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: **نحوه‌ی صحیح کار کردن با بوت استرپ**
نویسنده: سیروان عفیفی
تاریخ: ۱۳۹۳/۰۴/۰۲ ۲۳:۵
آدرس: www.dotnettips.info
گروه‌ها: CSS Framework, Twitter Bootstrap, LESS, Dynamic Css, bootstrap

DOM در حالت عادی بسیار نامرتب است. همچنین با افزودن کلاس‌های CSS، کد HTML به مراتب نامرتب‌تر از قبل می‌شود. بوت استرپ نیز شامل تعداد زیادی از کلاس‌های CSS می‌باشد که برای انجام وظایف خاصی به HTML اضافه می‌شوند.

روش متداول استفاده از بوت‌استرپ

Embedd کردن کلاس‌های CSS بوت‌استرپ به صورت مستقیم درون HTML

اغلب فریم‌ورک‌ها، از لحاظ معنایی یا semantic، دارای مشکل هستند. اگر به سورس HTML صفحاتی که با این نوع از فریم‌ورک‌ها ساخته شده باشند نگاهی بیندازید با حجم زیادی از کلاس‌هایی مانند و یا مواجه خواهید شد. نوشتن کدهای HTML به این صورت از لحاظ معنایی اشتباه است. مثلاً اگر بنا به دلایلی سازندگان بوت استرپ تصمیم بگیرند نام کلاس‌های را در نسخه بعدی این فریم‌ورک تغییر دهند (مانند تغییر نام کلاس‌ها در نسخه‌ی 3 بوت استرپ)، و یا اگر در آینده بخواهید از یک فریم‌ورک دیگر در سایت‌تان استفاده کنید. باید این تغییرات را در تمام صفحات سایت‌تان اعمال کنید؛ در نتیجه اینکار زمان زیادی را از شما صرف میکند.

راه‌حل؟

استفاده از CSS preprocessors

بوت‌استرپ، از Less برای اینکار استفاده می‌کند. Less در واقع یک CSS preprocessor نوشته شده با جاوا اسکریپت است که قابلیت اجرا در مرورگر را دارد. Less امکانات زیادی، از قبیل استفاده از توابع، متغیرها، Mixins و ... را در اختیار شما قرار می‌دهد. در واقع هدف از Less، نگهداری آسان و قابلیت توسعه فایل‌های CSS می‌باشد. در این حالت شما کدهای CSS خود را درون فایل‌هایی با پسوند Less می‌نویسید. در این حالت بجای پیوست کردن کلاس‌های بوت‌استرپ در کد HTML، آن را درون استایل‌شیت پیوست خواهید کرد. همانطور که عنوان شد، بوت‌استرپ با Less نوشته شده است. فایل‌های Less بوت‌استرپ را می‌توانید از مخزن کد گیت‌هاب آن دانلود نمائید یا اینکه از طریق نیوگت می‌توانید آن را نصب کنید:

```
PM> Install-Package Twitter.Bootstrap.Less
```

کار با Less خیلی ساده است. به عنوان مثال در کد زیر یک کلاس با نام loud داریم که استایل‌هایی را به آن اعمال کرده‌ایم. اکنون جهت استفاده مجدد از این استایل‌ها برای کلاسی دیگر، نیاز به نوشتن مجدد آن نیست. کافی همانند یک تابع در هر کلاسی آن را فراخوانی کنیم:

