

اصول نمایش اطلاعات مودال به کمک bootstrap در مطلب « [استفاده از modal dialogs مجموعه Twitter Bootstrap برای گرفتن تأیید از کاربر](#) » بررسی شدند.
 در این قسمت قصد داریم یک فرم Ajax ایی را در ASP.NET MVC به همراه تمام مسایل اعتبارسنجی، پردازش اطلاعات و غیره را به کمک Twitter Bootstrap و jQuery Ajax پیاده سازی کنیم.

تهیه افزونه jquery.bootstrap-modal-ajax-form.js

از این جهت که مباحث مرتبط با نمایش و پردازش فرم‌های مودال Ajax ایی به کمک Twitter Bootstrap اندکی نکته دار و طولانی هستند، بهتر است این موارد را به شکل یک افزونه، کپسوله کنیم. کدهای کامل افزونه jquery.bootstrap-modal-ajax-form.js را ادامه ملاحظه می‌کنید:

```
// 
(function ($) {
    $.bootstrapModalAjaxForm = function (options) {
        var defaults = {
            renderModalPartialViewUrl: null,
            renderModalPartialViewData: null,
            postUrl: '/',
            loginUrl: '/login',
            beforePostHandler: null,
            completeHandler: null,
            errorHandler: null
        };
        var options = $.extend(defaults, options);

        var validateForm = function (form) {
            // فعال سازی دستی اعتبار سنجی جی کوئری
            var val = form.validate();
            val.form();
            return val.valid();
        };

        var enableBootstrapStyleValidation = function () {
            $.validator.setDefaults({
                highlight: function (element, errorClass, validClass) {
                    if (element.type === 'radio') {
                        this.findByName(element.name).addClass(errorClass).removeClass(validClass);
                    } else {
                        $(element).addClass(errorClass).removeClass(validClass);
                        $(element).closest('.control-group').removeClass('success').addClass('error');
                    }
                    $(element).trigger('highlighted');
                },
                unhighlight: function (element, errorClass, validClass) {
                    if (element.type === 'radio') {
                        this.findByName(element.name).removeClass(errorClass).addClass(validClass);
                    } else {
                        $(element).removeClass(errorClass).addClass(validClass);
                        $(element).closest('.control-group').removeClass('error').addClass('success');
                    }
                    $(element).trigger('unhighlighted');
                }
            });
        };

        var enablePostBackValidation = function () {
            $('form').each(function () {
                $(this).find('div.control-group').each(function () {
                    if ($(this).find('span.field-validation-error').length &gt; 0) {
                        $(this).addClass('error');
                    }
                });
            });
        };
    };
}
// ]&gt;</pre>
</div>
<div data-bbox="495 956 530 970" data-label="Page-Footer">۱/۱۴</div>
```

```

var processAjaxForm = function (dialog) {
    $('form', dialog).submit(function (e) {
        e.preventDefault();

        if (!validateForm($(this))) {
            // اگر فرم اعتبار سنجی نشده، اطلاعات آن ارسال نشود
            return false;
        }

        // در اینجا می‌توان مثلا دکمه‌ای را غیرفعال کرد
        if (options.beforePostHandler)
            options.beforePostHandler();

        // اطلاعات نباید کش شوند
        $.ajaxSetup({ cache: false });
        $.ajax({
            url: options.postUrl,
            type: "POST",
            data: $(this).serialize(),
            success: function (result) {
                if (result.success) {
                    $('#dialogDiv').modal('hide');
                    if (options.completeHandler)
                        options.completeHandler();
                } else {
                    $('#dialogContent').html(result);
                    if (options.errorHandler)
                        options.errorHandler();
                }
            }
        });
        return false;
    });
};

var mainContainer = "<div id='dialogDiv' class='modal hide fade in'><div
id='dialogContent'></div></div>";
enableBootstrapStyleValidation(); // اعمال نکات خاص بوت استرپ جهت اعتبارسنجی یکپارچه با آن
$.ajaxSetup({ cache: false });
$.ajax({
    type: "POST",
    url: options.renderModalPartialViewUrl,
    data: options.renderModalPartialViewData,
    contentType: "application/json; charset=utf-8",
    dataType: "json",
    complete: function (xhr, status) {
        var data = xhr.responseText;
        if (xhr.status == 403) {
            window.location = options.loginUrl; // در حالت لاگین نبودن شخص اجرا می‌شود
        }
        else if (status === 'error' || !data) {
            if (options.errorHandler)
                options.errorHandler();
        }
        else {
            var dialogContainer = "#dialogDiv";
            $(dialogContainer).remove();
            $(mainContainer).appendTo('body');

            $('#dialogContent').html(data); // دریافت پویای اطلاعات مودال دیالوگ
            $.validator.unobtrusive.parse("#dialogContent"); // فعال سازی اعتبارسنجی فرمی که با
            // ایجکس بارگذاری شده
            enablePostBackValidation();
            // و سپس نمایش آن به صورت مودال
            $('#dialogDiv').modal({
                backdrop: 'static', // با کلیک کاربر روی صفحه، صفحه مودال بسته نمی‌شود
                keyboard: true
            }, 'show');
            // تحت نظر قرار دادن این فرم اضافه شده
            processAjaxForm('#dialogContent');
        }
    }
});
})(jQuery);
// ]]>

