

Twitter Bootstrap یک فریم ورک CSS بسیار محبوب سورس باز تولید برنامه‌های وب به کمک CSS، HTML و جاوا اسکریپت است. این فریم ورک حاوی بسیاری از المان‌های مورد نیاز جهت تولید وب سایت‌هایی زیبا، مانند دکمه‌ها، عناصر فرم‌ها، منوها، ویجت‌ها و غیره است. تمام این‌ها نیز همانطور که عنوان شد بر مبنای CSS، HTML و جاوا اسکریپت تهیه شده‌اند؛ بنابراین در هر نوع فناوری سمت سروری مانند PHP، ASP.NET، روبی و امثال آن قابل استفاده است.

دریافت Twitter Bootstrap

محل اصلی دریافت Twitter Bootstrap، آدرس ذیل است:

<http://twitter.github.com/bootstrap>

البته ما از آن در اینجا به شکل خام فوق استفاده نخواهیم کرد؛ زیرا نیاز است قابلیت‌های استفاده در محیط‌های راست به چپ فارسی نیز به آن اضافه شود. برای این منظور می‌توانید از یکی از دو بسته نیوگت ذیل استفاده نمایید:

Twitter.BootstrapRTL, <https://nuget.org/packages/Twitter.BootstrapRTL>

RTL Twitter Bootstrap, <https://nuget.org/packages/Twitter.Bootstrap.RTL>

و یا حتی از منابع سایت <http://rbootstrap.ir> نیز می‌توان استفاده کرد. برای نمونه دستور زیر را در کنسول پاورشل ویژوال استودیو وارد نمایید تا اسکریپت‌ها و فایل‌های CSS مورد نیاز به پروژه جاری اضافه شوند:

```
PM> Install-Package Twitter.BootstrapRTL
```

پس از نصب، شاهد افزوده شدن چنین فایل‌هایی به یک پروژه ASP.NET MVC خواهیم بود:



در اینجا فایل‌های min، نگارش‌های فشرده شده فایل‌های js یا css هستند که با توجه به امکانات اضافه شده به ASP.NET MVC4، از آن‌ها استفاده نخواهیم کرد و برای افزودن و تعریف آن‌ها از امکانات Bundling and minification توکار فریم ورک ASP.NET MVC به نحوی که در ادامه توضیح داده خواهد شد، استفاده می‌کنیم.

فایل‌های png اضافه شده، آیکون‌های مخصوص Twitter Bootstrap هستند که اصطلاحاً به آن‌ها sprite images نیز گفته می‌شود. در این نوع تصاویر، تعداد زیادی آیکون در کنار هم، برای بهینه سازی تعداد بار رفت و برگشت به سرور جهت دریافت تصاویر، طراحی شده و قرار گرفته‌اند.

فایل‌های js این مجموعه اختیاری بوده و برای استفاده از ویجت‌های Twitter Bootstrap مانند آکاردئون کاربرد دارند. این فایل‌ها برای اجرا، نیاز به jQuery خواهند داشت.

افزودن تعاریف اولیه Twitter Bootstrap به یک پروژه ASP.NET MVC

امکانات Bundling and minification در نوع پروژه‌های نسبتاً جامع‌تر ASP.NET MVC به صورت پیش فرض لحاظ شده است. اما اگر یک پروژه خالی را شروع کرده‌اید، نیاز است بسته نیوگت آن‌را نیز نصب کنید:

```
PM> Install-Package Microsoft.AspNet.Web.Optimization
```

پس از آن، کلاس کمکی BundleConfig ذیل را به پروژه جاری اضافه نمایید. به کمک آن قصد داریم امکانات Bundling and minification را فعال کرده و همچنین به آن بگوییم اسکریپت‌ها و فایل‌های CSS تعریف شده را به همان نحوی که معرفی می‌کنیم، پردازش کن (توسط کلاس سفارشی AsIsBundleOrderer).

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

```

using System.Web;
using System.Web.Optimization;

namespace Mvc4TwitterBootstrapTest.Helper
{
    /// <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;

            Bundle newBundle;
            if (HttpContext.Current.IsDebuggingEnabled)
            {
                newBundle = new Bundle(virtualPath);
            }
            else
            {
                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);
        }
    }
}

```

نکته دیگری که در این کلاس سفارشی در نظر گرفته شده، عدم فعال سازی مباحث Bundling and minification در حالت Debug است. اگر در وب کانفیگ، تنظیمات پروژه را بر روی Release قرار دهید فعال خواهد شد (مناسب برای توزیع) و در حالت توسعه برنامه (حالت دیباگ)، این اسکریپت‌ها و فایل‌های CSS، قابلیت دیباگ و بررسی را نیز خواهند داشت.

پس از افزودن کلاس‌های کمکی فوق، به فایل layout پروژه مراجعه کرده و تعاریف ذیل را به ابتدای فایل اضافه نمائید:

```
@using Mvc4TwitterBootstrapTest.Helper
<!DOCTYPE html>
<html>
<head>
    <title>@ViewBag.Title</title>
    @BundleConfig.AddStyles("~/Content/css",
        "~/Content/bootstrap.css",
        "~/Content/bootstrap-responsive.css",
        "~/Content/Site.css"
    )
    @BundleConfig.AddScripts("~/Scripts/js",
        "~/Scripts/jquery-1.9.1.min.js",
        "~/Scripts/jquery.validate.min.js",
        "~/Scripts/jquery.unobtrusive-ajax.min.js",
        "~/Scripts/jquery.validate.unobtrusive.min.js",
        "~/Scripts/bootstrap.min.js"
    )
    @RenderSection("JavaScript", required: false)
</head>
```

و اگر برنامه را اجرا کنید، بجای حداقل 8 مدخل اشاره کننده به فایل‌های اسکریپت و CSS مورد نیاز یک برنامه ASP.NET MVC، فقط دو مدخل ذیل را مشاهده خواهید نمود:

```
<!DOCTYPE html>
<html>
<head>
    <title>Index</title>
    <link href="/Content/css?v=vsUQD00Jg4AJ-RZH8jSRRCu_rj12U1nZrmSsaUyxAc1" rel="stylesheet"/>
    <script src="/Scripts/js?v=GezdoTDiWY3acc3mI2Ujm_7nKKzh6Lu1Wr8TGyyLpW41"></script>
</head>
```

به این ترتیب تعداد رفت و برگشت‌های مرورگر به سرور، برای دریافت فایل‌های مورد نیاز جهت رندر صحیح یک صفحه، به شدت کاهش خواهد یافت. همچنین این فایل‌ها در حالت release فشرده نیز خواهند شد؛ به همراه تنظیم خودکار هدر کش شدن آن‌ها برای مدتی مشخص، جهت کاهش بار سرور و کاهش پهنای باند مصرفی آن.

مفاهیم پایه‌ای Twitter Bootstrap

الف) Semantic class names

به عبارتی کلاس‌های Twitter Bootstrap دارای نام‌هایی معنا دار و مفهومی می‌باشند؛ مانند کلاس‌های CSS ای، به نام‌های Success، Error، Info و امثال آن. این نام‌ها مفهومی را می‌رسانند؛ اما در مورد نحوه پیاده سازی آن‌ها جزئیاتی را بیان نمی‌کنند. برای نمونه می‌توان کلاسی را به نام redText ایجاد کرد. هر چند این نام، توضیحاتی را در مورد علت وجودی‌اش بیان می‌کند، اما بسیار ویژه بوده و در مورد جزئیات پیاده سازی آن نیز اطلاعاتی را ارائه می‌دهد. در این حالت redText معنایی ندارد. چرا یک Text باید قرمز باشد؟ برای مثال این متن قرمز است چون مثلاً شخصی، به آن رنگ ویژه علاقه دارد، یا اینکه قرمز است بخاطر نمایش خطایی در صفحه؟ به همین جهت در Twitter Bootstrap از نام‌های مفهومی یاده شده، مانند Error استفاده می‌شود. نام‌هایی معنا دار اما بدون دقیق شدن در مورد ریز جزئیات پیاده سازی آن‌ها. در این حالت می‌توان [قالب جدیدی را تدارک دید](#) و با ارائه تعاریف جدیدی برای کلاس Error و نحوه نمایش دلخواهی را به آن اعمال نمود. یا برای نمونه نام rightside را برای نمایش ستونی در صفحه، در نظر بگیرید. این نام بسیار ویژه است؛ اما Sematic name آن می‌تواند sidebar باشد تا بدون دقیق شدن در جزئیات پیاده سازی آن، در چپ یا راست صفحه قابل اعمال باشد. Semantic class names کلیدهای هستند جهت استفاده مجدد از قابلیت‌های یک فریم ورک CSS.

