

Kendo UI DataSource جهت تامین داده‌های سمت کلاینت [ویجت‌های مختلف KendoUI](#) طراحی شده‌است و به عنوان یک اینترفیس استاندارد قابل استفاده توسط تمام کنترل‌های داده‌ای Kendo UI کاربرد دارد. Kendo UI DataSource امکان کار با منابع داده محلی، مانند اشیاء و آرایه‌های جاوا اسکریپتی و همچنین منابع تامین شده از راه دور، مانند JSONP، JSON و XML را دارد. به علاوه توسط آن می‌توان اعمال ثبت، ویرایش و حذف اطلاعات، به همراه صفحه بندی، گروه بندی و مرتب سازی داده‌ها را کنترل کرد.

استفاده از منابع داده محلی

در ادامه مثالی را از نحوه‌ی استفاده از یک منبع داده محلی جاوا اسکریپتی، مشاهده می‌کنید:

```
<script type="text/javascript">
$(function () {
    var cars = [
        { "Year": 2000, "Make": "Hyundai", "Model": "Elantra" },
        { "Year": 2001, "Make": "Hyundai", "Model": "Sonata" },
        { "Year": 2002, "Make": "Toyota", "Model": "Corolla" },
        { "Year": 2003, "Make": "Toyota", "Model": "Yaris" },
        { "Year": 2004, "Make": "Honda", "Model": "CRV" },
        { "Year": 2005, "Make": "Honda", "Model": "Accord" },
        { "Year": 2000, "Make": "Honda", "Model": "Accord" },
        { "Year": 2002, "Make": "Kia", "Model": "Sedona" },
        { "Year": 2004, "Make": "Fiat", "Model": "One" },
        { "Year": 2005, "Make": "BMW", "Model": "M3" },
        { "Year": 2008, "Make": "BMW", "Model": "X5" }
    ];

    var carsDataSource = new kendo.data.DataSource({
        data: cars
    });

    carsDataSource.read();
    alert(carsDataSource.total());
});
</script>
```

در اینجا cars آرایه‌ای از اشیاء جاوا اسکریپتی بیانگر ساختار یک خودرو است. سپس برای معرفی آن به Kendo UI، کار با مقدار دهی خاصیت data مربوط به new kendo.data.DataSource شروع می‌شود. ذکر new kendo.data.DataSource به تنهایی به معنای مقدار دهی اولیه است و در این حالت منبع داده مورد نظر، استفاده نخواهد شد. برای مثال اگر متد total آن را جهت یافتن تعداد عناصر موجود در آن فراخوانی کنید، صفر را بازگشت می‌دهد. برای شروع به کار با آن، نیاز است ابتدا متد read را بر روی این منبع داده مقدار دهی شده، فراخوانی کرد.

استفاده از منابع داده راه دور

در برنامه‌های کاربردی، عموماً نیاز است تا منبع داده را از یک وب سرور تامین کرد. در اینجا نحوه‌ی خواندن اطلاعات JSON بازگشت داده شده از جستجوی توئیتر را مشاهده می‌کنید:

```
<script type="text/javascript">
$(function () {
    var twitterDataSource = new kendo.data.DataSource({
        transport: {
            read: {
                url: "http://search.twitter.com/search.json",
                dataType: "jsonp",
                contentType: 'application/json; charset=utf-8',
                type: 'GET',
                data: { q: "#kendoui" }
            },
            schema: { data: "results" }
        }
    });
});
```

```

    },
    error: function (e) {
        alert(e.errorThrown.stack);
    }
  });
});
</script>

```

در قسمت transport، جزئیات تبادل اطلاعات با سرور راه دور مشخص می‌شود؛ برای مثال url ارائه دهنده‌ی سرویس، dataType بیانگر نوع داده مورد انتظار و data کار مقدار دهی پارامتر مورد انتظار توسط سرویس توئیتر را انجام می‌دهد. در اینجا چون صرفاً عملیات خواندن اطلاعات صورت می‌گیرد، خاصیت read مقدار دهی شده‌است. در قسمت schema مشخص می‌کنیم که اطلاعات JSON بازگشت داده شده توسط توئیتر، در فیلد results آن قرار دارد.

کار با منابع داده OData

علاوه بر فرمت‌های یاد شده، Kendo UI DataSource امکان کار با اطلاعاتی [از نوع OData](#) را نیز دارا است که تنظیمات ابتدایی آن به صورت ذیل است:

```

<script type="text/javascript">
    var moviesDataSource = new kendo.data.DataSource({
        type: "odata",
        transport: {
            read: "http://demos.kendoui.com/service/Northwind.svc/Orders"
        },
        error: function (e) {
            alert(e.errorThrown.stack);
        }
    });
});
</script>

```

همانطور که ملاحظه می‌کنید، تنظیمات ابتدایی آن اندکی با حالت remote data پیشین متفاوت است. در اینجا ابتدا نوع داده‌ی بازگشتی مشخص می‌شود و در قسمت transport، خاصیت read آن، آدرس سرویس را دریافت می‌کند.

