

دو آدرس `www.site.com` و `site.com` را در نظر بگیرید. در حالت متداول، هر دو به یک معنا هستند و هر دو به ریشه یک سایت اشاره می‌کنند؛ اما از دیدگاه مسایل اعتبار سنجی، خیر. کوکی‌های این دو یکسان نبوده و برای کاربران مشکل ساز خواهند شد. کاربری که از طریق آدرس `site.com` به سایت وارد شده، زمانیکه به لینک مفروض `www.site.com` وارد می‌شود (مثلا یکی از کاربران در بین مطالب ارسالی به این آدرس لینک داده) دیگر حالت لاگین قبلی خود را نخواهد داشت و به این ترتیب تصور می‌کند که سایت باگ دارد.

برای رفع این مشکل می‌توان کلیه کاربرانی را که به آدرس `site.com` وارد می‌شوند، به صورت خودکار به آدرس `www` دار آن هدایت کرد و مدیریت آدرس‌های سایت را یک دست و یکنواخت نمود:

```
using System.Web.Mvc;

namespace WebToolkit
{
    /// <summary>
    /// Ensure all of the asp.net mvc urls have www.
    /// </summary>
    public class MandatoryWww : ActionFilterAttribute
    {
        public override void OnActionExecuting(ActionExecutingContext filterContext)
        {
            if (!filterContext.RequestContext.HttpContext.Request.IsLocal)
            {
                string url =
filterContext.RequestContext.HttpContext.Request.Url.AbsoluteUri.ToLowerInvariant();
                if (!url.Contains("www"))
                {
                    url = url.Replace("http://", "http://www.");
                    url = url.Replace("https://", "https://www.");
                    filterContext.Result = new RedirectResult(url, true);
                }
            }
            base.OnActionExecuting(filterContext);
        }
    }
}
```

و برای استفاده از آن در فایل `global.asax.cs` برنامه خواهیم داشت:

```
public static void RegisterGlobalFilters(GlobalFilterCollection filters)
{
    filters.Add(new MandatoryWww());
}
```

نظرات خوانندگان

نویسنده: Javad Amiry
تاریخ: ۱۳۹۱/۰۹/۰۲ ۱:۱۸

سلام و تشکر.
اشکال این روش اینه که اگه درخواستی به برنامه برسه که توسط actionها پاسخ داده نمیشه، باز کماکان با مشکل قبلی مواجهیم. مثلا در رابطه با هندلرها یا فایل‌های استاتیک (البته خوب شرایطی هم هست که بعنوان مثال در رابطه فایل‌های استاتیک نیاز نیست ریدایرکشن صورت بگیره - کلی عرض میکنم). روش بهتر استفاده از یه HttpModule هست که البته باز این اشکال رو داره که (با توجه به تنظیمات) ممکنه برای همه درخواست‌ها اجرا نشه.
روشی بعدی که به نظر ایده آل هست و خودم همیشه استفاده میکنم UrlRewrite هست که امروزه تقریبا میشه گفت روی همه سرورها نصبه. یا اگه هم نصب نباشه، اکثر سرورا دسترسی ریموت به IIS میدن که میشه نصبش کرد.
ضمن این که جدای موضوع کوکی‌ها که خودش مبحث مفصلی هستش و فقط مربوط به اعتبارسنجی نمیشه، موضوع مهم دیگه نحوه برخورد موتورهای جستجو -مخصوصا گوگل- با سایتهایی که هر دو آدرس با و بدون www شون در دسترسه. یعنی مثلا از دیدگاه گوگل، حتما یکی از آدرس‌ها باید Redirect Permanently بشه روی اون یکی. البته این داستان، سر درازی داره. سعی کردم خلاصه کنم. باز هم ممنون از شما. پاینده باشید.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۹/۰۲ ۸:۱۰