ب) Compositional classes

اکثر کلاس‌های Twitter Bootstrap دارای محدوده کاری کوچکی هستند و به سادگی قابل ترکیب با یکدیگر جهت رسیدن به نمایی خاص می‌باشند. برای مثال به سادگی می‌توان به یک table سه ویژگی color، hover و width برگرفته شده از Twitter Bootstrap را انتساب داد و نهایتاً به نتیجه دلخواه رسید؛ بدون اینکه نگران باشیم افزودن کلاس جدیدی در اینجا بر روی سایر کلاس‌های انتساب داده شده، تاثیر منفی دارد.

Conventions (ج)

برای استفاده از اکثر قابلیت‌های این فریم ورک CSS یک سری قراردادهای پیش فرضی وجود دارند. برای مثال اگر از کلاس توکار pagination به همراه یک سری ul و li استفاده کنید، به صورت خودکار یک pager شکل ظاهر خواهد شد. یا برای مثال اگر به یک html table کلاس‌های table-hover table-striped table را انتساب دهیم، به صورت خودکار قراردادهای پیش فرض table مجموعه Twitter Bootstrap به آن اعمال شده، به همراه رنگی ساختن یک درمیان زمینه ردیف‌ها و همچنین فعال سازی تغییر رنگ ردیف‌ها با حرکت ماوس از روی آن‌ها.

طرح‌بندی صفحات یک سایت به کمک Twitter Bootstrap

بررسی Grid layouts

Layout به معنای طرح‌بندی و چیدمان محتوا در یک صفحه است. یکی از متداولترین روش‌های طرح‌بندی صفحات چه در حالت چاپی و چه در صفحات وب، چیدمان مبتنی بر جداول و گریدها است. از این جهت که نحوه سیلان و نمایش محتوا از چپ به راست و یا راست به چپ را به سادگی میسر می‌سازند؛ به همراه اعمال حاشیه‌های مناسب جهت قسمت‌های متفاوت محتوای ارائه شده. Grid در طرح‌بندی، نمایش بصری نخواهد داشت اما در ساختار صفحه وجود داشته و مباحثی مانند جهت، موقعیت و یکپارچگی و یکدستی طراحی را سبب می‌شود. به علاوه مرورگرها و مفهوم Grid نیز به خوبی با یکدیگر سازگار هستند. در دنیای HTML و CSS، طراحی‌ها بر اساس مفهوم ساختار مستطیلی اشیاء صورت می‌گیرد:



برای نمونه در اینجا تصویر [CSS Box Model](#) را مشاهده می‌کنید. به این ترتیب، هر المان دارای محدوده‌ای مستطیلی با طول و عرض مشخص، به همراه ویژگی‌هایی مانند Margin، Border و Padding است. در سال‌های اولیه طراحی وب، عموماً کارهای طراحی صفحات به کمک HTML Tables انجام می‌شد. اما با پخته‌تر شدن CSS، استفاده از Tables برای طراحی صفحات کمتر و کمتر گشت تا اینکه نهایتاً فریم ورک‌های CSS ایی پدید آمدند تا طراحی‌های مبتنی بر CSS را با ارائه گریدها، ساده‌تر کنند. مانند 960 GS، Blue print، و ... Twitter Bootstrap که طراحی مبتنی بر گریدهای CSS ایی را به مجموعه قابلیت‌های دیگر خود افزوده است.

بررسی Fixed Grids

در اینجا در صفحه layout برنامه، یک Div دربرگیرنده دو Div دیگر را مشاهده می‌کنید:

```
<body>
  <div>
    <div>
      <h1>
        Title Title
      </h1>
      Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text
    </div>
    <div>
      @RenderBody()
    </div>
  </div>
</body>
```

اگر در این حالت صفحه را در مرورگر بررسی کنیم، تمام قسمت‌های نمایش داده شده به همین نحو از بالا به پایین صفحه قرار خواهند گرفت. اما قصد داریم این محتوا را، دو ستونی نمایش دهیم. Div به همراه Title در یک طرف صفحه و Div دربرگیرنده محتوای صفحات، در قسمتی دیگر.

برای اینکار در Twitter Bootstrap از کلاسی به نام row استفاده می‌شود که بیانگر یک ردیف است. این کلاس را به خارجی‌ترین Div موجود اعمال خواهیم کرد. در یک صفحه، هر تعداد row ایی را که نیاز باشد، می‌توان تعریف کرد. داخل این ردیف‌ها، امکان تعریف ستون‌های مختلف و حتی تعریف ردیف‌های تو در تو نیز وجود دارد. هر ردیف Twitter Bootstrap از 12 ستون تشکیل می‌شود و برای تعریف آن‌ها از کلاس span استفاده می‌گردد. در این حالت جمع اعداد ذکر شده پس از 12 باید 12 را تشکیل دهند.

```
<body>
  <div class="row">
    <div class="span7">
      <h1>
        Title Title
      </h1>
      Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text
    </div>
    <div class="span5">
      @RenderBody()
    </div>
  </div>
</body>
```

برای نمونه در اینجا کلاس row به خارجی‌ترین Div موجود اعمال شده است. دو Div داخلی هر کدام دارای کلاس‌های span ایی جهت تشکیل ستون‌های داخل این ردیف هستند. جمع اعداد بکارگرفته شده پس از آن‌ها، عدد 12 را تشکیل می‌دهد. به این ترتیب به یک طراحی دو ستونی خواهیم رسید.



در این تصویر، قسمت `RenderBody` کار رندر اکشن متد `Index` کنترلر `Home` برنامه را با `View`ی معادل کدهای ذیل، انجام داده است:

```
@{
    ViewBag.Title = "Index";
}
<h2>
    Index</h2>
<div class="hero-unit">
    <h2>@ViewBag.Message</h2>
    <p>
        This is a template to demonstrate the way to beautify the default MVC template
        using Twitter Bootstrap website. It is pretty simple and easy.</p>
    <p>
        <a href="http://asp.net/mvc" class="btn btn-primary btn-large" style="color: White;">
            To learn more about ASP.NET MVC visit &raquo;</a></p>
    </div>
```