یک مثال: دریافت اطلاعات از ASP.NET Web API

یک پروژه‌ی جدید ASP.NET را آغاز کنید. تفاوتی نمی‌کند که Web forms باشد یا MVC؛ از این جهت که مباحث [Web API](#) در هر دو یکسان است.

سپس یک کنترلر جدید Web API را به نام ProductsController با محتوای زیر ایجاد کنید:

```

using System.Collections.Generic;
using System.Web.Http;

namespace KendoUI02
{
    public class Product
    {
        public int Id { set; get; }
        public string Name { set; get; }
    }

    public class ProductsController : ApiController
    {
        public IEnumerable<Product> Get()
        {
            var products = new List<Product>();
            for (var i = 1; i <= 100; i++)
            {
                products.Add(new Product { Id = i, Name = "Product " + i });
            }
            return products;
        }
    }
}

```

در این مثال، هدف صرفاً ارائه یک خروجی ساده JSON از طرف سرور است.
در ادامه نیاز است تعریف مسیریابی ذیل نیز به فایل Global.asax.cs برنامه اضافه شود تا بتوان به آدرس api/products سایت، دسترسی یافت:

```
using System;
using System.Web.Http;
using System.Web.Routing;

namespace KendoUI02
{
    public class Global : System.Web.HttpApplication
    {
        protected void Application_Start(object sender, EventArgs e)
        {
            RouteTable.Routes.MapHttpRoute(
                name: "DefaultApi",
                routeTemplate: "api/{controller}/{id}",
                defaults: new { id = RouteParameter.Optional }
            );
        }
    }
}
```

در ادامه فایلی را به نام Index.html (یا در یک View و یا یک فایل aspx دلخواه)، محتوای ذیل را اضافه کنید:

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <meta charset="utf-8" />
    <title>Kendo UI: Implementing the Grid</title>

    <link href="styles/kendo.common.min.css" rel="stylesheet" type="text/css" />
    <link href="styles/kendo.default.min.css" rel="stylesheet" type="text/css" />
    <script src="js/jquery.min.js" type="text/javascript"></script>
    <script src="js/kendo.all.min.js" type="text/javascript"></script>
</head>
<body>

    <div id="report-grid"></div>
    <script type="text/javascript">
        $(function () {
            var productsDataSource = new kendo.data.DataSource({
                transport: {
                    read: {
                        url: "api/products",
                        dataType: "json",
                        contentType: 'application/json; charset=utf-8',
                        type: 'GET'
                    }
                },
                error: function (e) {
                    alert(e.errorThrown.stack);
                },
                pageSize: 5,
                sort: { field: "Id", dir: "desc" }
            });

