

فرض کنید تعدادی ردیف در گزارشی نمایش داده شده‌اند. قصد داریم برای هر ردیف یک دکمه حذف را قرار دهیم. این حذف باید Ajax ایی باشد؛ به علاوه در حین حذف ردیف، پویانمایی محو آن ردیف را نیز سبب شود.

### عملیات عنوان

حذف	عنوان 1
حذف	عنوان 2
حذف	عنوان 3
حذف	عنوان 4

مدل و منبع داده برنامه

```
namespace jQueryMvcSample06.Models
{
    public class BlogPost
    {
        public int Id { set; get; }
        public string Title { set; get; }
        public string Body { set; get; }
    }
}
```

```
using System.Collections.Generic;
using jQueryMvcSample06.Models;

namespace jQueryMvcSample06.DataSource
{
    /// <summary>
    /// منبع داده فرضی جهت سهولت دموی برنامه
    /// </summary>
    public static class BlogPostDataSource
    {
        private static IList<BlogPost> _cachedItems;
        static BlogPostDataSource()
        {
            _cachedItems = createBlogPostsInMemoryDataSource();
        }

        /// <summary>
        /// هدف صرفا تهیه یک منبع داده آزمایشی ساده تشکیل شده در حافظه است
        /// </summary>
        private static IList<BlogPost> createBlogPostsInMemoryDataSource()
        {
            var results = new List<BlogPost>();
            for (int i = 1; i < 30; i++)
            {
                results.Add(new BlogPost { Id = i, Title = "عنوان " + i, Body = "متن ... متن ... متن "
+ i});
            }
            return results;
        }
    }
}
```

```

    }
    public static IList<BlogPost> LatestBlogPosts
    {
        get { return _cachedItems; }
    }
}

```

در اینجا مدل برنامه که ساختار نمایش یک سری مطلب را تهیه می‌کند، ملاحظه می‌کنید؛ به علاوه یک منبع داده فرضی تشکیل شده در حافظه جهت سهولت دموی برنامه.

### کنترلر برنامه

```

using System.Web.Mvc;
using System.Web.UI;
using jQueryMvcSample06.DataSource;
using jQueryMvcSample06.Security;

namespace jQueryMvcSample06.Controllers
{
    public class HomeController : Controller
    {
        [HttpGet]
        public ActionResult Index()
        {
            var postsList = BlogPostDataSource.LatestBlogPosts;
            return View(postsList);
        }

        [AjaxOnly]
        [HttpPost]
        [OutputCache(Location = OutputCacheLocation.None, NoStore = true)]
        public ActionResult DeleteRow(int? postId)
        {
            if (postId == null)
                return Content(null);

            //todo: delete post from db

            return Content("ok");
        }
    }
}

```

کنترلر برنامه بسیار ساده بوده و نکته خاصی ندارد. در حین اولین بار نمایش صفحه، لیست مطالب را به View مرتبط ارسال می‌کند. همچنین یک اکشن متد حذف ردیف‌های نمایش داده شده را نیز در اینجا تدارک دیده‌ایم. این اکشن متد از طریق ارسال اطلاعات به صورت Ajax، شماره مطلب را در اختیار برنامه قرار می‌دهد که توسط آن در ادامه برای مثال می‌توان این رکورد را از بانک اطلاعاتی حذف کرد. امضای متد DeleteRow بر اساس پارامترهای ارسالی توسط jQuery Ajax مشخص و تنظیم شده‌اند:

```
data: JSON.stringify({ postId: postId }),
```

### View برنامه

```

@model IEnumerable<jQueryMvcSample06.Models.BlogPost>
@{
    ViewBag.Title = "Index";
    var postUrl = Url.Action(actionName: "DeleteRow", controllerName: "Home");
}
<h2>
حذف یک ردیف از اطلاعات به همراه پویانمایی محو شدن اطلاعات آن
</h2>
<table>
<tr>