درک نحوه عملکرد `Grid` در `Twitter Bootstrap`

در مثال ذیل 5 ردیف را مشاهده می‌کنید:

```
<div class="row">
```

```

<div class="span1">1</div>
<div class="span1">1</div>
<div class="span1">1</div>
<div class="span1">1</div>
<div class="span1">1</div>
<div class="span1">1</div>
<div class="span1">1</div>
<div class="span1">1</div>
<div class="span1">1</div>
<div class="span1">1</div>
<div class="span1">1</div>
</div>
<div class="row">
  <div class="span3">3</div>
  <div class="span4">4</div>
  <div class="span5">5</div>
</div>
<div class="row">
  <div class="span5">5</div>
  <div class="span7">7</div>
</div>
<div class="row">
  <div class="span3">3</div>
  <div class="span7 offset2">7 offset 2</div>
</div>
<div class="row">
  <div class="span12">12</div>
</div>

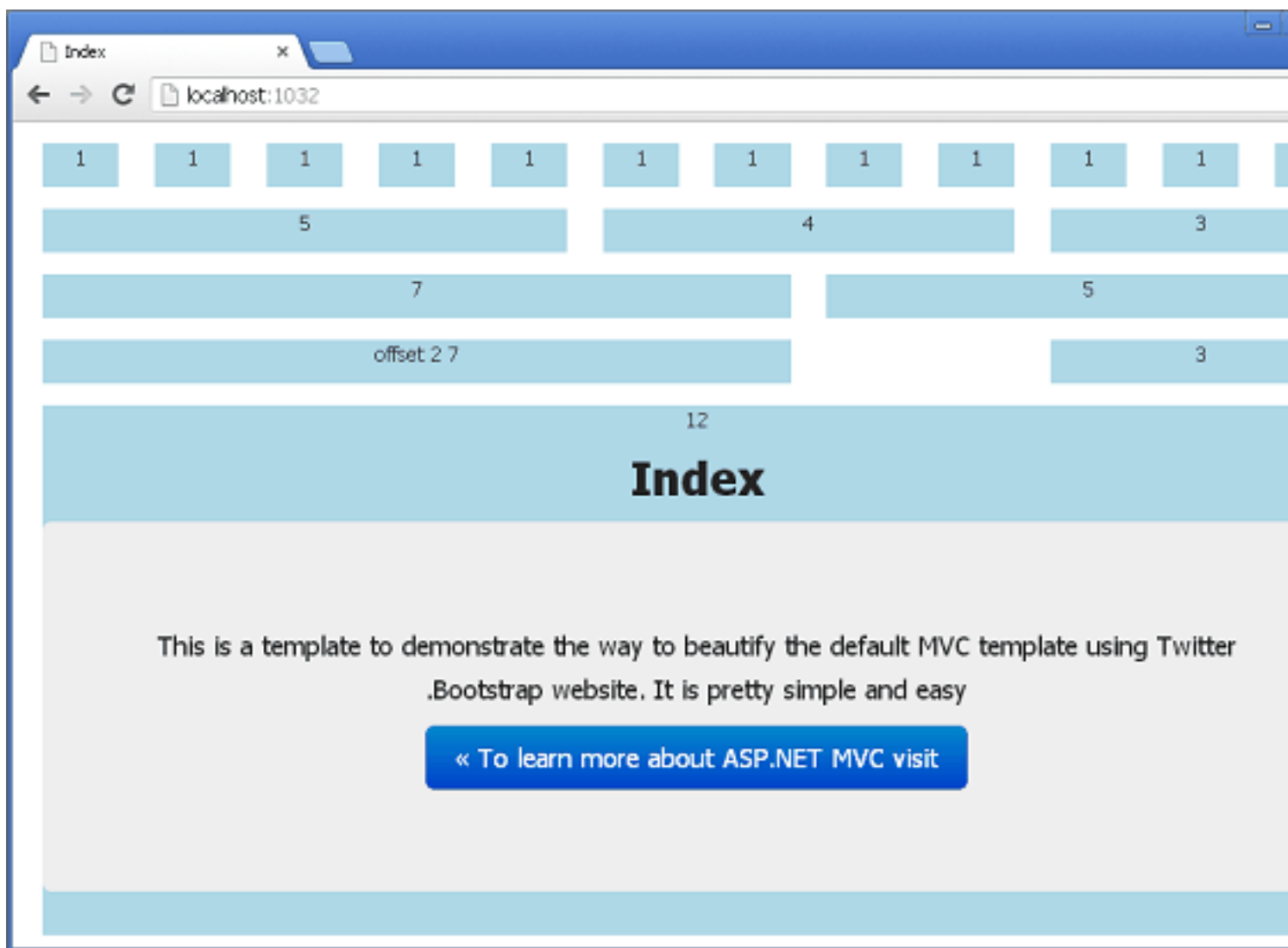
```

در هر ردیف، امکان استفاده از حداکثر 12 ستون وجود دارد که یا می‌توان مانند ردیف اول، 12 ستون را ایجاد کرد و یا مانند ردیف دوم، سه ستون با عرض‌های متفاوت (و یا هر ترکیب دیگری که به جمع 12 برسد). در ستون چهارم، از کلاس offset نیز استفاده شده است. این مورد سبب می‌شود ستون جاری به تعدادی که مشخص شده است به سمت چپ (با توجه به استفاده از حالت RTL در اینجا) رانده شود و سپس ترسیم گردد. یا اینکه می‌توان مانند ردیف آخر، یک ستون را به عرض 12 که در حقیقت 940 پیکسل است، ترسیم نمود. برای اینکه بتوانیم این گرید تشکیل شده و همچنین ستون‌ها را بهتر مشاهده کنیم، به فایل style.css سایت، تنظیم زیر را اضافه کنید:

```

[class*="span"]
{
background-color: lightblue;
text-align: center;
margin-top: 15px;
}

```

نکته جالب این گرید، Responsive یا واکنشگرا بودن آن است. در این حالت، عرض مرورگر را کم و زیاد کنید. خواهید دید که ستون‌ها در صورتیکه در عرض نمایشی جاری، قابل ارائه نباشند، به ردیف‌های بعدی منتقل خواهند شد. البته باید دقت داشت که این گرید هیچگاه یک ستون را نخواهد شکست. برای نمونه ردیف آخر، همواره با همان عرض ثابتش نمایش داده می‌شود و با کوچکتر کردن اندازه مرورگر، یک اسکرول افقی برای نمایش محتوای آن ظاهر خواهد شد.

یک نکته

اگر نمی‌خواهید که چنین رفتار واکنش‌گرایی بروز کند، نیاز است کلیه ردیف‌ها را در div ایی با کلاسی به نام container محصور کنید.

به این ترتیب ابتدا گرید نمایش داده شده، در میانه صفحه ظاهر خواهد شد (پیشتر از سمت راست شروع شده بود). همچنین دیگر با کوچک و بزرگ شدن اندازه مرورگر، ستون‌ها به شکل یک پشته بر روی هم قرار نخواهند گرفت. (اگر پس از این تنظیم، چنین قابلیتی را مشاهده نکردید و هنوز هم طراحی، واکنشگرا بود، تعریف bootstrap-responsive.css را نیاز است برای آزمایش، از هدر صفحه حذف کنید)

بررسی Fluid Grids

به گرید قسمت قبل از این جهت Fixed Grid گفته می‌شود که عرض هر span آن با یک عدد مشخص تعیین گشته است. اما در

حالت Fluid Grid، عرض هر span برحسب درصد تعیین می‌شود. بکارگیری درصد در اینجا به معنای امکان تغییر عرض یک ستون بر اساس عرض جاری Container آن است. در اینجا span12 دارای عرض 100 درصد خواهد بود. در مثال قبل، برای استفاده از Fluid grids، تنها کافی است هرجایی کلاسی مساوی row وجود دارد، به row-fluid تغییر کند. همچنین کلاس container را به container-fluid تغییر دهید. برای آزمایش آن، اندازه و عرض نمایشی مرورگر خود را تغییر دهید. اینبار مشاهده خواهید کرد که برخلاف حالت Fixed Grid، عرض ستون‌ها به صورت خودکار کم و زیاد می‌شوند. این مورد بر روی محتوای قرار گرفته در این ستون‌ها نیز تاثیر گذار است. برای مثال اگر یک تصویر را در حالت Fluid grid در ستونی قرار دهید، با تغییر عرض مرورگر، اندازه این تصویر نیز تغییر خواهند کرد؛ اما در حالت Fixed Grid خیر. حالت Fluid، شیوه متداول استفاده از bootstrap در اکثر سایت‌های مهمی است که تابحال از این فریم ورک CSS استفاده کرده‌اند.



مروری بر طراحی واکنشگرا یا Responsive

این روزها تعدادی از کاربران، با استفاده از ابزارهای موبایل و تبلت‌ها از وب سایت‌ها بازدید می‌کنند. هر کدام از این‌ها نیز دارای اندازه نمایشی متفاوتی می‌باشند. بنابراین نیاز خواهد بود تا حالت بهینه‌ای را جهت اینگونه وسایل نیز طراحی نمود. حالت بهینه در اینجا به معنای قابل خواندن بودن متون، امکان استفاده از لینک‌های ورود به صفحات مختلف و همچنین حذف اسکروول و مباحث زوم جهت مشاهده صفحات است.

