عنوان: RadioButtonList در ASP.NET MVC
نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۲/۱۲ ۰۹:۲۷:۰۰
آدرس: www.dotnettips.info
برچسب‌ها: MVC

برای تهیه یک RadioButtonList نیز می‌توان از همان نکته‌ی CheckBoxList استفاده کرد: نام عناصر radio button اضافه شده به صفحه را یکسان وارد می‌کنیم. به این ترتیب یک گروه تشکیل خواهد شد و زمانیکه اطلاعات این عناصر به سرور ارسال می‌شود، اینبار بجای یک آرایه، تنها مقدار کنترل انتخاب شده، ارسال می‌گردد. یک مثال:
یک پروژه جدید و خالی ASP.NET MVC را آغاز کنید. سپس کنترلر Home و View خالی Index را نیز ایجاد نمائید. محتویات این دو را به نحو زیر تغییر دهید:

```
@{
    ViewBag.Title = "Index";
}
<h2>
    Index</h2>
<fieldset>
    <legend>HandleForm1 (Normal)</legend>
    @using (Html.BeginForm(actionName: "HandleForm1", controllerName: "Home"))
    {
        @:your favorite tech: <br />
        @Html.RadioButton(name: "tech", value: ".NET", isChecked: true) @:DOTNET <br />
        @Html.RadioButton(name: "tech", value: "JAVA", isChecked: false) @:JAVA <br />
        @Html.RadioButton(name: "tech", value: "PHP", isChecked: false) @:PHP <br />
        <input type="submit" value="Submit" />
    }
</fieldset>
```

```
using System.Collections.Generic;
using System.Web.Mvc;

namespace MvcApplication23.Controllers
{
    public class HomeController : Controller
    {
        [HttpGet]
        public ActionResult Index()
        {
            return View();
        }

        [HttpPost]
        public ActionResult HandleForm1(string tech)
        {
            return RedirectToAction("Index");
        }
    }
}
```

در اینجا سه RadioButton با نامی یکسان در صفحه اضافه شده‌اند. سپس داخل متد HandleForm1 یک breakpoint قرار دهید. اکنون برنامه را اجرا کنید و فرم را به سرور ارسال نمائید. پارامتر tech با value عنصر انتخابی مقدار دهی خواهد شد.

تهیه یک RadioButtonList عمومی

اطلاعات فوق را می‌توان تبدیل به یک HtmlHelper با قابلیت استفاده مجدد نیز نمود:

```
@helper RadioButtonList(string groupName, IEnumerable<System.Web.Mvc.SelectListItem> items)
{
    <div class="RadioButtonList">
        @foreach (var item in items)
        {
            @item.Text
            <input type="radio" name="@groupName"
                   value="@item.Value"
                   @if (item.Selected) { <text>checked="checked"</text> }
               />
            <br />
        }
    </div>
}
```

برای مثال یک فایل را در مسیر app_code\Helpers.cshtml ایجاد کرده و اطلاعات فوق را به آن اضافه نمائید.
اینبار برای استفاده از آن خواهیم داشت:

```
using System.Collections.Generic;
using System.Web.Mvc;

namespace MvcApplication23.Controllers
{
    public class HomeController : Controller
    {
        [HttpGet]
        public ActionResult Index()
        {
            ViewBag.Tags = new[]
                           {
                               new SelectListItem { Text = ".NET", Value = "Val1", Selected = true },
                               new SelectListItem { Text = "JAVA", Value = "Val2", Selected = false },
                               new SelectListItem { Text = "PHP", Value = "Val3", Selected = false }
                           };
            return View();
        }

        [HttpPost]
        public ActionResult HandleForm2(string preferredTechnology)
        {
            return RedirectToAction("Index");
        }
    }
}
```

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

<fieldset>
    <legend>HandleForm2 (Helper)</legend>
    @using (Html.BeginForm(actionName: "HandleForm2", controllerName: "Home"))
    {
        @:your favorite tech: <br />
        @Helpers.RadioButtonList("preferredTechnology", (SelectListItem[])ViewBag.Tags)
        <input type="submit" value="Submit" />
    }
</fieldset>
```

متد سفارشی تهیه شده، یک آرایه از SelectListItem ها را دریافت کرده و به صورت خودکار تبدیل به RadioButtonList می‌کند. بر اساس نام آن می‌توان به مقدار انتخاب شده ارسالی به سرور در کنترلر مرتبط، دسترسی یافت.

تهیه یک Templated helper سفارشی

در عمل زمانیکه با مدل‌ها کار می‌کنیم و اطلاعات برنامه قرار است Strongly typed باشند، مرسوم است لیستی از انتخاب‌ها را به صورت یک enum تعریف کنند. برای مثال مدل زیر را به برنامه اضافه کنید:

```
using System.ComponentModel.DataAnnotations;

namespace MvcApplication23.Models
{
    public enum Gender
    {
        [Display(Name = "مرد")]
        Male,
        [Display(Name = "زن")]
        Female,
    }

    public class User
    {
        [ScaffoldColumn(false)]
        public int Id { set; get; }

        [Display(Name = "نام")]
        public string Name { set; get; }

        [Display(Name = "جنسیت")]
        [UIHint("EnumRadioButtonList")]
        public Gender Gender { set; get; }
    }
}
```

قصد داریم یک Templated helper سفارشی را به نام EnumRadioButtonList، ایجاد کنیم تا در زمان فراخوانی متد Html.EditorForModel، به صورت خودکار enum تعریف شده را به صورت یک RadioButtonList نمایش دهد.
برای این منظور فایل جدید Views\Shared\EditorTemplates\EnumRadioButtonList.cshtml را به برنامه اضافه کنید. محتوای آن‌را به نحو زیر تغییر دهید:

```
@using System.ComponentModel.DataAnnotations
@using System.Globalization
@model Enum
@{
    Func<Enum, string> getDescription = enumItem =>
    {
        var type = enumItem.GetType();
        var memInfo = type.GetMember(enumItem.ToString());
        if (memInfo != null && memInfo.Any())
        {
            var attrs = memInfo[0].GetCustomAttributes(typeof(DisplayAttribute), false);
            if (attrs != null && attrs.Any())
                return ((DisplayAttribute)attrs[0]).GetName();
        }
        return enumItem.ToString();
    };

    var listItems = Enum.GetValues(Model.GetType())
                        .OfType<Enum>()
                        .Select(enumItem =>
                                    new SelectListItem()
                                    {
                                        Text = getDescription(enumItem),
                                        Value = enumItem.ToString(),
                                        Selected = enumItem.Equals(Model)
                                    });

    string prefix = ViewData.TemplateInfo.HtmlFieldPrefix;
    ViewData.TemplateInfo.HtmlFieldPrefix = string.Empty;

    int index = 0;
    foreach (var li in listItems)
```

```
    {
        string fieldName = string.Format(CultureInfo.InvariantCulture, "{0}_{1}", prefix, index++);
    <div class="editor-radio">
        @Html.RadioButton(prefix, li.Value, li.Selected, new { @id = fieldName })
        @Html.Label(fieldName, li.Text)
    </div>
    }

    ViewData.TemplateInfo.HtmlFieldPrefix = prefix;
}
```

در اینجا به کمک Reflection به اطلاعات enum دریافتی دسترسی خواهیم داشت. بر این اساس می‌توان نام عناصر آن‌را یافت و تبدیل به یک RadioButtonList کرد. البته کار به همینجا ختم نمی‌شود. در این بین باید دقت داشت که ممکن است از ویژگی Display (مانند مدل نمونه فوق) بر روی تک تک عناصر یک enum نیز استفاده شود. به همین جهت این مورد نیز باید پردازش گردد.

نهایتا برای استفاده از این Templated helper سفارشی، کنترلر و View برنامه را به نحو زیر می‌توان تغییر داد:

```
using System.Collections.Generic;
using System.Web.Mvc;
using MvcApplication23.Models;

namespace MvcApplication23.Controllers
{
    public class HomeController : Controller
    {
        [HttpGet]
        public ActionResult Index()
        {
            var user = new User { Id = 1, Name = "name 1", Gender = Gender.Male };
            return View(user);
        }

        [HttpPost]
        public ActionResult HandleForm3(User user)
        {
            return RedirectToAction("Index");
        }
    }
}
```

```
@model MvcApplication23.Models.User
@{
    ViewBag.Title = "Index";
}
<h2>
    Index</h2>
<fieldset>
    <legend>HandleForm3 (EditorForModel)</legend>
    @using (Html.BeginForm(actionName: "HandleForm3", controllerName: "Home"))
    {
        @Html.EditorForModel()
        <input type="submit" value="Submit" />
    }
</fieldset>
```

برای استفاده از یک templated helper سفارشی چندین روش وجود دارد:

الف) همانند مثال فوق از ویژگی UIHint استفاده شود.

ب) نام فایل را به enum.cshtml تغییر دهیم. به این ترتیب از این پس کلیه enumها در صورت استفاده از متد Html.EditorForModel، به صورت خودکار تبدیل به یک RadioButtonList می‌شوند.

ج) متد زیر نیز همین کار را انجام می‌دهد:

```
@Html.EditorFor(model => model.EnumProperty, "EnumRadioButtonList")
```

نظرات خوانندگان

نویسنده: RezaBoojari
تاریخ: ۰۹:۵۲:۵۰ ۱۳۹۱/۰۲/۱۲

سلام معلم اینترنتی عزیزم.
این روز رو واقعا از صمیم قلب تبریک میگم.