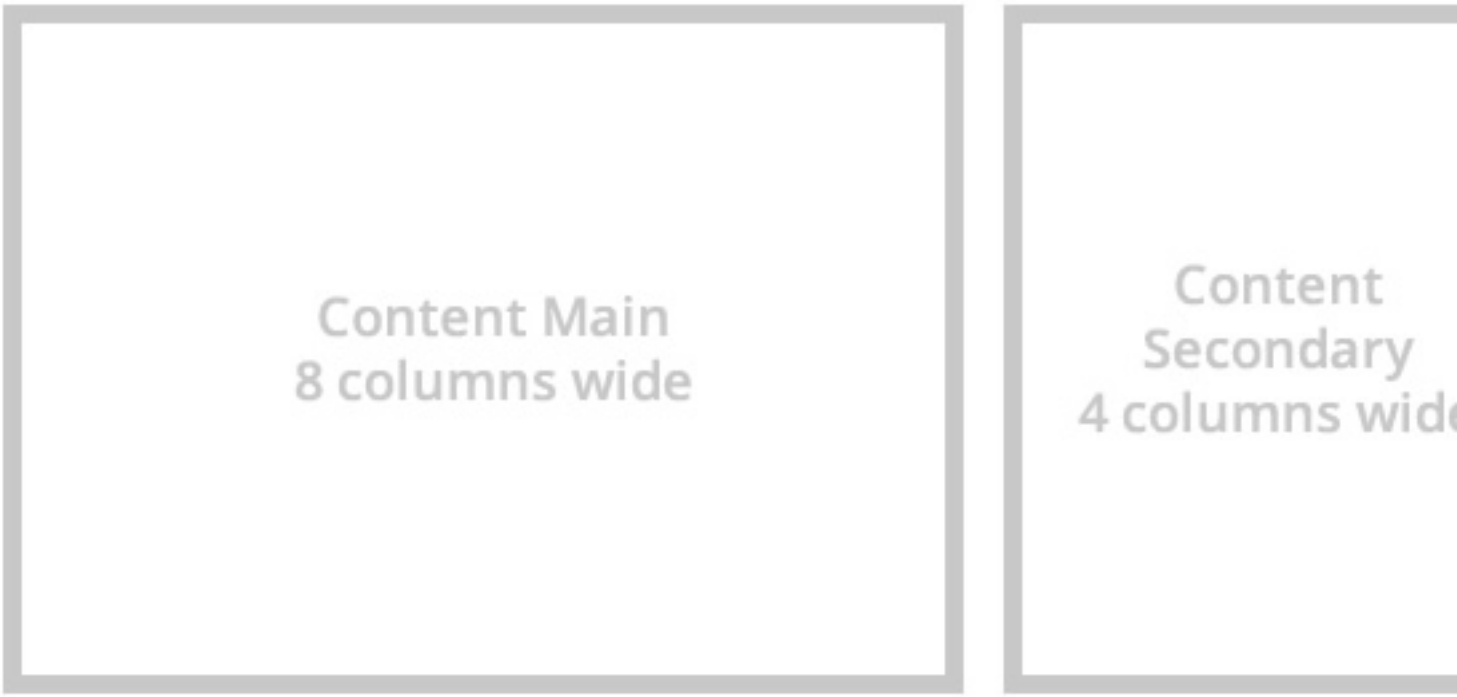
```
.loud {
  color: red;
}

// Make all H1 elements loud
h1 {
  .loud;
}
```

نکته: در Visual Studio 2012 Update 2 به بعد به صورت توکار از فایل‌های Less پشتیبانی می‌شود. (توسط پلاگین Web Essentials)

استفاده از Mixins

با استفاده از Mixins می‌توانیم عناصر داخل صفحات‌مان را به صورت Semantic تعریف نمائیم. به عنوان مثال می‌خواهیم با استفاده از سیستم گرید بوت‌استرپ، ساختاری مانند تصویر زیر را داشته باشیم:



در حالت معمول با استفاده از کلاس‌های CSS بوت‌استرپ می‌توانیم اینکار را انجام دهیم:

```
<div class="container">
    <div class="row">
        <div class="col-md-8">
            Content - Main
        </div>
        <div class="col-md-4">
            Content - Secondary
        </div>
    </div>
</div>
```

کد فوق را بهتر است به این صورت بنویسیم:

```
<div class="wrapper">
    <div class="content-main">
         Content - Main
    </div>
    <div class="content-secondary">
        Content - Secondary
    </div>
</div>
```

در بوت استرپ از Less Mixins جهت اعمال استایل هایی مانند row و column می‌توانیم استفاده کنیم. به طور مثال برای اعمال استایل به کلاس‌های فوق می‌توانیم به این صورت عمل کنیم:

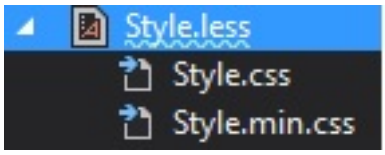
```
// Core variables and mixins
@import "variables.less";
@import "mixins.less";
.wrapper {
  .make-row();
}
.content-main {
```

```
  .make-lg-column(8);
}
.content-secondary {
  .make-lg-column(3);
  .make-lg-column-offset(1);
}
```

کد فوق بعد از کامپایل به کد زیر تبدیل خواهد شد:

```
.wrapper {
  margin-left: -15px;
  margin-right: -15px;
}
.content-main {
  position: relative;
  min-height: 1px;
  padding-left: 15px;
  padding-right: 15px;
}
@media (min-width: 1200px) {
  .content-main {
    float: left;
    width: 66.66666666666666%;
  }
}
.content-secondary {
  position: relative;
  min-height: 1px;
  padding-left: 15px;
  padding-right: 15px;
}
@media (min-width: 1200px) {
  .content-secondary {
    float: left;
    width: 25%;
  }
}
@media (min-width: 1200px) {
  .content-secondary {
    margin-left: 8.333333333333332%;
  }
}
```

