آموزش MVC – مقایسه ی View های با نوع مشخص شده (Strongly-typed views) و View هایی با نوع نامشخص در زمان اجرا (dynamic views) در اپلیکیشن MVC

سه روش برای ارسال اطلاعات از Controller (کلاس های #C) به view (فایل میزبان HTML و CSS) در ASP.NET MVC 3 وجود دارد:

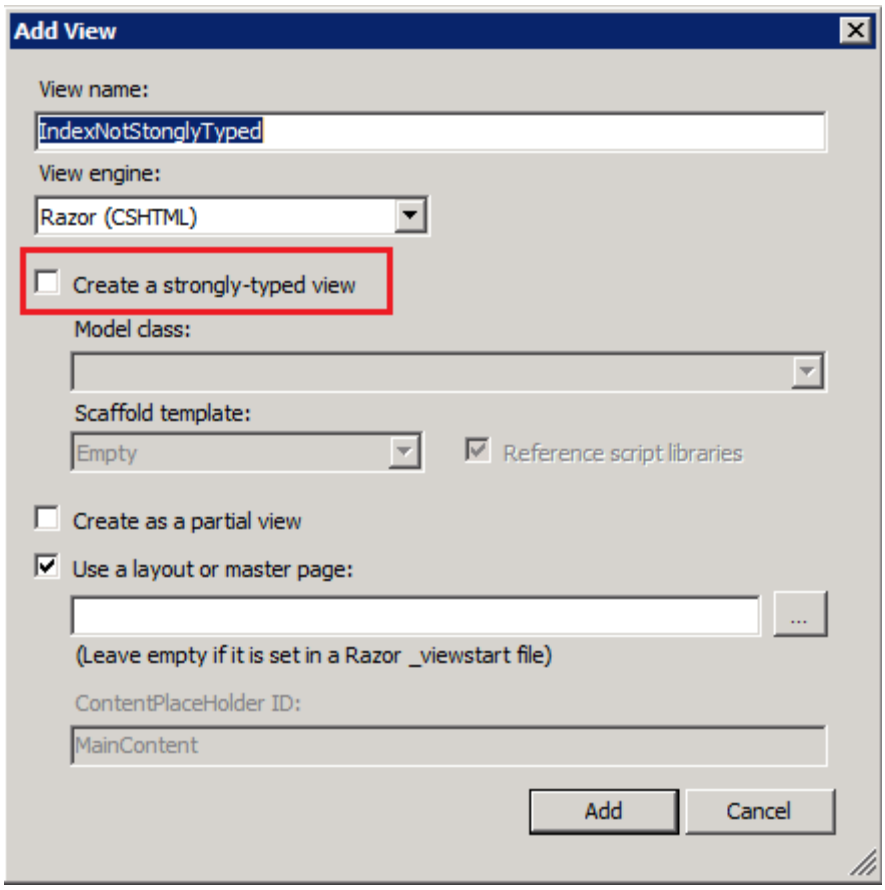
1. به عنوان یک model object (کلاس های حاوی اطلاعات ساخته شده از روی جداول دیتابیس) که شدیدا وبسته به نوع هستند و نوع آن ها قبلا مشخص شده/strongly typed model object .
2. به صورت یک نوع پویا و نامشخص در زمان اجرا (dynamic type) با استفاده از دستور @model
3. استفاده از ViewBag (یک آبجکت با نوع نامشخص در زمان اجرا/داینامیک که برای ارسال داده از کلاس کنترلر به view استفاده می شود)

در این آموزش یک اپلیکیشن ساده مبتنی بر mvc که سه وبلاگ معروف را لیست می کند، جهت مقایسه و شرح تفاوت view های strongly-typed و dynamic پیاده سازی می کنیم. Controller مربوطه با یک لیست ساده از وب لاگ های برنامه نویسان مشهور بدین صورت آغاز می شود:

```
using System.Collections.Generic;
using System.Web.Mvc;
namespace Mvc3ViewDemo.Controllers {
  public class Blog {
    public string Name;
    public string URL;
  }
  public class HomeController : Controller {
    List<Blog> topBlogs = new List<Blog>
   {
     new Blog { Name = "ScottGu", URL = "http://weblogs.asp.net/scottgu/"},
     new Blog { Name = "Scott Hanselman", URL = "http://www.hanselman.com/blog/"},
     new Blog { Name = "Jon Galloway", URL = "http://www.asp.net/mvc"}
   };
    public ActionResult IndexNotStonglyTyped() {
      return View(topBlogs);
    }
    public ActionResult About() {
      ViewBag.Message = "Welcome to ASP.NET MVC!";
      return View();
```

```
    }
  }
}
```

بر روی متد IndexNotStonglyTyped() راست کلیک کرده و یک view مبتنی بر Razor (view خروجی گرامر Razor) اضافه کنید.



