

همانطوری که میدونید نسخه [MVC 4 RC](#) در دسترس قرار گرفته و خالی از لطف نیست که یک بررسی درباره امکانات جدیدش انجام بشه. ابتدا سعی میکنم یک لیست کلی از امکانات این تکنولوژی داشته باشیم و بعد نگاهی هم به Razor و تغییراتش خواهیم داشت.

ASP.NET Web API

Refreshed and modernized default project templates

New mobile project template

Many new features to support mobile apps

Recipes to customize code generation

Enhanced support for asynchronous methods

لیست فوق شامل برترین ویژگی‌های این نسخه است که در پستهای آینده هر کدام از این‌ها بررسی خواهند شد.

تغییرات Razor

Razor از نسخه MVC4 Beta شاهد تغییرات و بهبودهایی بود که این تغییرات بنیادی و رادیکالی نبودند و فقط در جهت بهبود حس کاربری آن صورت گرفت. این حس از آن جهت است که شما نیاز به نوشتن کد کمتری دارید.

استفاده نکردن از @Url.Content

در نسخه قبلی از امکان ذکر شده برای مشخص کردن مسیرهای CSS و فایل‌های JS هم استفاده میشد حالا بجای استفاده از آن میشود:

در نسخه‌های قبلی:

```
<script src="@Url.Content("~/Scripts/Site.js")"></script>
```

در نسخه 4:

```
<script src("~/Scripts/Site.js")></script>
```

زمانی که Razor حرف را ~/ تشخیص می‌دهد خروجی یکسانی با حالت قبلی برای ما درست می‌کند.

شرط‌ها (Conditions)

در نسخه قبلی برای استفاده از attribute ها که ممکن بود Null باشند مجبور به چک کردن آنها بودیم:

```
<div @{if (myClass != null) { <text>class="@myClass"</text> } }>Content</div>
```

اما حالا با خیال راحت می‌توان نوشت:

```
<div class="@myClass">Content</div>
```

که اگر attribute ما null باشد به صورت اتوماتیک تشخیص داده میشود و کد زیر رندر میشود

```
<div>Content</div>
```

در ضمناً کد بالا فقط مربوط به چک کردن Nullable نیست و از آن می‌توان در Boolean ها هم استفاده کرد.

```
<input checked="@ViewBag.Checked" type="checkbox"/>
```

که اگر مقدار True نباشد:

```
<input type="checkbox"/>
```

بطور خلاصه همیشه گفت MVC4 تغییراتش نسبت به نسخه قبلی تو خیلی از زمینه‌ها مربوط میشه بهبود ابزارهای موجود در کل کار با این ابزار بسیار برای من لذت بخشه.
ادامه دارد...