- درخواست‌های مرتبط با فایل‌های استاتیک که عموما به صورت نسبی مقدار دهی می‌شوند، صرفا بر اساس آدرس جاری سایت توسط مرورگرها تفسیر و صورت می‌گیرند. اگر آدرس جاری سایت www.site.com است، برای نمونه درخواست اسکریپت آن www.site.com/script.js خواهد بود و نه حالت دیگری. بنابراین نیازی به پردازش مسیرهای نسبی فایل‌های غیرپویا توسط فیلتر فوق نیست. کل کاری که فیلتر ذکر شده انجام می‌دهد [Redirect کلی](#) کاربر است به مسیر دیگری و نه UrlRewrite تمام آدرس‌های سایت.
- روش فوق با تمام نگارش‌های مختلف IISهای موجود با حداقل دسترسی و حداقل توان پردازشی کار می‌کند. در اینجا تنها کاربر یکبار به مسیری دیگر هدایت خواهد شد و نیازی نیست تا تمام آدرس‌های سایت بازنویسی شوند.
- در مثال فوق [Redirect Result](#) استفاده شده از نوع Permanent است (آرگومان دوم آن true است).

خلاصه فیلتر MandatoryWww فوق، اجرای [سطر زیر است](#) ، بدون نیاز به تکرار آن در تمام صفحات و کنترلرهای سایت:

```
Response.RedirectPermanent
```

نویسنده: Javad Amiry
تاریخ: ۱۳۹۱/۰۹/۰۲ ۱۲:۵۶

- درباره فایل‌های استاتیک که داخل پرائتز توضیح دادم که حالت کلی رو در نظر گرفتیم. وگرنه اگه بخوایم همه جزئیات و حالت‌ها رو در نظر بگیریم خودش شاید یه مقاله طولانی و مفصل بشه (اگه فرصتی پیدا کنم سعی میکنم درباره ش بنویسم). در مورد UrlRewrite هم دقیقا همین کاری که شما با فیلتر فوق انجام میدید، در اون حالت با UrlRewrite انجام میدیم. متوجه منظورتون نشدم که تفاوت نتیجه‌ها کجاست؟
- در مورد IIS حق با شماست. خوب من اینطور فرض کردم که کسی که برنامه ای با ASP.NET MVC 3 مینویسه، حداقل به یه سرور با IIS 7 دسترسی داره. ولی در حالت کلی گفته شما صحیحه. خوب در مورد HttpModule و UrlRewrite هم کاربر تنها یک بار ریدایرکت میشه. واقعا متوجه نشدم که منظورتون از «تمام آدرس‌های سایت بازنویسی شوند» چیه.
- درسته متوجهم.
چرا کاملا درسته. در کنار این روش بنده هم دو تا روش دیگه (که به نظر شخصی خودم ساده‌تر و کاراتر هستن) رو معرفی کردم. هر سه تاشون عملا به نتیجه دارن، با تفاوت تو روش رسیدن به نتیجه.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۹/۰۲ ۱۳:۱۶

- در حالت کلی هیچ نیازی به ریدایرکشن فایل‌های استاتیک نیست که در مطلب جاری مشکلی به شمار روند. نه تأثیری در اعتبار سنجی دارند و نه به صورت پیش فرض وارد موتور اجرایی ASP.NET می‌شوند. به هر آدرسی که اشاره می‌کنند تا زمانیکه قابل یافت شدن در سایت باشند، جهت نمایش صحیح سایت کافی هستند؛ حتی اگر به یک آدرس خارج از سایت مانند مخزن کدی در کدپلکس اشاره کنند (مانند خیلی از سایت‌ها و وبلاگ‌ها که فایل‌های تصاویر، js، css و غیره خود را در سایت‌های ثالث هاست می‌کنند).

- امکان اجرای یک سایت MVC4 با IIS6 [وجود دارد](#) و نیاز به یک دست کردن آدرس‌های ورود به سایت جهت رفع سوء تفاهم توضیح داده شده در مقدمه مطلب برآورده خواهد شد.

نویسنده: Javad Amiry
تاریخ: ۱۳۹۱/۰۹/۰۲ ۱۴:۴۴

- بله حق با شماست. اگه به نمونه کارهای بنده هم توجه بفرمایید، همه فایل‌های استاتیک رو روی یه زیردامنه ای به اسم cdn.site.com یا static.site.com هاست کردم. سعی میکنم منظورمو توی یه مطلب جداگانه ای واضح شرح بدم. فکر میکنم برای اینکه کامنتو کوتاه کنم، خیلی بد نوشتم و نتونستم منظورمو برسونم.