یکی از ویژگی‌های CSS به نام media چنین قابلیتی را فراهم می‌سازد. برای نمونه قسمتی از فایل bootstrap-responsive.css دارای چنین تعاریفی است:

```
@media (min-width: 768px) and (max-width: 979px) {
    .hidden-desktop {
        display: inherit !important;
    }
}
```

توسط این تنظیمات، شیوه نامه تعیین شده، تنها به صفحاتی با اندازه‌هایی بین 768 تا 979 پیکسل اعمال می‌شود (تعدادی از تبلت‌ها برای نمونه).

Bootstrap برای مدیریت اندازه‌های مختلف وسایل موبایل، شیوه‌نامه‌های خاصی را تدارک دیده است که از اندازه 480px و یا کمتر، تا 1200px و یا بیشتر را پوشش می‌دهد.

به این ترتیب با اندازه 940px که پیشتر در مورد آن بحث شد، اندازه span12 به صورت خودکار به اندازه‌های متناسب با صفحات نمایشی کوچکتر تنظیم می‌گردد. همچنین برای اندازه‌های صفحات کوچکتر از 768px به صورت خودکار از Fluid columns استفاده می‌گردد.

تنها کاری که برای اعمال این قابلیت باید صورت گیرد، افزودن تعاریف فایل bootstrap-responsive.css به هدر صفحه است که در قسمت قبل انجام گردید. این فایل باید پس از فایل اصلی bootstrap.css اضافه شود.

کلاس‌های کمکی طراحی واکنشگرا

Bootstrap شامل تعدادی کلاس کمکی در فایل bootstrap-responsive.css خود می‌باشد شامل visible-phone، visible-tablet و visible-desktop به علاوه hidden-phone، hidden-tablet و hidden-desktop. به این ترتیب می‌توان محتوای خاصی را جهت وسایل ویژه‌ای نمایان یا مخفی ساخت.

برای مثال محتوای مشخص شده با کلاس hidden-desktop، در اندازه وسایل دسکتاپ نمایش داده نخواهد شد.

برای آزمایش آن، شش div را با کلاس‌های یاد شده و محتوایی دلخواه ایجاد کرده و سپس اندازه عرض مرورگر را تغییر دهید تا بهتر بتوان مخفی یا نمایان ساختن محتوا را بر اساس اندازه صفحه جاری درک کرد.

یکی از کاربردهای این قابلیت، قرار دادن تبلیغاتی با اندازه‌های تصاویری مشخص برای وسایل مختلف است؛ بجای اینکه منتظر شویم تا Fluid layout اندازه تصاویر را به صورت خودکار کوچک یا بزرگ کند، که الزاما بهترین کیفیت را حاصل نخواهد ساخت.

```
<div class="container-fluid">
  <div class="row-fluid">
    <div class="span4">
      <div class="visible-phone">
        visible-phone</div>
      </div>
    <div class="span4">
      <div class="visible-tablet">
        visible-tablet</div>
      </div>
    <div class="span4">
      <div class="visible-desktop">
        visible-desktop</div>
      </div>
    </div>
  </div>
</div>
<div class="container-fluid">
  <div class="row-fluid">
    <div class="span4">
      <div class="hidden-phone">
        hidden-phone</div>
      </div>
    </div>
  </div>
</div>
```

```
<div class="span4">
  <div class="hidden-tablet">
    hidden-tablet</div>
  </div>
<div class="span4">
  <div class="hidden-desktop">
    hidden-desktop</div>
  </div>
</div>
</div>
```



نظرات خوانندگان

نویسنده: احمدعلی شفیعی
تاریخ: ۱۰:۸ ۱۳۹۲/۰۳/۱۹

محدودیت ۱۲ تایی ستون‌ها به اندازه‌ی صفحه بستگی دارد. برای من ۱۳ تاست.

نویسنده: وحید نصیری
تاریخ: ۱۷:۳۵ ۱۳۹۲/۰۳/۱۹

شعار این کتابخانه چنین چیزی هست: « [Bootstrap is built on responsive 12-column grids](#) ». یعنی حالت پیش فرض طراحی آن 12 ستونی است. اگر به فایل css آن مراجعه کنید، یک چنین عرض‌هایی قابل مشاهده است:

```
.span12 {
  width: 940px;
}
.span1 {
  width: 60px;
}
```

البته این‌ها برای حالت fixed است و نه fluid که بر اساس درصد تعریف شده

```
.row-fluid .span12 {
  width: 100%;
  *width: 99.94680851063829%;
}
```

container هم با عرض 940 پیکسل تعریف شده. همچنین یک GutterWidth را هم باید در بین ستون‌ها در نظر داشت. حالت پیش فرض:

```
@gridColumns: 12;
@gridColumnWidth: 60px;
@gridGutterWidth: 20px;
```

البته میشه مثلا عرض رو بیشتر از 940 تعریف کرد. برای اینکار باید css رو ویرایش و از ابزارهایی مانند [less استفاده کرد](#) و یا امکان [سفارشی سازی اون](#) هم هست.

نویسنده: وحید نصیری
تاریخ: ۱۲:۴۸ ۱۳۹۲/۰۳/۲۰

یک نکته تکمیلی

در مطلب فوق یک چنین تعریفی را مشاهده نمودید:

```
@BundleConfig.AddStyles("~/Content/css",
    "~/Content/bootstrap.css",
    "~/Content/bootstrap-responsive.css",
    "~/Content/Site.css"
)
```

در پوشه content، فایل‌های bootstrap.min.css و bootstrap.css وجود دارند (اگر بسته نیوگت آن‌را نصب کنید). در این بین، اگر فایل bootstrap.css را اندکی ویرایش کنید (مثلا نام فونت‌ها یا اندازه آن‌ها را تغییر دهید)، متوجه خواهید شد که تغییرات شما اصلا اعمال نمی‌شوند. علت چیست؟! سیستم Bundling به صورت هوشمند در صورت وجود فایل‌های css.min، از آن‌ها استفاده خواهد کرد (حتی در حالت دیباگ) و نه

از فایل bootstrap.css ذکر شده در تنظیمات. به همین جهت ممکن است مدتی سر درگم شوید که چرا هر تغییری را که اعمال می‌کنم، تاثیری ندارد!
بنابراین یا فایل‌های css.min را حذف کنید و یا مکان آن‌ها را تغییر دهید تا در کنار فایل‌های غیرفشرده شده پیش فرض نباشند.

نویسنده: یزدان
تاریخ: ۱۳۹۲/۰۳/۳۱ ۱:۳۷

آیا برای web form هم کارایی داره و میشه در web form از اون استفاده کرد ؟

```
@BundleConfig.AddStyles("~/Content/css",
    "~/Content/bootstrap.css",
    "~/Content/bootstrap-responsive.css",
    "~/Content/Site.css"
)
```

در وب فرم به چه صورت این رو اضافه کنیم ؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۳/۳۱ ۹:۰۰

در ASP.NET 4.5 مباحث Bundling and Minification [جزو استاندارد آن](#) است. در ASP.NET 4.0 می‌شود از امکانات ASP.NET MVC 4 هم استفاده کرد (همان Bundling and Minification کامپایل شده برای دات نت 4 است).

نویسنده: ایلیا اکبری فرد
تاریخ: ۱۳۹۲/۰۴/۰۱ ۱۱:۵۰

با سلام.