```

- توابع `enablePostbackValidation` و `enableBootstrapStyleValidation` در مطلب « [اعمال کلاس‌های ویژه اعتبارسنجی](#) » [Twitter bootstrap به فرم‌های ASP.NET MVC](#) بررسی شدند.

- این افزونه با توجه به مقدار `renderModalPartialViewUrl`، یک `partial view` را از برنامه ASP.NET MVC درخواست می‌کند.
 - سپس این `partial view` را به صورت خودکار به صفحه اضافه کرده و آن‌را به صورت `modal` نمایش می‌دهد.
 - پس از افزودن فرم Ajax ایی دریافتی، مسایل اعتبارسنجی را به آن اعمال کرده و سپس دکمه `submit` آن‌را تحت کنترل قرار می‌دهد.

- در زمان `submit`، ابتدا بررسی می‌کند که آیا فرم معتبر است و اعتبارسنجی آن بدون مشکل است؟ اگر اینچنین است، اطلاعات فرم را به آدرس `postUrl` به صورت Ajax ایی ارسال می‌کند.

کدهای مدل برنامه

```
using System.ComponentModel;
using System.ComponentModel.DataAnnotations;

namespace Mvc4TwitterBootstrapTest.Models
{
    public class User
    {
        public int Id { set; get; }

        [DisplayName("نام")]
        [Required(ErrorMessage="تکمیل کنید")]
        public string Name { set; get; }

        [DisplayName("نام خانوادگی")]
        [Required(ErrorMessage = "تکمیل کنید")]
        public string LastName { set; get; }
    }
}
```

در اینجا یک مدل ساده‌را به همراه ویژگی‌های اعتبارسنجی و نام‌های نمایشی خواص ملاحظه می‌کنید.

کدهای کنترلر برنامه

```
using System.Web.Mvc;
using Mvc4TwitterBootstrapTest.Models;

namespace Mvc4TwitterBootstrapTest.Controllers
{
    public class ModalFormAjaxController : Controller
    {
        [HttpGet]
        public ActionResult Index()
        {
            return View(); //نمایش صفحه اولیه
        }

        [HttpPost] //برای این حالت امن‌تر است
        //[AjaxOnly]
        public ActionResult RenderModalPartialView()
        {
            //رندر پارشال ویوو صفحه مودال به همراه اطلاعات مورد نیاز آن
            return PartialView(viewName: "_ModalPartialView", model: new User { Name = "", LastName = "" });
        }

        [HttpPost]
        //[AjaxOnly]
        public ActionResult Index(User user) //ذخیره سازی اطلاعات
        {
            if (this.ModelState.IsValid)
            {
                //todo: SaveChanges;
                return Json(new { success = true });
            }

            this.ModelState.AddModelError("", "خطایی رخ داده است");
        }
    }
}
```

```

        return PartialView("_ModalPartialView", user);
    }
}