لازم است تیک چک باکس Create a strongly-typed view را بردارید. view خروجی در ابتدا محتوای چندانی ندارد:

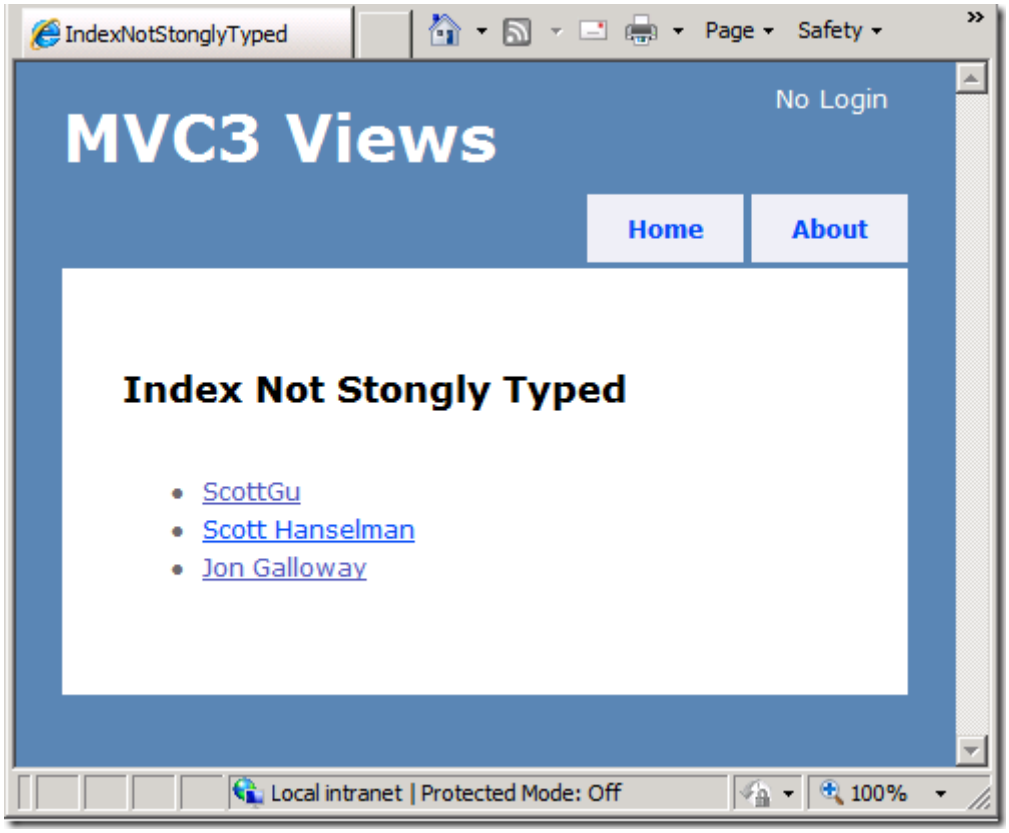
```
@{
  ViewBag.Title = "IndexNotStonglyTyped";
}
<h2>IndexNotStonglyTyped</h2>
```

در اولین خط از فایل Views\Home\IndexNotStonglyTyped.cshtml، دستور model@ و کلیدواژه ی dynamic را درج نمایید.

```
@model dynamic
```

از آنجایی که از کلیدواژه ی dynamic استفاده کرده و نوع view خود را از قبل مشخص نکرده اید (strongly-typed view تعریف نشده)، ابزار intellisense از محیط کاری Visual هیچ کمکی به شما نمی کند. نسخه ی کامل کد در زیر نمایش داده شده است:

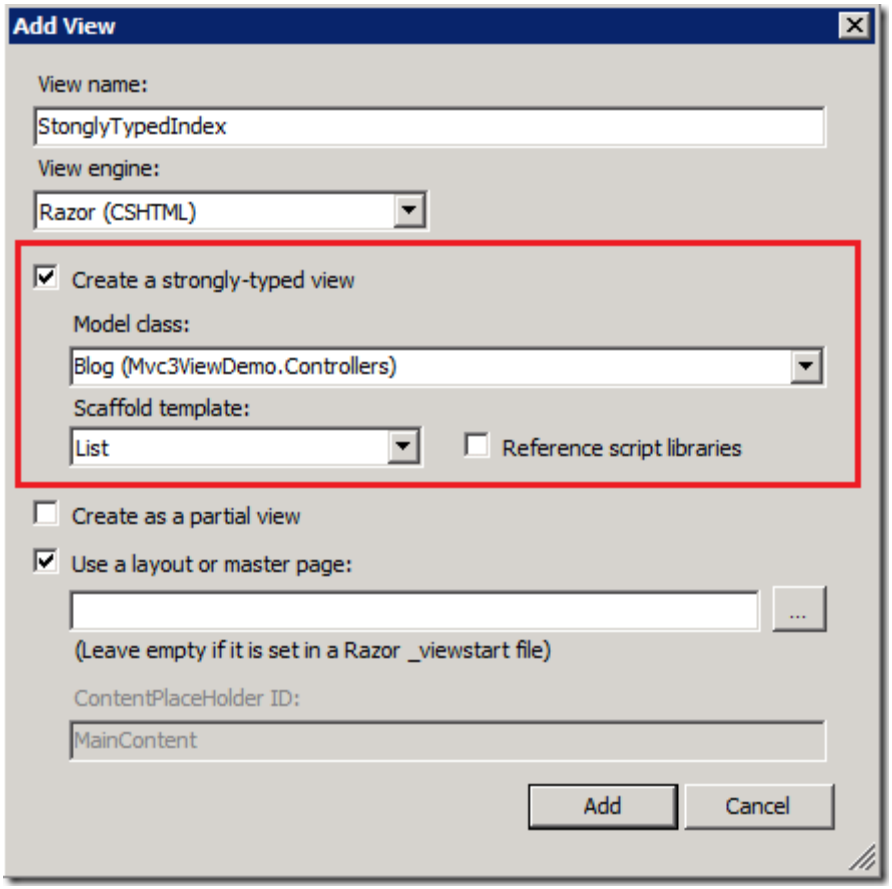
```
@model dynamic
@{
  ViewBag.Title = "IndexNotStonglyTyped";
}
<h2>Index Not Stongly Typed</h2>
<p>
 <ul>
@foreach (var blog in Model) {
  <li>
   <a href="@blog.URL">@blog.Name</a>
  </li>
}
 </ul>
</p>
```



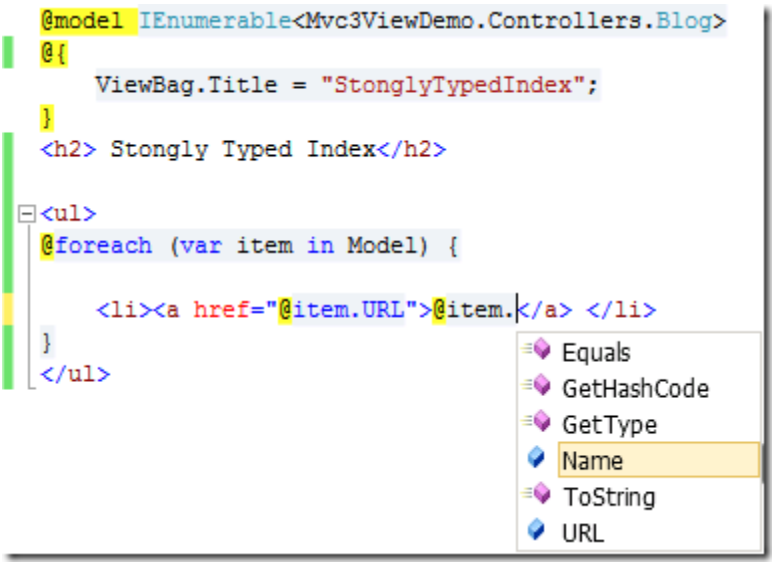
اکنون می خواهیم یک view که نوع آن قبلا مشخص شده/شدیدا به نوع وابسته است (Strongly-typed) را به اپلیکیشن MVC خود اضافه کنیم. برای این منظور کد زیر را به کنترلر (کلاس C#) اضافه می کنیم:

```
public ActionResult StonglyTypedIndex() {
  return View(topBlogs);
}
```

همان طور که می بینید تابع ;(topBlogs)View return که در view قبلی (dynamic) فراخوانی شد، در نمونه ی حاضر نیز صدا زده می شود. داخل تابع ()StonglyTypedIndex راست کلیک کرده، گزینه ی Add View را انتخاب نمایید. این بار از منوی کشویی Model class، کلاس Blog و از منوی Scaffold template نیز آیتم List را انتخاب نمایید.



داخل قالب آماده ی view جدید (view template)، بر خلاف نمونه ی قبلی می توانیم از ابزار Intellisense استفاده کنیم.


@model IEnumerable<Mvc3ViewDemo.Controllers.Blog>
@{
ViewBag.Title = "StonglyTypedIndex";
}
<h2> Stongly Typed Index</h2>

@foreach (var item in Model) {
@item.
}

Equals
GetHashCode
GetType
Name
ToString
URL