

در مقاله « [استفاده از Razor در فایل‌های CSS و JavaScript](#) » با نحوه‌ی استفاده از Razor در فایل‌های Js و Css آشنا شدید. در مقاله‌ی جاری با روش دیگری، با نحوه‌ی استفاده از Syntax Razor در فایل‌های Css آشنا خواهید شد.

در ابتدا بعد از ایجاد یک پروژه‌ی جدید، نیاز دارید تا اسمبلی [RazorEngine](#) را توسط [Package Manager Console](#) به پروژه اضافه نمایید.

```
Install-Package RazorEngine -Version 3.7.0
```

در گام بعدی نیاز است در کنترلری، یک اکشن متد را تعریف نماید که خروجی آن از نوع رشته خواهد بود و دستورات زیر در آن تعریف می‌شوند:

```
using System.Web.Mvc;
using RazorEngine;

namespace dynamicCSS.Controllers
{
    public class StyleController : Controller
    {
        /// <summary>
        /// نام متد ارجاعی به فایل سی اس اس
        /// </summary>
        /// <returns></returns>
        public string Index()
        {
            //The ContentType property specifies the HTTP content type for the response. If no
            //ContentType is specified, the default is text/HTML.
            Response.ContentType = "text/css";
            //استفاده از متد
            //ReadAllText
            //فایل رو خوانده و سپس از متد
            //Parse in Razor Class
            //به صورت رشته برگشت خواهیم داد
            return Razor.Parse(System.IO.File.ReadAllText(Server.MapPath("/Content/Site.css")));
        }
    }
}
```

در خط 21، فایل Css موجود در پوشه‌ی Content واقع در ریشه‌ی پروژه، خوانده شده و با متد Parse در کلاس Razor پردازش و بازگشت داده می‌شود. در کد زیر تمامی متدهای موجود در کلاس Razor را می‌توانید ملاحظه کنید:

```
#region Assembly RazorEngine.dll, v2.1.4039.23635
// Your Address\dynamicCSS\packages\RazorEngine.2.1\lib\NetFramework 4.0\RazorEngine.dll
#endregion

using RazorEngine.Templating;
using System;
using System.Collections.Generic;

namespace RazorEngine
{
    public static class Razor
    {
        public static TemplateService DefaultTemplateService { get; }
        public static IDictionary<string, TemplateService> Services { get; }

        public static void AddResolver(Func<string, string> resolverDelegate);
        public static void AddResolver(ITemplateResolver resolver);
        public static void Compile(string template, string name);
        public static void Compile(string template, Type modelType, string name);
        public static void CompileWithAnonymous(string template, string name);
    }
}
```

```
public static string Parse(string template, string name = null);
public static string Parse<T>(string template, T model, string name = null);
public static string Run(string name);
public static string Run<T>(T model, string name);
public static void SetActivator(Func<Type, ITemplate> activator);
public static void SetActivator(IAActivator activator);
public static void SetTemplateBase(Type type);
}
}
```

در این حالت می‌توان از دستورات Razor در فایل Css نیز استفاده کرد:

```
@{
    // در اینجا دو متغیر با کلمه کلیدی
    // var
    // ساخته و به صورت پیش فرض مقدار دهی نمودیم
    var redColor = "red";
    var sizeMode = "100px";
}

h1 {
    // روش استفاده از متغیرها
    color: @redColor !important;
    font-size : @sizeModel !important;
}
```

و در انتها می‌بایست در Layout پروژه، آدرس فایل Css را مشخص کرد:

```
// تغییر آدرس فایل به اکشن متد در کنترلر
//Home
//<link href="/Content/Site.Css" rel="stylesheet" />
// شکل صحیح آدرس دهی
<link href="@Url.Action("Style", "Home")" rel="stylesheet" />
```

نکته: در صورتیکه متغیری بعد از دستورات استفاده شده تعریف گردد، با خطای زیر روبرو خواهید شد:

Server Error in '/' Application.

Unable to compile template. Cannot use local variable 'redColor' before it is declared

Other compilation errors may have occurred. Check the Errors property for more information.

Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.

Exception Details: RazorEngine.Templating.TemplateCompilationException: Unable to compile template. Cannot use local variable 'redColor' before it is declared

Other compilation errors may have occurred. Check the Errors property for more information.

Source Error:

```
Line 32:      {
Line 33:          Response.ContentType = "text/css";
Line 34:          return Razor.Parse(System.IO.File.ReadAllText(Server.MapPath("/Content/Site.css")));
Line 35:      }
Line 36:  }
```

Source File: c:\Users\Omid Nasr\Documents\Visual Studio 2013\Projects\Css\Css\Controllers\HomeController.cs **Line:** 34

Stack Trace:

```
[TemplateCompilationException: Unable to compile template. Cannot use local variable 'redColor' before it is declared]
Other compilation errors may have occurred. Check the Errors property for more information.]
RazorEngine.Compilation.DirectCompilerServiceBase.CompileType(TypeContext context) in c:\Users\abbottm\Documents\GitHub\RazorEngine\src\Core\RazorEngine.
```

در خروجی نهایی تگ h1 با فونت **100** پیکسل و رنگ **قرمز** به نمایش در می‌آید:

در صورتیکه خروجی نهایی به شکل صحیح اجرا نگردید، برای تست صحیح بودن گام‌های قبلی می‌توانید اکشن متد را در مرورگر اجرا کنید

localhost:1599/Home/Style