

حذف تمامی تگ‌های یک عبارت HTML

این تابع و عبارت باقاعده به کار رفته در آن هنگام جستجو بر روی یک فایل html که حاوی انبوهی از تگ‌ها است می‌تواند مفید باشد و یا جهت حذف هر نوع فرمت اعمالی به یک متن.

```
private static readonly Regex _htmlRegex = new Regex("<.*?>", RegexOptions.Compiled);  
/// <summary>  
/// حذف تمامی تگ‌های موجود  
/// </summary>  
/// <param name="html">ورودی اچ تی ام ال</param>  
/// <returns></returns>  
public static string CleanTags(string html)  
{  
    return _htmlRegex.Replace(html, string.Empty);  
}
```

حذف یک تگ ویژه بدون حذف محتویات آن

فرض کنید می‌خواهید تمام تگ‌های script بکار رفته در یک محتوای html را حذف کنید.

```
private static readonly Regex _contentRegex = new Regex(@"<\/?script[^\>]*?>", RegexOptions.Compiled | RegexOptions.IgnoreCase);  
/// <summary>  
/// تنها حذف یک تگ ویژه  
/// </summary>  
/// <param name="html">ورودی اچ تی ام ال</param>  
/// <returns></returns>  
public static string CleanScriptTags(string html)  
{  
    return _contentRegex.Replace(html, string.Empty);  
}
```

حذف یک تگ خاص به همراه محتویات آن

فرض کنید می‌خواهیم در محتوای html دریافتی اثری از تگ‌ها و کدهای جاوا اسکریپتی یافت نشود.

```
private static readonly Regex _safeStrRegex = new Regex(@"<script[^\>]*?>[\s\S]*?</script>", RegexOptions.Compiled | RegexOptions.IgnoreCase);  
/// <summary>  
/// حذف یک تگ ویژه به همراه محتویات آن  
/// </summary>  
/// <param name="html">ورودی اچ تی ام ال</param>  
/// <returns></returns>  
public static string CleanScriptsTagsAndContents(string html)  
{  
    return _safeStrRegex.Replace(html, "");  
}
```

و اگر فرض کنیم که متدهای فوق در کلاسی به نام CRegexHelper قرار گرفته‌اند، کلاس آزمون واحد آن به صورت زیر می‌تواند باشد:

```
using NUnit.Framework;
```

```
namespace testWinForms87
{
    [TestFixture]
    public class CTestRegexHelper
    {
        #region Methods (3)
        // Public Methods (3)

        [Test]
        public void TestCleanScriptsTagsAndContents()
        {
            Assert.AreEqual(
                CRegexHelper.CleanScriptsTagsAndContents("data1 <script> ... </script> data2"),
                "data1 data2");
        }

        [Test]
        public void TestCleanScriptTags()
        {
            Assert.AreEqual(
                CRegexHelper.CleanScriptTags("<b>data1</b> <script> ... </script> data2"),
                "<b>data1</b> ... data2");
        }

        [Test]
        public void TestCleanTags()
        {
            Assert.AreEqual(
                CRegexHelper.CleanTags("<b>data</b>"),
                "data");
        }

        #endregion Methods
    }
}
```