```

```

        <th>
        عملیات
    </th>
    <th>
    عنوان
    </th>
</tr>
@foreach (var item in Model)
{
    <tr>
        <td>
            <span id="row-@item.Id">حذف</span>
        </td>
        <td>
            @item.Title
        </td>
    </tr>
}
</table>
@section JavaScript
{
    <script type="text/javascript">
        $(function () {
            $('span[id^="row"]').click(function () {
                var span = $(this);
                var postId = span.attr('id').replace('row-', '');
                var tableRow = span.parent().parent();
                $.ajax({
                    type: "POST",
                    url: '@postUrl',
                    data: JSON.stringify({ postId: postId }),
                    contentType: "application/json; charset=utf-8",
                    dataType: "json",
                    complete: function (xhr, status) {
                        var data = xhr.responseText;
                        if (xhr.status == 403) {
                            window.location = "/login";
                        }
                        else if (status === 'error' || !data || data == "nok") {
                            alert('خطایی رخ داده است');
                        }
                        else {
                            $(tableRow).fadeOut(600, 0, function () {
                                $(tableRow).remove();
                            });
                        }
                    }
                });
            });
        });
    </script>
}

```

کدهای View برنامه را در ادامه ملاحظه می‌کنید. اطلاعات مطالب دریافتی به صورت یک جدول در صفحه نمایش داده شده‌اند. در هر ردیف توسط یک span که با css تزئین گردیده است، یک دکمه حذف را تدارک دیده‌ایم. برای اینکه در حین کار با jQuery بتوانیم id هر ردیف را بدست بیاوریم، این id را در قسمتی از این span اضافه شده قرار داده‌ایم. در کدهای اسکریپتی صفحه، ابتدا کلیک بر روی کلیه span‌هایی که id آن‌ها با row شروع می‌شود را مونیتور خواهیم کرد:

```

$('span[id^="row"]').click(function () {

```

سپس هر زمان که بر روی یکی از این span‌ها کلیک شد، می‌توان بر اساس span جاری، id و همچنین tableRow مرتبط را استخراج کرد:

```

var span = $(this);
var postId = span.attr('id').replace('row-', '');
var tableRow = span.parent().parent();

```

اکنون که به این اطلاعات دسترسی پیدا کرده‌ایم، تنها کافی است آن‌ها را توسط متد ajax به کنترلر برنامه برای پردازش نهایی ارسال نمائیم. همچنین در پایان کار عملیات، توسط متدهای fadeTo و remove ایی که ملاحظه می‌کنید، سبب حذف نمایشی یک

ردیف به همراه پویانمایی محو آن خواهیم شد.

دریافت کدها و پروژه کامل این قسمت

[jQueryMvcSample06.zip](#)

## نظرات خوانندگان

نویسنده: صابر فتح الهی  
تاریخ: ۱۰:۵۹ ۱۳۹۲/۰۹/۲۶

در این روش، در صورتی که در طرف کنترلر از ValidateAntiForgeryToken استفاده شده باشد، این روش با خطا مواجه میشه درسته؟

نویسنده: وحید نصیری  
تاریخ: ۱۱:۴۲ ۱۳۹۲/۰۹/۲۶

توکن را به view اضافه کنید (یکبار در بالای صفحه). در قسمت JSON.stringify مقدار آن را خوانده و ارسال کنید:

```
var token = $('[name=__RequestVerificationToken]').val();
$.ajax({
    // .....
    data: { __RequestVerificationToken: token, ..... },
    // .....
});
```

نویسنده: صابر فتح الهی  
تاریخ: ۱۳:۵۰ ۱۳۹۲/۰۹/۲۶

این روشم امتحان کردم قبلا اما خطای زیر دریافت میشه (توی فایرباگ چک کردم)

The required anti-forgery form field "\_\_RequestVerificationToken" is not present

در حالی که در بالای ویو این دستور هست

```
@Html.AntiForgeryToken()
```

نویسنده: وحید نصیری  
تاریخ: ۱۵:۵۹ ۱۳۹۲/۰۹/۲۶

با روش زیر امتحان کنید:

```
function addToken(data) {
    data.__RequestVerificationToken = $("input[name=__RequestVerificationToken]").val();
    return data;
}
$.ajax({
    // .....
    data: addToken({ postId: postId }), // اضافه کردن توکن
    dataType: "html", // نوع داده مهم است
    // .....
});
```