```

کدهای کنترلر برنامه در این حالت از سه قسمت تشکیل می‌شود:
 الف) متد Index حالت HttpGet که صفحه ابتدایی را نمایش خواهد داد.
 ب) متد RenderModalPartialView یک partial view اضافه شده به برنامه به نام ModalPartialView.cshtml را بازگشت می‌دهد. این partial view در حقیقت همان فرمی است که قرار است به صورت مودال نمایش داده شود و پردازش آن نیز Ajax ایی باشد.
 ج) متد Index حالت HttpPost که نهایتاً اطلاعات فرم مودال را دریافت خواهد کرد. اگر پردازش موفقیت آمیز بود، نیاز است همانند کدهای فوق return Json صورت گیرد. در غیراینصورت مجدداً همان partial view را بازگشت دهید.

کدهای Index.cshtml

```

@{
    ViewBag.Title = "Index";
    var renderModalPartialViewUrl = Url.Action("RenderModalPartialView", "ModalFormAjax");
    var postDataUrl = Url.Action("Index", "ModalFormAjax");
}
<h2>
    Index</h2>
<a href="#" class="btn btn-primary" id="btnCreate">ثبت اطلاعات</a>
@section JavaScript
{
    <script type="text/javascript">
        $(function () {
            $('#btnCreate').click(function (e) {
                e.preventDefault(); // صورت معمول عمل نکند
                $.bootstrapModalAjaxForm({
                    postUrl: '@postDataUrl',
                    renderModalPartialViewUrl: '@renderModalPartialViewUrl',
                    renderModalPartialViewData: {},
                    loginUrl: '/login',
                    beforePostHandler: function () {
                    },
                    completeHandler: function () {
                        // Refresh: صفحه زیرین باشد
                        // location.reload();
                    },
                    errorHandler: function () {
                    }
                });
            });
        });
    </script>
}

```

این کدها متناظر هستند با کدهای view اکشن متد Index در حالت Get.
 - در اینجا یک لینک ساده در صفحه قرار گرفته و به کمک کلاس btn مجموعه bootstrap به شکل یک دکمه مزین شده است.
 - در ادامه نحوه استفاده از افزونه‌ای را که در ابتدای بحث طراحی کردیم، ملاحظه می‌کنید. کار با آن بسیار ساده است و تنها باید مسیرهای ارسال اطلاعات نهایی به سرور یا postDataUrl و مسیر دریافت partial view رندر شده یا renderModalPartialViewUrl به آن معرفی شود. سایر مسایل آن خودکار است.

کدهای ModalPartialView.cshtml یا همان فرم مودال برنامه

```

@model Mvc4TwitterBootstrapTest.Models.User
<div class="modal-header">
    <button type="button" class="close" data-dismiss="modal" aria-hidden="true">

```

```

        &times;</button>
    </div>
    <h5>افزودن کاربر جدید</h5>
</div>
@using (Html.BeginForm("Index", " ModalFormAjax", FormMethod.Post, new { @class = "modal-form" }))
{
    <div class="modal-body">
        @Html.ValidationSummary(true, null, new { @class = "alert alert-error alert-block" })
        <fieldset class="form-horizontal">
            <legend>مشخصات کاربر</legend>
            <div class="control-group">
                @Html.LabelFor(model => model.Name, new { @class = "control-label" })
                <div class="controls">
                    @Html.EditorFor(model => model.Name)
                    @Html.ValidationMessageFor(model => model.Name, null, new { @class = "help-inline"
                })
                </div>
            </div>
            <div class="control-group">
                @Html.LabelFor(model => model.LastName, new { @class = "control-label" })
                <div class="controls">
                    @Html.EditorFor(model => model.LastName)
                    @Html.ValidationMessageFor(model => model.LastName, null, new { @class = "help-
inline" })
                </div>
            </div>
        </fieldset>
    </div>

    <div class="modal-footer">
        <button class="btn btn-primary" type="submit">ارسال</button>
        <button class="btn" data-dismiss="modal" aria-hidden="true">انصراف</button>
    </div>
}

```

در اینجا [اطلاعات فرمی را](#) ملاحظه می‌کنید که قرار است به صورت مودال نمایش داده شود. نحوه طراحی آن بر اساس [نکات form-horizontal](#) است. همچنین div‌های modal-header، modal-body و modal-footer نیز به این فرم ویژه اضافه شده‌اند تا به خوبی توسط bootstrap پردازش گردد. حاصل نهایی این مبحث را در دو شکل ذیل ملاحظه می‌کنید. صفحه index نمایش دهنده یک دکمه و در ادامه باز شدن یک فرم مودال، پس از کلیک بر روی دکمه ثبت اطلاعات.

