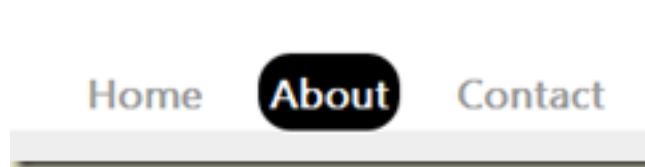


راه حل‌های مختلفی جهت Highlight کردن لینک صفحه جاری وجود دارد و مهم‌ترین کاربرد آن در منوی اصلی سایت است.



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 بهینه‌ای 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:

۹:۲۶ ۱۳۹۱/۰۶/۲۰

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

نویسنده:

وحید نصیری

تاریخ:

۱۲:۳۷ ۱۳۹۱/۰۷/۰۶

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

```
//-----جاری صفحه به ازای لینکهای بالای صفحه
$(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 می کند.

نویسنده:

امیرحسین مرجانی

تاریخ:

۱۶:۲۳ ۱۳۹۱/۰۷/۲۷

ممنونم

نکته خوبی گفتید ... چندوقتی بود دنبال مطلب شما می گشتم که الان به چشمم خورد.

نویسنده:

صابر فتح الهی

تاریخ:

۱۲:۴۸ ۱۳۹۲/۰۱/۰۷

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 می شه که حتما آدرس مورد نظر با آدرس لینک برابر باشه، در صورتی که لینک مورد نظر یک صفحه فرعی باشه و ان لینک در لیست منو وجود نداشته باشه هیچ لینکی رنگی نمی شه، مثلا در سایت [دات نت تیپس](#) در صورتی که کاربر روی لینک جزئیات یک پروژه کلیک کنه لینک پروژه ها 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 می کنم لطفا تصحیح بفرمایید.

نویسنده:

وحید نصیری

تاریخ:

۱۲:۵۸ ۱۳۹۲/۰۱/۰۷

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 پروژه ها 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:

۱۵:۱۱ ۱۳۹۲/۰۱/۰۸

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 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:

۰۳:۳۳ ۱۳۹۲/۰۳/۱۸

منظور از ناحیه headermenu چیست؟

نویسنده:

وحید نصیری

تاریخ:

۱۷:۱۴ ۱۳۹۲/۰۳/۱۹

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

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 کافیت 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)