نویسنده: صابر فتح الهی  
تاریخ: ۱۷:۴۰ ۱۳۹۲/۰۹/۲۶

متشکرم از پاسخ شما  
اینم امتحان کردم نشد

نویسنده: وحید نصیری  
تاریخ: ۱۷:۵۷ ۱۳۹۲/۰۹/۲۶

من قبل از ارسال، امتحانش کرده بودم. جهت تکمیل بحث، پروژه آن پیوست شد:

[jQueryMvcSample06\\_v2.zip](#)

نویسنده: صابر فتح الهی  
تاریخ: ۱۸:۴۸ ۱۳۹۲/۰۹/۲۶

بله درست کار کرد  
من این خط را حذف نکرده بودم

```
contentType: "application/json; charset=utf-8",
```

این خط را کامنت کردم درست شد. از صبر و شکیبایی شما متشکرم

نویسنده: صابر فتح الهی  
تاریخ: ۲۲:۳۳ ۱۳۹۲/۰۹/۲۶

البته راه دیگری هم پیدا کردم.  
به نظرم جالب اومد.  
ابتدا یک اکشن فیلتر تعریف شده و در آن هدر درخواست خوانده شده و با محتوی کوکی چک می‌شود در صورتی که برابر باشند عمل انجام خواهد شد با این تفاوت باید توکن در زمان ارسال در هدر درخواست قرار گیرد مانند زیر:  
کد اکشن فیلتر:

```
public class ValidateJsonAntiForgeryTokenAttribute : ActionFilterAttribute
{
    #region Methods (1)
    // Public Methods (1)
    /// <summary>
    /// Called when [action executing].
    /// </summary>
    /// <param name="actionContext">The action context.</param>
    public void OnActionExecuting(HttpContext actionContext)
    {
        try
        {
            var cookieName = AntiForgeryConfig.CookieName;
            var headers = actionContext.Request.Headers;
            var cookie = headers
                .GetCookies()
                .Select(c => c[AntiForgeryConfig.CookieName])
                .FirstOrDefault();
            var rvt = headers.GetValues("__RequestVerificationToken").FirstOrDefault();
            AntiForgery.Validate(cookie != null ? cookie.Value : null, rvt);
        }
        catch
        {
            actionContext.Response =
            actionContext.Request.CreateErrorResponse(HttpStatusCode.Forbidden, "Unauthorized request.");
        }
    }
    #endregion Methods
}
```

سپس در اکشن دلخواه باید این اکشن فیلتر را بکار برد (در درخواست‌های json)

```
[HttpPost]
//[ValidateAntiForgeryToken]
[ValidateJsonAntiForgeryToken]
[OutputCache(Location = OutputCacheLocation.None, NoStore = true)]
public virtual ActionResult DeletePhoto(int? id)
{
```

```
string ret="error";
if (id != null && id > 0)
{
    bool result = _imageGalleryService.Value.Photos.Value.Delete(id.ToInt32());
    ret = result ? "ok" : "nok";
}
return Content(ret);
//return Json(new { ret });
}
```

سپس در طرف ویو به صورت زیر عمل شود

```
$.ajax({
    type: "POST",
    url: '@Url',
    data: JSON.stringify({ id: Id }),
    contentType: "application/json; charset=utf-8",
    // این قسمت اضافه شود
    headers: { __RequestVerificationToken:
        $("input[name=__RequestVerificationToken]").val() },
    dataType: "json",
    .....
```

امیدوارم برای دوستان هم کاربرد داشته باشد

نویسنده: ساشا  
تاریخ: ۱۵:۴ ۱۳۹۲/۰۹/۲۷

برای Web Form هم به همین صورت است ؟ آیا این مثال برای وب فرم هم کاربرد دارد ؟ نحوه استفاده آن یکسان است ؟

نویسنده: ساشا  
تاریخ: ۱۵:۵ ۱۳۹۲/۰۹/۲۷

لطفا سوره پروژه رو بگذارید ، تشکر

نویسنده: وحید نصیری  
تاریخ: ۱۵:۱۳ ۱۳۹۲/۰۹/۲۷

خیر (سمت سرور آن یکسان نیست). این دوره مربوط به MVC است؛ مثال آن هم مرتبط است به MVC. سوره پروژه هم در بحث [پیوست شده](#) (دو بار).