Index

ثبت اطلاعات

Index

ثبت اطلاعات

×

افزودن کاربر جدید

مشخصات کاربر

نام

لطفا نام را تکمیل کنید

نام خانوادگی

لطفا نام خانوادگی را تکمیل کنید

انصراف

ارسال

نظرات خوانندگان

نویسنده: وحید نصیری
تاریخ: ۱۴:۵۶ ۱۳۹۲/۰۴/۰۵

دریافت مثال این قسمت: [BootstrapPopupAjaxForm.zip](#)

+ [نحوه بازسازی وابستگی‌ها](#)

نویسنده: دانشجو
تاریخ: ۲۳:۲۳ ۱۳۹۲/۰۴/۱۸

عرض پاپ آپ چطور تعیین میشه؟
امکان تمام صفحه شدن آن وجود دارد؟

نویسنده: وحید نصیری
تاریخ: ۲۳:۳۷ ۱۳۹۲/۰۴/۱۸

با css قابل تنظیم است. برای تنظیم عرض و سایر مشخصات تمام صفحات مودال:

```
body .modal {  
    width: 560px;  
    margin-left: -280px;  
}
```

و یا در مثال فوق برای یک صفحه خاص:

```
$('#dialogDiv').modal({  
    backdrop: 'static', // صفحه مودال بسته نمی‌شود  
    keyboard: true  
}, 'show').css({  
    'width': '100px',  
    'max-width': '1000px',  
    'margin-left': '10px' });
```

نویسنده: علی رضا امیدیان
تاریخ: ۱۸:۲۹ ۱۳۹۲/۰۴/۲۷

با سلام
در اکشن زیر

```
public ActionResult Index(User user) // ذخیره سازی اطلاعات  
{  
    if(this.ModelState.IsValid)  
    {  
        //todo: SaveChanges;  
        return Json(new{ success = true});  
    }  
  
    this.ModelState.AddModelError("", "خطایی رخ داده است");  
    return PartialView("_ModalPartialView", user);  
}
```

اگر 4 خط اول را کامنت کنیم، یعنی فرض کنیم خطایی اتفاق می‌افتد، که در برنامه‌های واقعی امکان پذیر است، از بار دوم به بعد، فرم از حالت مودال خارج می‌شود. علت چیست و راه حل کدام است؟

نویسنده: وحید نصیری
تاریخ: ۱۹:۵۸ ۱۳۹۲/۰۴/۲۷

در متد processAjaxForm در قسمت options.errorHandler آن چون dialogContent مجدداً مقدار دهی می‌شود، نیاز است سیم کشی‌های آن مجدداً برقرار شوند، شامل مباحث اعتبارسنجی و همچنین تحت کنترل قرار دادن دکمه submit آن. یعنی به صورت زیر

```
} else {
    $('#dialogContent').html(result);

    $.validator.unobtrusive.parse("#dialogContent");
    enablePostbackValidation();
    processAjaxForm('#dialogContent');

    if (options.errorHandler)
        options.errorHandler();
}
```

نویسنده: علی رضا امیدیان
تاریخ: ۱۴:۳۷ ۱۳۹۲/۰۴/۲۸

با سلام و تشکر از پاسخ شما

نویسنده: پیام
تاریخ: ۸:۵۵ ۱۳۹۲/۰۶/۰۱

با سلام آقای نصیری من برای استفاده از این مثال تو سایت خودم مجبور از بسته jquery-1.9.1 استفاده کنم این درحالی که من برای اجرای منوی superfish باید از بسته ای jquery-1.6 به پایین استفاده کنم برای حل همچنین مشکلاتی چه کاری باید کنم ؟