در قسمت Bundling وقتی از کد شما استفاده میکنم تنها فقط Site.css را باندل میکند و سایر css ها باندل نمی‌شوند. وقتی کد زیر را کامنت کردم مشکل حل شد.

```
@BundleConfig.AddStyles("~/Content/css",
    "~/Content/bootstrap.css",
    "~/Content/Site.css")
...
var existing = BundleTable.Bundles.GetBundleFor(virtualPath);
//if (existing != null)
//    return;
```

کامنت کردن این کد مشکلی ایجاد میکند؟

نویسنده: میثم
تاریخ: ۱۳۹۲/۰۴/۰۹ ۱۱:۴۰

فایل bootstrap.css بصورت ناقص فشرده می‌شود.
بطور دقیق‌تر تا رسیدن به کاراکترهای :: بدون مشکل کار پیش می‌رود.
در حالت اجرا وقتی فایل را css را باز میکنیم فقط کلاس‌ها و شناسه‌های قبل از :: وجود دارند.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۴/۰۹ ۱۱:۵۵

در پروژه MVC خود، خط فرمان پاورشل نیوگت را باز کنید و دستور Update-Package را صادر کنید. تقریباً تمام اجزای MVC4 مرتباً به روز می‌شوند؛ منجمله همین بحث Web.Optimization آن:

PM> Update-Package

نویسنده: سیروان عفیفی
تاریخ: ۱۹:۲۲ ۱۳۹۲/۰۵/۱۸

دو مدخل برای من اضافه میشن

```
<link href="/Content/myCSS?v=LBa4VrARIJBaPHOXLQoCWVPKu_c6QBb4D7npv-dq5TA1" rel="stylesheet"/>
<script src="/Scripts/js?v=rk-13czPS8KW6usE0pK8aIglVqPOVEKWoSXV4N-5N2Q1"></script>
```

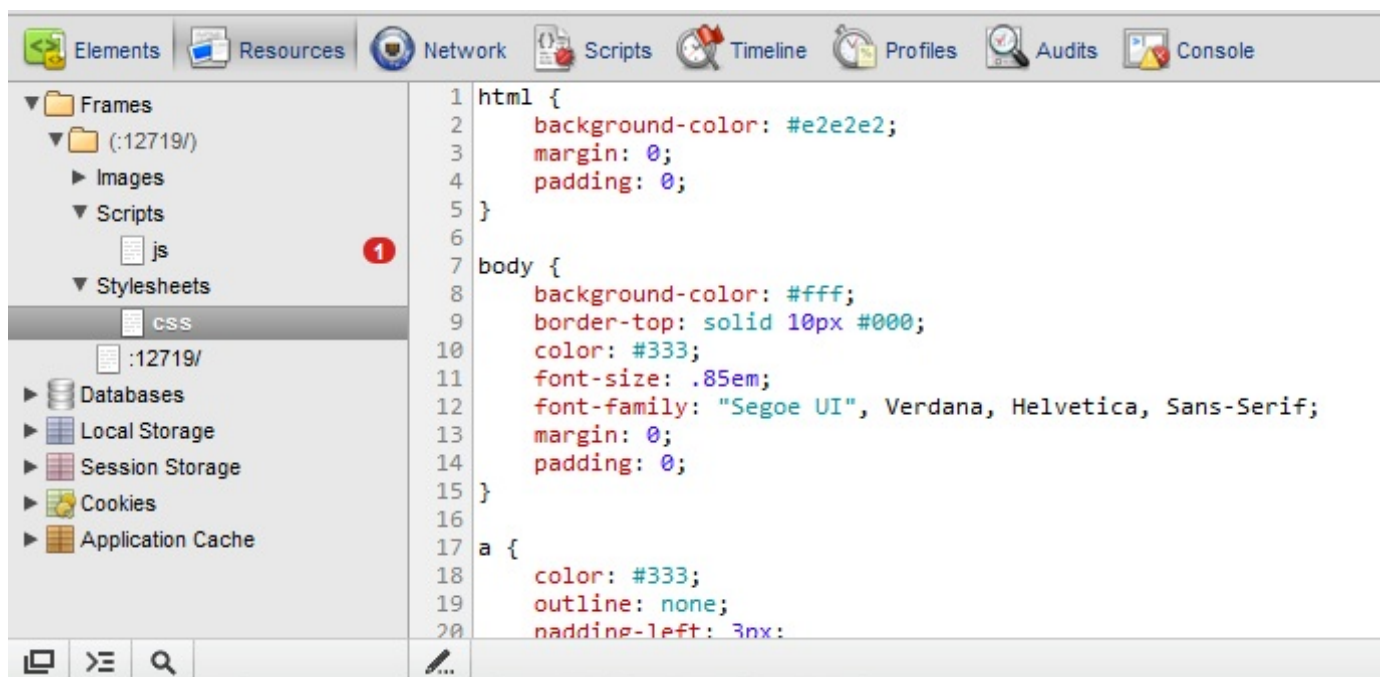
منتها برای CSS فقط فایل Style.css و برای جاوااسکریپت فقط jQuery رو اضافه میکنه، به نظرتون مشکل چی میتونه باشه؟

نویسنده: وحید نصیری
تاریخ: ۲۱:۷ ۱۳۹۲/۰۵/۱۸

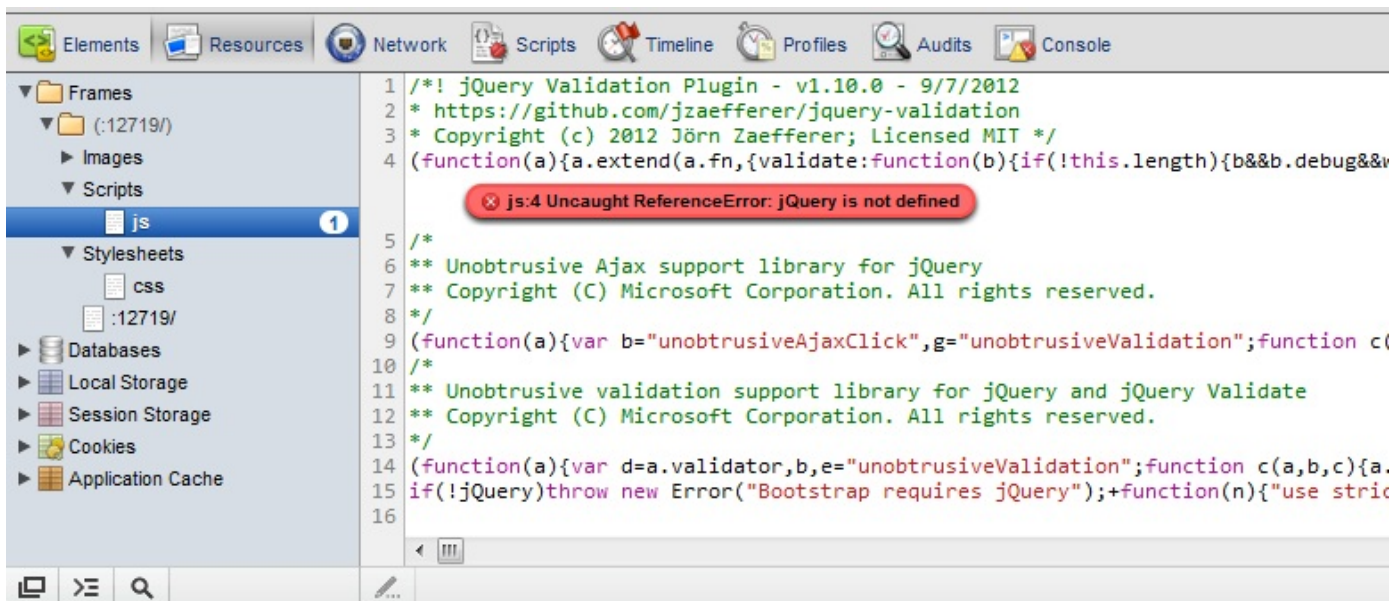
اگر واقعا مشکلی وجود داشته باشد، در همین لینک‌های پویایی که ارائه دادید، با کلیک بر روی آن‌ها عبارت «Minification failed. Returning unminified contents» ظاهر خواهد شد ([یک نمونه مثال در اینجا](#)) و سپس در سطرهای بعدی علت خطای کامپایل ذکر می‌شود که باید برطرف شود.

