gif"; break; }
                    case 45: { FileName = "/Images/User/Weather/thundershowers.png"; break; }
                    case 46: { FileName = "/Images/User/Weather/19.gif"; break; }
                    case 47: { FileName = "/Images/User/Weather/thundershowers.png"; break; }
                    case 3200: { FileName = "/Images/User/Weather/1211810662.png"; break; }
                }
                }
                <img alt='@item.Text' title='@item.Text' src='@FileName'>
            </div>
            <div>
                <span>@item.High°</span>
                <span>@item.Low°</span>
            </div>
            @EndTD
            }
        }
        @EndTable
    </div>
```

```
</div>
```

من عکس‌های مربوط به وضعیت آب و هوا را در فایل ضمیمه قرار دادم.
چنانچه در مورد RSS وضعیت آب و هوای یاهو اطلاعات دقیق‌تری را می‌خواهید بدانید به این لینک بروید.
در آموزش بعدی قصد دارم برایتان این بخش را توضیح دهم که بر اساس IP بازدید کننده سایت شما، اطلاعات آب و هوایی شهر بازدید کننده را برایش در سایت نمایش دهد.

Files-06bf65bac63d4dd694b15fc24d4cb074.zip

موفق باشید

نظرات خوانندگان

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۲/۱۱ ۰:۵

ممنون از شما. چند نکته جزئی در مورد بهبود کیفیت کدها. در Viewهای MVC باید کدنویسی صرفا به نمایش اطلاعات View خلاصه شود. یعنی switch داخل آن بهتر است تبدیل به یک extension method شده و نهایتا استفاده شود (برای تمیز کردن View).
همچنین استفاده از switch 1 و 2 و 10000 اصطلاحا به magic code مشهور هستند. یعنی مشخص نیست معنای این اعداد چی هست. این‌ها را عموما بهتر است تبدیل به enum کرد و بعد استفاده نهایی.

نویسنده: شقایق اشتری
تاریخ: ۱۳۹۲/۱۲/۱۲ ۸:۳۹

ممنونم از شما که مشکل کد نویسی من را گفتید . extension methodها منظورتون همون helperها هست ؟ لینکی دارید که در مورد این extension methodها آموزشی داده باشد چون من می‌خوام کدهامو تا جایی که می‌شود بهینه بنویسم .

نویسنده: شاهین کیاست
تاریخ: ۱۳۹۲/۱۲/۱۲ ۹:۱۵

یک مثال در اینجا
قسمت آخر (**Ifهای شرطی را در Viewها را در متدهای کمکی کپسوله کنید)**

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۲/۱۲ ۹:۵۱

قسمت switch ایی را که در View نوشتید، تبدیل کنید به یک متد کمکی در کلاسی خارج از View: (مهم نیست متد الحاقی باشد یا خیر؛ فقط داخل View نباشد)

```
public static string GetFileName(int code)
{
    switch (code)
    {
       case 0: return "/Images/User/Weather/Tornado.png";
  //...
}
```

بعد یک خاصیت محاسباتی به نام FileName به مدل مورد استفاده اضافه کنید:

```
public class YahooWeatherRssItem
    {
        public int Code { get; set; }
        //...
        public string FileName
        {
          get { return Util.GetFileName(Code); }
        }
    }
```

به این صورت View از کدهای محاسبات یافتن FileName خالی می‌شود.

نویسنده: علی
تاریخ: ۱۳۹۲/۱۲/۱۲ ۱۰:۷

یه راه دیگه هم استفاده از Yahoo Query Language هست، می تونید با کوئری زیر اطلاعات آب و هوای شهر مورد نظر رو در قالب JSON دریافت کنید

```
select * from weather.forecast where woeid in (select woeid from geo.placefinder where text="CityName")
```

نویسنده: شقایق اشتری
تاریخ: ۱۳۹۲/۱۲/۱۲ ۱۰:۱۳

یه سوال دیگه که داشتم اگر بین کدهایمان ، کد html وجود داشت باز هم باید آنها را از view جدا کنیم و به صورت extension method بنویسیم؟

یا اینکه جداسازی کدها از View را زمانی انجام می‌دهیم که چندین خط کد پشت سر هم نوشته باشیم و بین آنها کد html یی نباشد ؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۲/۱۲ ۱۰:۴۱

- نه الزاما. اگر چندین جا استفاده و تکرار می‌شود یا منطق طولانی دارد، روش « نوشتن HTML Helpers ویژه، به کمک امکانات Razor » می‌تواند مفید باشد.
- منطق قرار گرفته در View فقط باید کار «نهایی» نمایشی را انجام دهد. اگر در View مثلا مستقیما با دیتابیس کار می‌کنید، محاسبات مرتبط با فیلد خاصی را انجام می‌دهید و کلا هر منطقی که «نهایی» بودن نمایش اطلاعات را زیر سؤال ببرد، باید از View جدا شده و به کنترلر و زیرساخت آن منتقل شود.

نویسنده: محسن تقی پور
تاریخ: ۱۳۹۲/۱۲/۲۰ ۰:۸

میشه راجبه این روش کمی بیشتر توضیح بدین

نویسنده: Anaritius
تاریخ: ۱۳۹۲/۱۲/۲۱ ۰:۴۴

سلام
ممنون از مقاله خوبتون
امکانش هست سورس کد webForm ش رو هم بذارید واسه دانلود؟
ممنون

نویسنده: علی
تاریخ: ۱۳۹۲/۱۲/۲۱ ۱۰:۸

اینجا یک مثال از نحوه کار با YQL رو توضیح داده