نویسنده: وحید نصیری
تاریخ: ۹:۰۰ ۱۳۹۲/۰۶/۰۱

- از آخرین نسخه افزونه یاد شده استفاده کنید. اگر ندارد، با نویسنده آن تماس بگیرید و درخواست ارتقاء آن را بدهید.
- jquery-1.9.1 نباید تداخلی با کار شما داشته باشد. یک سری از مسایل از نگارش 2 آن به بعد هست که [حذف شدند](#) و منسوخ شده اعلام گردیدند. برای رفع آن ، نیاز است از کتابخانه [jQuery Migrate](#) نیز استفاده کنید.

نویسنده: سیروان عقیفی
تاریخ: ۲۰:۳۴ ۱۳۹۲/۰۷/۱۰

ممنون از شما واقعاً مطلب جالبی بود،

فقط یک سوال :

کلاس‌های مدل من Product و Category است، نحوه نمایش گروه محصولات در DropDownList به چه صورتی است؟ ابتدا من یک ViewModel ایجاد کردم :

```
public class ProductCategoryViewModel
{
    public Product Product{get;set;}
    public Category Category{get;set;}
}
```

و سپس در RenderModalPartialView به PartialView مربوطه پاس دادم :

```
public ActionResult RenderModalPartialView()
{
    //رندر پارشال ویوو صفحه مودال به همراه اطلاعات مورد نیاز آن
    return PartialView(viewName: "_ModalPartialView", model: new ProductCategoryViewModel());
}
```

در PartialView هم برای نمایش نام گروه محصولات از طریق ViewBag لیست گروه محصولات داده شده را در یک

DropDownList به صورت زیر استفاده کردم :

```
@Html.DropDownListFor(model => model.Product.CategoryId, (SelectList)ViewBag.CategryNames, "--گروه--")
(موردنظر را انتخاب کنید)
```

اما روش فوق کارساز نیست و با کلیک بر روی باتن Modal را نمایش نمی‌دهد .

و به سوال دیگه اینکه چطور قالب [T4 تولید شده](#) رو برای حالت فرم‌های مودال بازنویسی کنیم؟ البته ظاهراً باید توی قسمت :

```
if(mvcHost.IsPartialView) {
#>
<#
```

قالب موردنظر (ModalPartialView_) برای حالت مودال فرم رو باید بنویسم، من با سینتکس T4 آشنا نیستم ، میشه لطف کنید راهنمایی کنید؟

نویسنده: وحید نصیری
تاریخ: ۲۱:۵۳ ۱۳۹۲/۰۷/۱۰

- این مطلب برای بوت استرپ 2 نوشته شده. (از فایل‌های نگارش 3 استفاده نکنید)
+ چه چیزی رو نمایش نمی‌ده؟ کلا صفحه باز نمیشه؟ یا اینکه صفحه باز میشه اما دراپ داون آن خالی است؟ اگر صفحه باز نمیشه که سورت کامل این قسمت [در اولین نظر بحث](#) ارسال شده. مقایسه کنید چه مواردی رو لحاظ نکردید. همچنین برنامه رو [در فایرباگ دیباگ کنید](#) شاید اسکریپتی فراموش شده. اگر دراپ داون صفحه باز شده خالی است، طبیعی هست. چون در متد RenderModalPartialView که نوشتید ViewBag ایی مقدار دهی نشده. بنابراین دیتاسورت مدنظر شما نال هست. روش بهتر این است که یک خاصیت به ViewModel ایی که تعریف کردید اضافه کنید:

```
public IEnumerable<SelectListItem> Names { get; set; }
```

بعد این خاصیت را در زمان بازگشت اطلاعات از اکشن متد مقدار دهی کنید:

```
model: new ProductCategoryViewModel { Names = ... }
```

- نیازی نیست کلا بازنویسی شود. در Razor view engine تفاوتی بین فایل نهایی View و Partial View نیست. به همین جهت برای اینکه این دو با هم اشتباه گرفته نشوند یک _ در ابتدای نام partial view قرار می‌دهند. فقط چند div مخصوص modal-header و موارد دیگری که در متن ذکر شده را باید به قالب قبلی اضافه کنید تا برای فرم‌های مودال هم کار کند. تفاوت دیگری ندارد.