نویسنده: سیروان عفیفی
تاریخ: ۰:۴۱ ۱۳۹۲/۰۵/۱۹

حاصل کلیک بر روی لینک‌های پویا تولید شده به صورت زیر است :
مثلا با کلیک بر روی css تولید شده فایل Style.css را مشاهده میکنم :



و با کلیک بر روی فایل js تولید شده نیز به صورت زیر می‌باشد :



نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۵/۱۹ ۰۵:۵۸

چند سؤال اینجا هست که از راه دور قابل بررسی نیست:

- چطور تعاریف فایل‌ها رو در کدها اضافه کردید؟ (اگر خطای نبود jQuery رو بدون خطای کامپایل که ذکر شد گرفتید، یعنی اصلا jQuery تعریف نشده در تعریف bundle نهایی)
- ساختار پوشه‌های شما چیست؟ محل قرارگیری این فایل‌ها به چه ترتیبی است؟
- از چه نگارشی از اسکریپت‌ها و فایل‌ها استفاده می‌کنید؟ از کجا تهیه شده‌اند؟

ضمناً هیچ الزامی برای استفاده از minification در این مبحث نیست. فقط برای تکمیل بحث عنوان شده. به علاوه در قسمت‌های بعدی این بحث، سورس هر قسمت [پیوست شده](#) که شامل فایل‌های اسکریپت و CSS متناظر با هر بحث می‌شود. این فایل‌ها در این سری ویژه، مورد تأیید هستند و آزمایش شده‌اند.

نویسنده: فرزاد
تاریخ: ۱۳۹۲/۰۶/۱۸ ۱۱:۱۲

سلام؛ بعد از اضافه کردن کلاس AsIsBundleOrderer به Bundleconfig با خطای:

```
'MvcApplication7.Helper.AsIsBundleOrderer' does not implement interface member
'System.Web.Optimization.IBundleOrderer.OrderFiles(System.Web.Optimization.BundleContext,
System.Collections.Generic.IEnumerable<System.Web.Optimization.BundleFile>')
```

مواجه می‌شود. در واقع از کلاس AsIsBundleOrderer ارور می‌گیره. ممنون میشم راهنمایی کنید.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۶/۱۸ ۱۱:۲۷

- در نگارش جدید System.Web.Optimization، امضای FileInfo متد OrderFiles تبدیل شده به BundleFile. (همین تغییر نام کافی است)
- در زمان نگارش این مطلب، FileInfo بوده.

نویسنده: احسان شیروان
تاریخ: ۱۷:۳۲ ۱۳۹۲/۰۷/۰۴

با سلام ضمن تشکر از شما بابت آموزش‌های بسیار خوب
یه سوال داشتم من از C#.net , VS2010 دارم استفاده می‌کنم برای استفاده کردن از Bootstrap تمامی مراحل یاد شده توسط
شما رو مو به مو اجرا کردم اما متاسفانه به هیچ وجه کار نداد البته خطا هم نمیده اما مباحث مربوط به ریسپانسیو بودن صفحه و
حتی نمایش spanها جوابی نگرفتم آیا این بدلیل ورژن VS2010 هست یا دات نت؟ یا یه چیز دیگه ؟
ظاهرا سایتی که ایجاد کردم هیچ درکی از span نداره در حالی که تمام فایل‌های bootstrap رو طبق آموزش روی سایت نصب
کردم
ممنون میشم راهنماییم نمایید

نویسنده: وحید نصیری
تاریخ: ۱۷:۵۷ ۱۳۹۲/۰۷/۰۴

- مطالبی که من در مورد بوت استرپ منتشر کردم مربوط به نگارش 2 آن است. نگارش 3 آن (که مدتی است منتشر شده و از
طریق لینک‌های ابتدای بحث به صورت پیش فرض دریافت می‌شود) دگرگون شده و خیلی از مباحث آن تغییر کرده‌اند. فایل‌های
بوت استرپ 2 در قسمت‌های بعدی این بحث، [پیوست شده‌اند](#) که شامل فایل‌های اسکریپت و CSS متناظر با هر بحث می‌شود .
+ [راهنمای مهاجرت از بوت استرپ 2 به 3](#)

نویسنده: صابر فتح الهی
تاریخ: ۱۶:۴۴ ۱۳۹۲/۰۹/۱۰

سلام؛ من توی ASP.NET Webform از قابلیت Bundle استفاده کردم (قبلا درست کار می‌کرد) اما توی نسخه 4.5 کار نمی‌کنه، بعد از
دیباگ فهمیدم اصلا فایلها با BundleCollection اضافه نمیشه و زمانی که می‌خواد خروجی بگیره با خطای نبود مدخل مورد نظر
مواجه میشه. Update-Package را هم اجرا کردم اما همون خطا رو دریافت می‌کنم.

نویسنده: وحید نصیری
تاریخ: ۱۸:۴۳ ۱۳۹۲/۰۹/۱۰

جزئیات روش خودتان را با روش یکی از اعضای تیم دات نت مطابقت بدید (مربوط به ASP.NET 4.5 است):
[Adding Bundling and Minification to Web Forms](#)

نویسنده: افشار محبی
تاریخ: ۱۷:۳۷ ۱۳۹۲/۰۹/۲۰

باید span را به col-md عوض کنید. همان طور که در راهنمای مهاجرت از ۲ به ۳ که لینکش را آقای نصیری مشخص کرده اند
توضیح داده شده:

replace .span* with .col-md-*

بنده هم دقایقی را معطل کشف این تغییر بوت‌استرپ شدم.

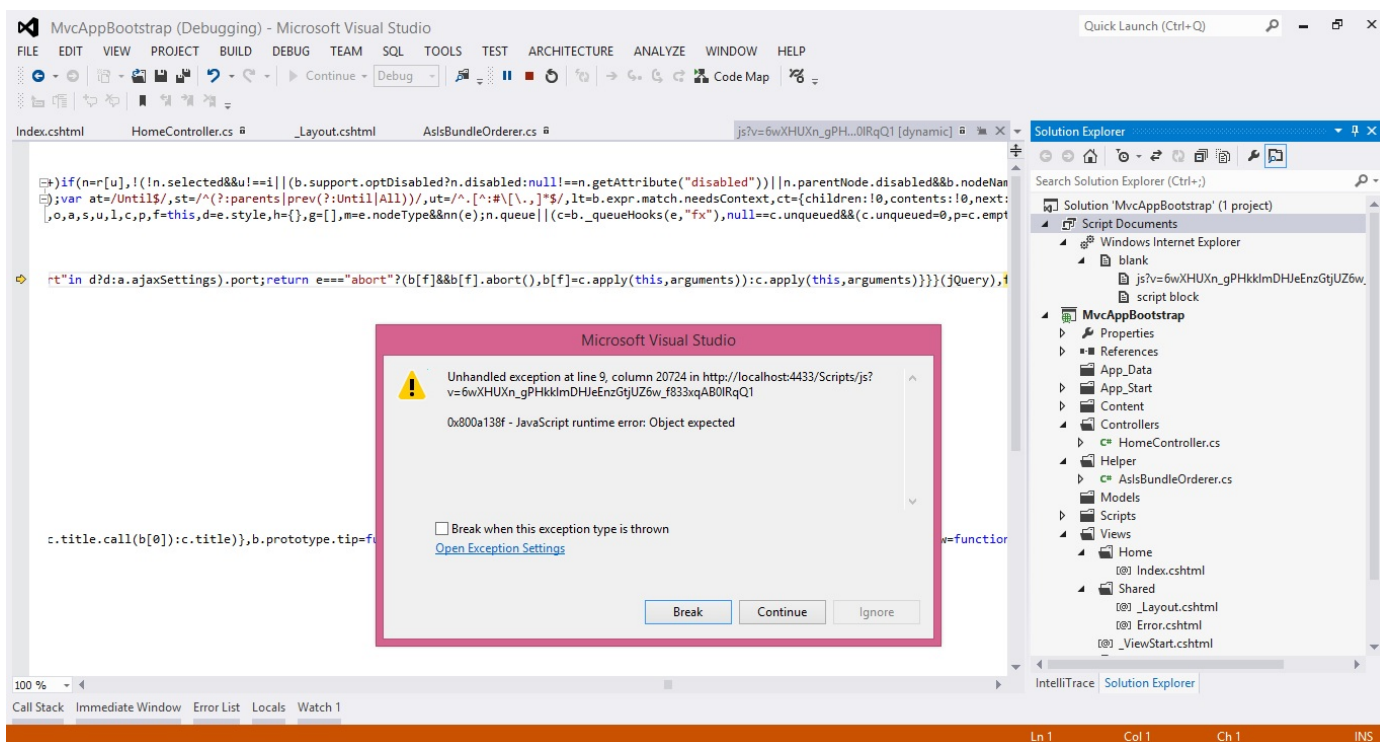
نویسنده: وحید نصیری
تاریخ: ۱۷:۴۵ ۱۳۹۲/۰۹/۲۰