نویسنده: رضا منصوری  
تاریخ: ۱۰:۱۶ ۱۳۹۲/۱۱/۰۷

سلام. ابتدا از مطلب خوبتون تشکر میکنم . واقعا بسیار جالب و کاربردیه .  
من از این مثال استفاده کردم و درست کار میکنه ولی وقتی میخوام از کاربر تایید برای حذف بگیرم ( با استفاده از [jquery-impromptu](#) ) به مشکلی پیش میاد .  
حالت عادی که Id مربوط به پیغاممون درست انتخاب میشه.

```

209         $('span[id^="row"]').click(function () {
210             var span = $(this);
211             var messageid = span.attr('id').replace('row-', '');
212             var tableRow = span.parent().parent();
213             $.ajax({
214                 type: "POST",
215                 url: '/User/DeleteMessageSend',
216                 data: JSON.stringify({ id: messageid }),
217                 contentType: "application/json; charset=utf-8",
218                 dataType: "json",
219                 complete: function (xhr, status) {
220                     var data = xhr.responseText;
221                     if (xhr.status == 403) {
222                         window.location = "/login";

```

ولی وقتی تایید میگیرم

```

214         focus: 2,
215         submit: function(e, v, m, f) {
216             if (!v) {
217                 return;
218             } else {
219                 var span = $(this);
220                 var messageid = span.attr('id').replace('row-', '');
221                 var tableRow = span.parent();
222                 $.ajax({
223                     type: "POST",
224                     url: '/User/DeleteMessageSend',
225                     data: JSON.stringify({ id: messageid }),
226                     contentType: "application/json; charset=utf-8",
227                     dataType: "json",

```

موس روی messageid هستش و نوشته‌ی قرمز مقدار id هستش که قطعاً درست نیست.

مسئله مشکل از این خطه که id رو انتخاب میکنه

```

var span = $(this);
var messageid = span.attr('id').replace('row-', '');

```

ولی نمیدونم چجوری id رو انتخاب کنم!



اینم کدها که تایید کاربرو بهش اضافه کردم

```

<script type="text/javascript">
$(function() {
    $('span[id^="row"]').click(function() {
        $.prompt("آیا برای حذف اطلاعات موجود اطمینان دارید ؟",
            {
                title: 'Title',
                buttons: { "بله": true, "خیر": false },
                focus: 2,
                submit: function(e, v, m, f) {
                    if (!v) {
                        return; // وقتی کاربر تایید نکرد
                    } else { // وقتی کاربر تایید کرد
                        var span = $(this);
                        var messageid = span.attr('id').replace('row-', '');
                        var tableRow = span.parent().parent();
                        $.ajax({
                            type: "POST",
                            url: '@postUrl',
                            data: JSON.stringify({ id: messageid }),
                            contentType: "application/json; charset=utf-8",
                            dataType: "json",
                            complete: function(xhr, status) {
                                var data = xhr.responseText;
                                if (xhr.status == 403) {
                                    window.location = "/login";
                                } else if (status === 'error' || !data || data == "nok") {
                                    var noty = window.noty({ text: "خطایی رخ داده", type:
'error', layout: 'center', timeout: 2500 });
                                } else {
                                    $(tableRow).fadeOut(600, 0, function() {
                                        $(tableRow).remove();
                                    });
                                    var noty = window.noty({ text: "پیغام حذف شد", type:
'information', layout: 'center', timeout: 2500 });
                                }
                            }
                        });
                    }
                }
            }
        );
    });
});
});
</script>

```

نویسنده: وحید نصیری  
تاریخ: ۱۴:۳۸ ۱۳۹۲/۱۱/۰۷

زمانیکه \$.prompt نمایان می‌شود، this ذکر شده به قسمتی از طرحبندی خود این افزونه اشاره می‌کند و نه لایه زیرین آن که جدول ما است. بنابراین سه سطر مربوط به یافتن Id و سطر جاری را پیش از فراخوانی \$.prompt قرار دهید.