نویسنده: سیروان عقیفی
تاریخ: ۲۱:۵۵ ۱۳۹۲/۰۷/۱۰

تغییر کدهای T4 برای حالت فرم‌های Modal به صورت زیر است :

```
<#
  if (!mvcHost.IsContentPage) {
#>

<#
  }
#>
<div class="modal-header">
  <button type="button" class="close" data-dismiss="modal" aria-hidden="true">
    &times;</button>
```

```

    <#= mvcHost.ViewDataType.Name #>
</div>
@using (Html.BeginForm()) {
<div class="modal-body">
    @Html.ValidationSummary(true, null, new { @class = "alert alert-error alert-block" })

    <fieldset class="form-horizontal">
        <legend><#= mvcHost.ViewDataType.Name #></legend>

<#
foreach (ModelProperty property in GetModelProperties(mvcHost.ViewDataType)) {
    if (!property.IsPrimaryKey && !property.IsReadOnly && property.Scaffold) {
#>
        <div class="control-group">
<#
            if (property.IsForeignKey) {
#>
                @Html.LabelFor(model => model.<#= property.Name #>, "<#= property.AssociationName #>", new
{@class="control-label"})
<#
            } else {
#>
                @Html.LabelFor(model => model.<#= property.Name #>, new {@class="control-label"})
<#
            }
#>

            <div class="controls">
<#
                if (property.IsForeignKey) {
#>
                    @Html.DropDownList("<#= property.Name #>", String.Empty)
<#
                } else {
#>
                    @Html.EditorFor(model => model.<#= property.Name #>)
<#
                }
#>
                @Html.ValidationMessageFor(model => model.<#= property.Name #>, null, new {@class="help-
inline"})
</div>
            </div>

<#
        }
    }
#>
</fieldset>
</div>

<div class="modal-footer">
    <button class="btn btn-primary" type="submit">
        ارسال</button>
    <button class="btn" data-dismiss="modal" aria-hidden="true">
        انصراف</button>
</div>
}

```

حالا تنها مشکلم نحوه نمایش گروه محصولات داخل DropDownList است.

نویسنده: مسعود بهرامی
تاریخ: ۵:۳۸ ۱۳۹۲/۰۹/۰۲

حال یه سوال اگه بخواهیم مثلا پس از اینکه کاربر اطلاعات را به شکل درست وارد کرده و اطلاعات هم به درستی ذخیره شده باشه یعنی تمام کار به درستی انجام شده باشه پس از بستن پنجره مودال یه پیام مثلا به شکل پاپ آپ به کاربر نشون بدیم مبنی بر صحت انجام کار باید به چه صورت عمل کرد یا مثل بالا جایی که شما نوشتین خطایی رخ داده است. منظورم شبیه چیزیه که تو همین وبلاگ عمل میشه یعنی اینجا پس از مثلا نوشتن این نظر و ارسال بدون اشکال به سرور به بنده یه پیام نشون میده که بدونم نتیجه‌ی کارمو با تشکر (:)

نویسنده: وحید نصیری
تاریخ: ۹:۲۶ ۱۳۹۲/۰۹/۰۲

این مورد پیش بینی شده در این مثال. اگر به خروجی اکشن متد دقت کنید، چنین چیزی است:

```
return Json(new { success = true });
```

همین مورد یعنی بررسی مقدار success دریافتی، جایی که عملیات Ajax ایی ارسال اطلاعات پایان می‌یابد انجام شده:

```
if (result.success) {
    $('#dialogDiv').modal('hide');
    if (options.completeHandler)
        options.completeHandler();
}
```

بنابراین تنها کاری که باید انجام بدید، قرار دادن کدهای نمایش اطلاعات نهایی در callbackهای completeHandler و با errorHandler مربوط به افزونه \$.bootstrapModalAjaxForm است. اینجا دست شما باز است. اگر علاقمند بودید از یک alert ساده استفاده کنید یا از همین روش نمایش صفحات مودال [به نحوی دیگر](#) یا از صدها [افزونه jQuery](#) موجود برای نمایش پیام‌ها.