همانطور که قبلاً عنوان شد ویژوال استودیو به راحتی توسط افزونه Web Essentials از فایل‌های Less پشتیبانی می‌کند. در نتیجه کامپایل فایل‌های Less داخل ویژوال استودیو توسط این افزونه به راحتی قابل انجام می‌باشد. یک قابلیت جالب دیگر در رابطه با فایل‌هایی Less، تولید نسخه‌های CSS عادی و فشرده نهایی توسط افزونه Web Essentials می‌باشد. به طور مثال شما می‌توانید نسخه minified شده را به Layout تان اضافه کنید. بعد از هربار تغییر در فایل Less این فایل نیز به روز خواهد شد:



همچنین برای دیگر اجزای بوت‌استرپ نیز می‌توانید به این صورت عمل کنید:

```
<!-- Before -->
<a href="#" class="btn danger large">Click me!</a>

<!-- After -->
<a href="#" class="annoying">Click me!</a>

a.annoying {
  .btn;
  .btn-danger;
```

```
  .btn-large;
}
```

خب، با استفاده از این حالت، کدهای HTML به‌صورت مرتب‌تر، قابل‌انعطاف‌تر و همچنین از لحاظ معنایی(Semantic) استاندارد خواهند بود. بنابراین با آمدن یک فریم‌ورک جدید، به راحتی امکان سوئیچ‌کردن برای ما میسر و آسان‌تر از قبل خواهد شد.

نظرات خوانندگان

نویسنده: سعید شیرزادیان
تاریخ: ۱۳۹۳/۰۴/۰۳ ۰:۰

سلام. مطلب جالبی بود.آیا قابلیت فوق برای حالت بدون Less امکان دارد. من در یک پروژه از بوت استرپ استفاده کردم و یکسری تغییرات فارسی نیز در آن اعمال شده است. راهکاری دارید؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۴/۰۳ ۰:۵۷

نسخه‌ی LESS راست به چپ بوت استرپ هم موجود است .
ضمنا نگارش 3.2 بوت استرپ قرار است که به صورت رسمی RTL را پشتیبانی کند.

نویسنده: حسن محمدی
تاریخ: ۱۳۹۳/۰۴/۰۳ ۱۳:۳۵

با سلام و تشکر از سایت بی نهایت خوبتون
مقاله عالی بود اگر امکانش بود یه تمپلیت کوچیک سایتو بوت استرپ و less بگید بی نهایت ممنون میشم. من خودم تو استفاده از less خیلی مشکل دارم خیلی‌ها هم مثل خودم می‌شناسنم. من آپدیت 2 ویژوال استودیو 2013 و webessenial رو نصب کردم تا فایل‌های less رو برام باز کرد.
فقط bootstrap.less رو بدون مشکل باز میکنه و نسخه css رو میاره ولی مثلا هر فایل دیگه ای مثل alerts.less رو باز میکنم پیغام خطا میده که مثلا variable @alert-padding is undefined یا چیزهای دیگرو میگه undefined به نظرتون مشکل از چیه؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۴/۰۳ ۱۳:۴۵

مشکلی ندارد . نسخه‌ی less بوت استرپ از چندین و چند فایل تشکیل شده که نهایتا از کامپایل این‌ها یک فایل نهایی تولید می‌شود. برای مثال اگر فایل bootstrap.less را باز کنید، ترتیب import فایل‌ها قابل ملاحظه هستند. ابتدا فایل variables.less ذکر شده، چون مقادیر پیش فرض متغیرهای سایر فایل‌ها در آن موجود است و همینطور الی آخر. فایل نهایی که سبب کامپایل تمام تغییرات خواهد شد، فایل bootstrap.less است.

نویسنده: حسن محمدی
تاریخ: ۱۳۹۳/۰۴/۰۳ ۱۳:۵۸

ممنون که پاسخ دادید من اگر بخوام مثلا navbar رو تغییر بدم باید bootstrap.less رو ویرایش کنم یا navbar.less ؟
بعد چیکار کنم که خطا برطرف شه و این ترتیبو بفهمه و مثلا navbar.less رو باز میکنم دیگه خطا نده؟
ممنون

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۴/۰۳ ۱۴:۲۶

- هیچکدام. فایل navbar.less تشکیل شده از یک سری متغیر. این متغیرها در فایل variables.less مقدار دهی شده‌اند. بنابراین برای تغییر آن باید فایل variables.less ویرایش شوند (در این فایل، Navbar را جستجو کنید).
- یک کپی از فایل اصلی bootstrap.less را مثلا به نام test.less ایجاد کنید (با همان محتوا). المان‌های مختلف آن‌را حذف کنید تا به حداقل وابستگی‌هایی که برای کامپایل navbar.less نیاز است برسید:

```
// Core variables and mixins
@import "variables.less";
@import "mixins.less";
```

```
// Core CSS
@import "forms.less";

// Components
@import "navbar.less";

// Utility classes
@import "utilities.less";
```