- باز هم حق با شماست. بنده هم عرض نکردم همیشه هاست کرد. خود من هم قبلا اینکارو انجام دادم (روی سرور برتینا؛ اردیبهشت 90). ببینید، منظورم اینه که مثلا امروز اکثرمون Win 7 یا 8 رو سیستممون داریم یا بیشترمون مثلا VS 2012 نصب کردیم. کلا دارم عرض میکنم که همیشه گرایشمون به آپدیت بودن هست. با توجه به هزینه هاستینگ که واقعا همینطور داره میاد پایین و شرکتهایی که شدیداً با هم در رقابت هستن و گرایش‌های درونی خود ما برنامه نویس ها، فکر میکنم کمتر کسی پیدا بشه که امروز تمایل داشته باشه کلا هیچ سایتی رو (حالا MVC بودن یا نبودنش یه طرف) روی IIS 6 هاست بکنه. چند وقت پیش یادمه یه مطلبی نوشته بودید درباره چک لیست برنامه‌های MVC و اون تو یه جمله ای که خیلی خوشم اومد و تو ذهنم حک شد این مضمون بود که IE 6 و 7 رو به رحمت ایزدی بییوندانیم D: من منظورم دقیقا در همین راستاست. بعنوان نمونه من دقیقا 2 روز پیش یه هاستی رو ثبت کردم که روی ویندوز سرور 2012 با SQLServer 2012 و دات نت 4.5 و IIS 8 با 1 گیگ فضا و پهنای باند نامحدود و 300 مگ رم، هزینه یک ساله ش با یه دامنه com شد فقط 60 دلار. دسترسی ریموت به IIS و SQLServer هم میده و از Web Deploy هم پشتیبانی میکنه. خوب همین پکیج خیلی ارزونتر هم گیر میومد. ولی این شرکت معتبرتر بود، گرونتر میداد که گرونش اینقدر شد. نمیدونم تونستم منظورمو برسونم یا نه.

- با اجازتون یه مطلب کوتاهی در راستای پست شما و استفاده از UrlRewrite توی این [آدرس](#) نوشتم و مثال مورد نظرمو اونجا ذکر کردم. در مورد فایل‌های استاتیک و سایر موارد هم حتما بیشتر توضیح میدم.

با احترام فراوان. پاینده باشید.

نویسنده: صابر فتح الهی
تاریخ: ۱۳۹۲/۰۵/۱۱ ۲:۳۲

با سلام؛ من روش شما و آقای امیری بررسی کردم هردو جواب میده اما ایرادی که به روش آقای امیری وارد است اینه که در صورتی که ما چند دامنه داشته باشیم همه آنها بروی یک ادرس ریدایرکت میشود.

نویسنده: هادی دانش پژوه
تاریخ: ۱۳۹۳/۰۲/۰۸ ۱۷:۵۰

سلام و با تشکر. می‌خواستم بدونم چجوری میتونم پسوند آدرس Url رو حذف کنم، نمی‌دونم تا چه اندازه برای کیفیت سایت مفید هست ولی بیشتر سایت‌های که من می‌بینم این کار رو انجام می‌دنند. من با asp.net form کار می‌کنم.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۲/۰۸ ۱۸:۲۵

» [مروری بر قابلیت جدید ASP.NET FriendlyUrls](#) «

« [ASP.Net WebForms در Url Routing](#) »

« [Intelligencia.UrlRewriter](#) توسط [Intelligencia.UrlRewriter](#) »

نویسنده: بهاره فیضی
تاریخ: ۱۳۹۳/۰۷/۰۳ ۱۲:۸

سلام؛ همین کاری که فرمودید در اینجا رو چطور میشه در asp.net معمولی انجام داد؟ با توجه به اینکه IIS هم 6 هست. ممنون.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۷/۰۳ ۱۳:۶

- نسخه‌ی بهبود یافته مطلب جاری [در اینجا](#) ارسال شده «فیلتر نهایی اصلاح شده: [ForceWww.cs](#)»

- تمام مطالب آن به سادگی توسط یک HTTP Module برای وب فرم‌ها قابل بازنویسی است. برای مثال: [ForceWwwModule.cs](#)