نویسنده: مسعود بهرامی
تاریخ: ۱۶:۵۰ ۱۳۹۲/۰۹/۰۲

مرسی؛ به مورد دیگه با عرض پوزش تو این فرم مودال چون Ajax هستش به مشکل که داریم وقتی که بخوایم دیتا سورس رو مثلاً برای کومبو باکس تلریک پر کنیم مثل مثال ساده‌ی زیر

```
@(Html.Kendo().ComboBoxFor(model => model.ParentId)
    .DataTextField("Name")
    .DataValueField("Id")
    .HtmlAttributes(new { style = "width:150px" })
    .Filter("contains")
    .AutoBind(false)
    .MinLength(3)
    .Placeholder(CategoryResource.Category)
    .DataSource(source =>
    {
        source.Read(read =>
        {
            read.Action("GetAllGroups", "Utility");
        })
        .ServerFiltering(false);
    })
)
```

چون تو این حالت اصلاً کومبو کندو رندر نمیشه برای این حالت که مشکل اساسی هم هستش آیا راهی وجود داره من اطلاعات رو تو ViewBag هم ریختم و تو کد بالا بجای دیتا سورس از خاصیت باید اون بصورت دستی استفاده کردم یعنی اونو دستی باید کردم به ViewBag اما باز هم قایده نداشت چون کومبو باکس کندو اصلاً Render نمیشه ممنون میشم اگه راهنمایی کنین :

نویسنده: وحید نصیری
تاریخ: ۱۹:۲ ۱۳۹۲/۰۹/۰۲

[کمی بالاتر](#) در جواب سؤال مشابهی پاسخ دادم. باید یک ViewModel درست کنید جایی که return PartialView توسط عملیات Ajax ایی درخواست می‌شود، منبع داده View در حال رندر را تامین کند.

نویسنده: مجید
تاریخ: ۲۲:۰ ۱۳۹۲/۰۹/۱۴

اگه بخوایم از بوت استرپ 3 استفاده کنیم چه تغییری رو باید انجام بدیم؟

نویسنده: وحید نصیری
تاریخ: ۲۳:۲۸ ۱۳۹۲/۰۹/۱۴

در دوره « [ارتقاء به Bootstrap 3](#) » بحث شده.

نویسنده: حسین
تاریخ: ۱۷:۳۸ ۱۳۹۲/۱۱/۲۵

با سلام و تشکر
موقعی که اطلاعات اینجوری باشه درست کار میکنه و نتیجه رو نمایش میده

```
return Json( "test");
```

```
completeHandler: function (data) {  
    alert(data)  
}
```

ولی موقعی که مثلا اینجوری مینویسم

```
return Json(  
    new  
    {  
        success = true,  
        message = Resources.Global.Texts.Successful  
    });
```

```
completeHandler: function (data) {  
    alert(data.success)  
}
```

خطای مربوطه هم Cannot read property 'message' of undefined میده تو فایر باگ هم میبینم اطلاعات در response وجود داره ولی اینجا نمایش نمیده

نویسنده: وحید نصیری
تاریخ: ۱۷:۴۷ ۱۳۹۲/۱۱/۲۵

پروژه سورس باز هست. به قسمت success سورس آن دقت کنید (سطر 74 کد افزونه):

```
success: function (result) {  
    if (result.success) {  
        $('#dialogDiv').modal('hide');  
        if (options.completeHandler)  
            options.completeHandler();  
    }  
}
```

به completeHandler فراخوانی شده، اطلاعاتی ارسال نمی‌شود.
مطابق نیاز خودتان این قسمت را تغییر داده و استفاده کنید. مثلا result دریافتی را به آن متد ارسال کنید.