توضیحات تکمیلی در اینجا: « [بررسی سیستم جدید گرید بوت استرپ 3](#) »

نویسنده: مجتبی فخاری
تاریخ: ۲۳:۰۵ ۱۳۹۲/۱۲/۰۳

بعد از نصب از طریق دستور Install-Package Twitter.BootstrapRTL و بعد هم دستور Install-Package PM>

Microsoft.AspNet.Web.Optimization بسته‌ها رو نصب کردم و یک پوشه به اسم Helper نیز اضافه نمودم و کلاسهای AsIsBundleOrderer را اضافه نمودم و در layout هم کدهای شما را آوردم ولی حالا در زمان اجرا خطای زیر را میدهد؟



میشه راهنمایی بفرمائید؟ با تشکر

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۲/۰۳ ۲۳:۱۳

مطلب جاری و سری مرتبط با آن برای بوت استرپ 2 نوشته شد. امروز اگر دستور `install-Package Twitter.BootstrapRTL` را صادر کنید، بوت استرپ 3.1 را نصب می‌کند که سیستم آن در بسیاری از موارد 180 درجه دگرگون شده نسبت به نگارش 2. بنابراین برای کار با نگارش 2 بهتر است دستور زیر را صادر کنید:

```
PM> Install-Package Twitter.BootstrapRTL -Version 2.3.2.1
```

و یا از مطالب [ارتقاء به بوت استرپ 3](#) کمک بگیرید.
ضمناً فایل‌های مرتبط با نگارش 2، در قسمت‌های بعدی همین سری، در نظرات هر قسمت [پیوست شده‌اند](#).

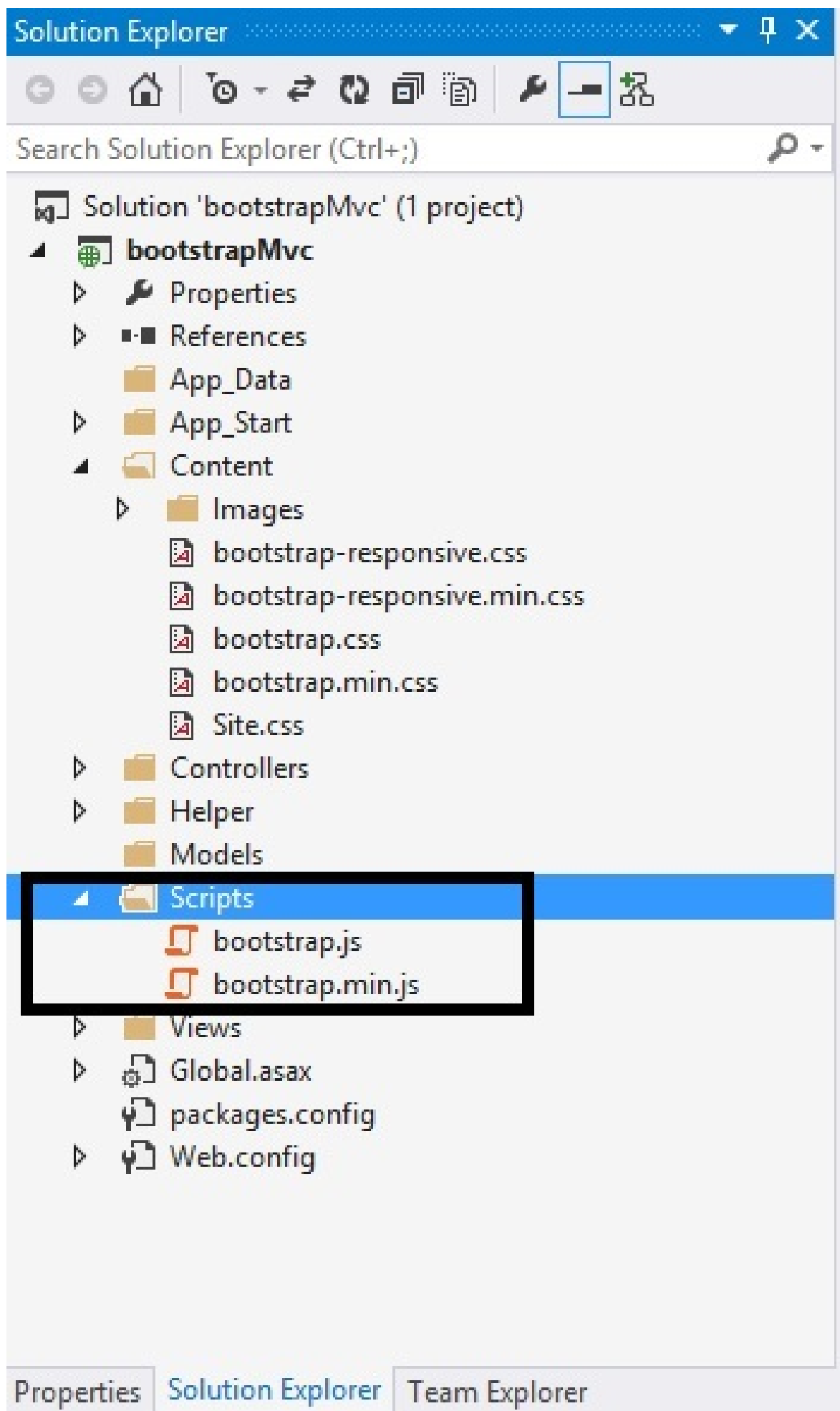
نویسنده: ایمان باقری
تاریخ: ۱۳۹۳/۰۱/۱۹ ۱۳:۲۵

سلام. من توی پروژم تو قسمت اسکریپت‌ها فقط `bootstrap.js` و `bootstrap.min.js` رو دارم و مابقی جاوا اسکریپت‌ها و جی کوئری هارو ندارم. با چه دستوری میتونم اونارو اضافه کنم. چون الان منوهای پایین افتادنی و یکسری چیزها دیگه کار نمیکنه. در ضمن پروژه خالی mvc شروع کردم. (تو حالت basic خودش پیش فرض داره)

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۱/۱۹ ۱۳:۲۸

[کمی بالاتر](#) پاسخ دادم «مطلب جاری و سری مرتبط با آن برای بوت استرپ 2 نوشته شد ...». (چندین بار)

من هم بوت استرپ 2 رو گرفتم فقط فایل‌های جی کوئری و validation اون رو ندارم
فایل‌های jquery-1.9.1.min.js jquery.unobtrusive-ajax.min.js jquery.validate.min.js و
jquery.validate.unobtrusive.min.js



نویسنده:

وحید نصیری

تاریخ:

۱۳:۴۱ ۱۳۹۳/۰۱/۱۹

عنوان کردید حالت Basic پروژه‌های MVC 5 را شروع کردید. پیش فرض استاندارد MVC 5 [بوت استرپ 3](#) است و نه دو. فایل‌های کامل [بوت استرپ 2](#) را از طریق دستور ذیل که پیشتر نیز ذکر شده بود دریافت کنید (تمام اجزای js آن با هم یکی شده‌اند در قالب یک فایل نهایی):

```
PM> Install-Package Twitter.BootstrapRTL -Version 2.3.2.1
```

نویسنده:

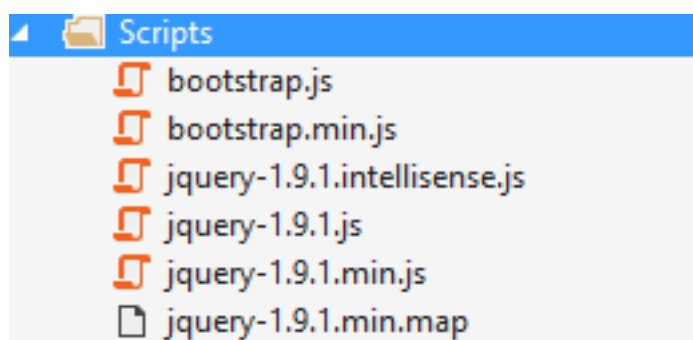
ایمان باقری

تاریخ:

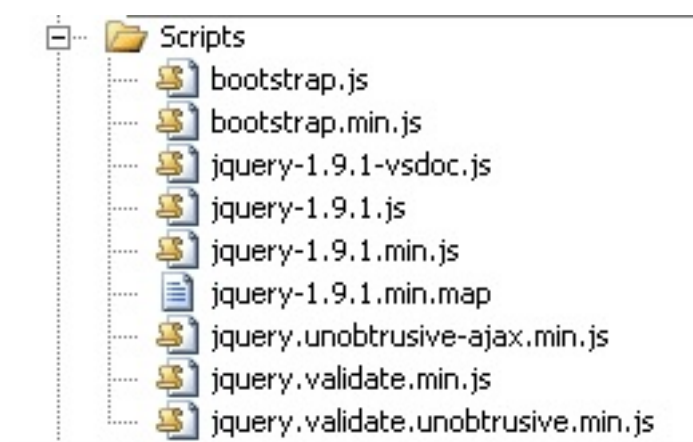
۱۴:۳ ۱۳۹۳/۰۱/۱۹

من حالت empty پروژه 4 mvc رو شروع کردم (بالا هم عرض کردم) و بوت استرپ هم از سایت rbootstrap.ir نسخه 2.3.2 v رو گرفتم.

البته الان که از دستور `PM> Install-Package Twitter.BootstrapRTL -Version 2.3.2.1` استفاده کردم jquery-1.9.1.min.js رو هم نصب کرد



و مشکل منوها حل شد. فقط فایل‌های validation و ajax رو همچنان ندارم



نویسنده:

وحید نصیری

تاریخ:

۱۴:۱۲ ۱۳۹۳/۰۱/۱۹

فایل‌های دیگری را که یاد کردید، جزو بسته‌ی بوت استرپ نیستند. آن‌ها را دستی از یک پروژه‌ی MVC دیگر کپی کنید. از نیوگت هم قابل دریافت هستند: ([^](#) و [^](#))

نویسنده: M.Shakeri

تاریخ: ۱۴:۲۱ ۱۳۹۳/۰۶/۲۱

شما اشاره کردید به این که اگر پروژه رو در حالت release قرار بدیم فشرده سازی اعمال میشه یعنی اینکه در فایل web.config در قسمت compilation خصوصیت debug رو حذف کنیم:

```
<compilation debug="true" targetFramework="4.5" />
```

این کارو انجام دادم اما وقتی سورس صفحه رو نگاه می‌کنم فشرده سازی اعمال نشده به فایل‌های css و js !

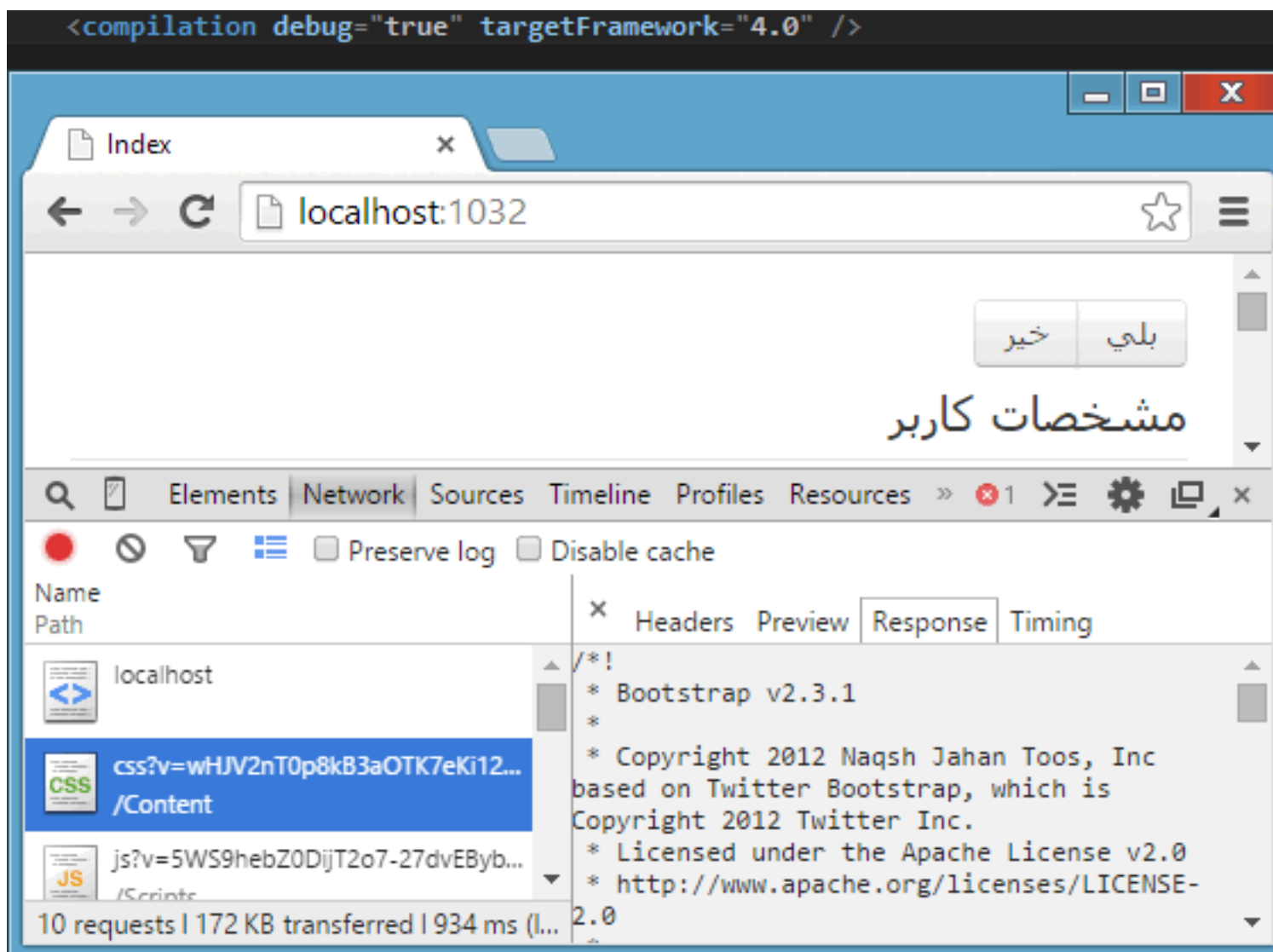
نویسنده: وحید نصیری

تاریخ: ۱۴:۴۲ ۱۳۹۳/۰۶/۲۱

پس از بررسی مجدد، مشکلی مشاهده نشد:
- رندر حالت فشرده شده در حالت غیر دیباگ:



- رندر حالت معمولی و اصل آن در حالت دیباگ:



در کل اگر این روش برای شما کار نمی‌کند، شرط `HttpContext.Current.IsDebuggingEnabled` را از متد `addBundle` حذف کنید و `else` آنرا نگه دارید.

نویسنده: امیرتنها
تاریخ: ۱۳۹۳/۱۰/۰۵ ۲۳:۱۷

با سلام

هنگام اجرا این خطا رو می‌ده .

'does not contain a definition for 'AddStyles

Line 19: `@BundleConfig.AddStyles("~/Content/css",`

Line 20: `"~/Content/bootstrap.css",`

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۱۰/۰۵ ۲۳:۴۵

یکبار نظرات این بحث را کامل مطالعه کنید. این مطلب برای بوت استرپ 2 نوشته شد، در آن زمان. نگارش 3 آن [در اینجا](#) بحث شده. همچنین این کلاس‌ها [تغییراتی داشته‌اند](#) که پیشتر در نظرات بحث شده.