نویسنده: مجتبی فخاری
تاریخ: ۲۳:۳۴ ۱۳۹۲/۱۲/۲۰

با سلام
منظورتون رو از مطلب زیر نفهمیدم. میشه بگید چطوری میتونیم با استفاده از alert خود بوت استرپ نتیجه را از کنترلر به فرم مدال ارسال کنیم و در آنجا نمایش دهیم؟ «این مورد پیش بینی شده در این مثال. اگر به خروجی اکشن متد دقت کنید، چنین چیزی است:

return Json(new { success = true });	1
--	---

همین مورد یعنی بررسی مقدار success دریافتی، جایی که عملیات Ajax ایی ارسال اطلاعات پایان می‌یابد انجام شده:

if (result.success) {	1
\$('#dialogDiv').modal('hide');	2
if (options.completeHandler)	3
options.completeHandler();	4

بنابراین تنها کاری که باید انجام بدید، قرار دادن کدهای نمایش اطلاعات نهایی در callbackهای completeHandler ویا errorHandler مربوط به افزونه \$.bootstrapModalAjaxForm است. اینجا دست شما باز است. اگر علاقمند بودید از یک alert ساده استفاده کنید»

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۲/۲۰ ۲۳:۵۸

من اینجا یک خاصیت دلخواه به نام success اضافه کرده‌ام. دست شما باز است. تعداد خاصیت‌های بیشتری را به شیء anonymous بازگشت داده شده از کنترلر اضافه کنید. محدودیتی ندارد. این شیء نهایتاً در سمت کلاینت تحت عنوان result دریافت شده. یا مستقیماً با آن کار کنید یا این result را به متد completeHandler ارسال کنید تا در جای دیگری توسط completeHandler تعریف شده قابل استفاده باشد.

نویسنده: ع طاهری
تاریخ: ۱۳۹۲/۱۲/۲۵ ۱۴:۰۶

سلام.
برای پیغامهایی که در این سایت نمایش داده می‌شوند مانند "پیغامهای خطا" به رنگ قرمز یا "پیغامهای info" به رنگ آبی و ... که در وسط صفحه ظاهر شده و بعد از چند ثانیه ناپدید می‌شوند از modal استفاده شد؟ لطفاً راهنماییم کنید که چگونه میتوانم همچنین پیغامهایی را نمایش دهم.
با سپاس فراوان

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۱۲/۲۵ ۱۴:۲۴

از [noty](#) استفاده شده.

نویسنده: مجتبی فخاری
تاریخ: ۱۳۹۳/۰۲/۰۱ ۰:۳۵

با سلام
اگر بخواهیم برای Edit از این مدل استفاده کنیم چطوری می‌توانیم اطلاعات مثلاً Id را به متد مورد نظر پاس بدهیم. آیا باید در این خط Id, renderModalPartialViewData: {}, ارسال نماییم؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۰۲/۰۱ ۰:۴۴

```
$.ajax({
    type: "POST",
    url: options.renderModalPartialViewUrl,
```

```
data: options.renderModalPartialViewData,
```

مطابق سورس فوق، data ارسالی عملیات ajax با renderModalPartialViewData مقدار دهی می‌شود.

نویسنده: M.Shakeri

تاریخ: ۱۴:۵۷ ۱۳۹۳/۰۵/۲۵

تو مثالی که آپلود کردین اگه پارامتر بفرستیم خطای 500 میده.

```
$.bootstrapModalAjaxForm({
    postUrl: '@postDataUrl',
    renderModalPartialViewUrl: '@renderModalPartialViewUrl',
    renderModalPartialViewData: { id: 1},
```

```
public ActionResult RenderModalPartialView(int id)
{
    //رندر پارشال ویوو صفحه مودال به همراه اطلاعات مورد نیاز آن
    return PartialView(viewName: "_ModalPartialView", model: new User { Name = "", LastName = 
"" });
}
```

✖ POST <http://localhost:1032/ModalFormAjax/RenderModalPartialView> 500 (Internal Server Error)

نویسنده: وحید نصیری

تاریخ: ۱۵:۳۶ ۱۳۹۳/۰۵/۲۵

```
renderModalPartialViewData: JSON.stringify({ id: 1 }),
```