            $("#report-grid").kendoGrid({
                dataSource: productsDataSource,
                autoBind: true,
                scrollable: false,
                pageable: true,
                sortable: true,
                columns: [
                    { field: "Id", title: "#" },
                    { field: "Name", title: "Product" }
                ]
            });
        });
    </script>
</body>
</html>
```

- ابتدا فایل‌های اسکریپت و CSS مورد نیاز Kendo UI اضافه شده‌اند.
- گرید صفحه، در محل div ایی با id مساوی report-grid تشکیل خواهد شد.
- سپس DataSource ایی که به آدرس api/products اشاره می‌کند، تعریف شده و در آخر productsDataSource را توسط یک kendoGrid نمایش داده‌ایم.
- نحوه‌ی تعریف productsDataSource، در قسمت استفاده از منابع داده راه دور ابتدای بحث توضیح داده شد. در اینجا فقط دو خاصیت pageSize و sort نیز به آن اضافه شده‌اند. این دو خاصیت بر روی نحوه‌ی نمایش گرید نهایی تاثیر گذار هستند. تعداد رکورد هر صفحه را مشخص می‌کند و sort نحوه‌ی مرتب سازی را بر اساس فیلد Id و در حالت نزولی قرار می‌دهد.
- در ادامه، ابتدایی‌ترین حالت کار با kendoGrid را ملاحظه می‌کنید.
- تنظیم dataSource و autoBind: true (حالت پیش فرض)، سبب خواهند شد تا به صورت خودکار، اطلاعات JSON از مسیر api/products خوانده شوند.
- سه خاصیت بعدی صفحه بندی و مرتب سازی خودکار ستون‌ها را فعال می‌کنند.
- در آخر هم دو ستون گرید، بر اساس نام‌های خواص کلاس Product تعریف شده‌اند.

| # | Product |
|---|-----------|
| 5 | Product 5 |
| 4 | Product 4 |
| 3 | Product 3 |
| 2 | Product 2 |
| 1 | Product 1 |

سورس کامل این قسمت را از اینجا می‌توانید دریافت کنید:

[KendoUI02.zip](#)

نظرات خوانندگان

نویسنده: ژوپتر
تاریخ: ۱۳۹۳/۱۱/۰۹ ۱۰:۲۷

سلام

چرا زمان اجرا به جای نمایش اطلاعات گرید، پیام undefined داده می‌شود؟ بنده از MVC استفاده کردم و کاملاً مطابق مقاله مسیریابی و ... را اعمال کردم.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۱۱/۰۹ ۱۱:۱۰

[روی چه سطری](#) پیام خطا دریافت کردید؟

حین کار با کتابخانه‌های جاوا اسکریپتی باید مدام کنسول developer مرورگر را باز نگه دارید تا بتوانید خطاها را بهتر بررسی و دیباگ کنید.

نویسنده: ژوپتر
تاریخ: ۱۳۹۳/۱۱/۱۱ ۹:۳

خطا توسط error handler، گرفته می‌شود وقتی این بخش نیز حذف می‌شود مرورگر فقط منتظر دریافت اطلاعات از api/products می‌ماند و خطایی از کد گرفته نمی‌شود.

| Clear | Persist | All | HTML | CSS | JavaScript | XHR | Images | Plugins | Media | Fonts | Size | Time |
|-------------|---------|-----|------|-----|------------|-----|--------|---------|-------|-------|----------------------------|-----------------------|
| + | + | + | + | + | + | + | + | + | + | + | 40.8 KB | 1ms |
| + | + | + | + | + | + | + | + | + | + | + | 9.3 KB | 1ms |
| + | + | + | + | + | + | + | + | + | + | + | 91.9 KB | 2ms |
| + | + | + | + | + | + | + | + | + | + | + | 1.9 MB | 1ms |
| + | + | + | + | + | + | + | + | + | + | + | 58.3 KB | 8ms |
| + | + | + | + | + | + | + | + | + | + | + | 27.8 KB | 2ms |
| + | + | + | + | + | + | + | + | + | + | + | 4.3 KB | 47ms |
| + | + | + | + | + | + | + | + | + | + | + | 6.0 KB | 2ms |
| + | + | + | + | + | + | + | + | + | + | + | 606 B | 1ms |
| + | + | + | + | + | + | + | + | + | + | + | 0 B | 4ms |
| + | + | + | + | + | + | + | + | + | + | + | 366 B | 6ms |
| 13 requests | | | | | | | | | | | 2.1 MB (2.0 MB from cache) | 446ms (onload: 1.53s) |

| Clear | Persist | All | HTML | CSS | JavaScript | XHR | Images | Plugins | Media | Fonts | Size | Time |
|-------|---------|-----|------|-----|------------|-----|--------|---------|-------|-------|---------|------|
| + | + | + | + | + | + | + | + | + | + | + | 58.3 KB | 8ms |
| + | + | + | + | + | + | + | + | + | + | + | 27.8 KB | 2ms |
| + | + | + | + | + | + | + | + | + | + | + | 4.3 KB | 47ms |

| Headers | Response | HTML | Cache |
|---|----------|------|-------|
| <p>Response Headers</p> <p>Cache-Control: private</p> <p>Content-Length: 4412</p> <p>Content-Type: text/html; charset=utf-8</p> <p>Date: Sat, 31 Jan 2015 05:20:19 GMT</p> <p>Server: Microsoft-IIS/8.0</p> <p>X-AspNet-Version: 4.0.30319</p> <p>X-Powered-By: ASP.NET</p> <p>X-SourceFiles: =?UTF-8?B?QspcVXNlcnNcTW9oc2VuXERlc2t0b3BcTVZDV2ViQXBwbG1jYXRpb25cTVZDV2ViQXBwbG1jYXRpb25cSG9tZVxhcG1ocHJvZHVjZHM=?</p> | | | |

نویسنده: وحید نصیری
تاریخ: ۱۳۹۳/۱۱/۱۱ ۱۰:۳

خطای 404 به معنای یافتن نشدن مسیر تنظیمی url: "api/products" در برنامه شما است.

مطابق تصویر، مسیر `home /api/product` در حال جستجو است که به ریشه‌ی سایت اشاره نمی‌کند.

- آیا در ASP.NET MVC از یک اکشن متد برای بازگرداندن لیست جی‌سون محصولات استفاده کرده‌اید؟ اگر بله، از مطلب «[نحوه صحیح تولید Url در ASP.NET MVC](#)» کمک بگیرید؛ مثلاً آدرس آن چنین شکلی را پیدا خواهد کرد:

```
@Url.Action("method_name", "Home")
```

- اگر وب API است در یک برنامه‌ی MVC، از روش زیر استفاده کنید:

```
'@Url.RouteUrl("DefaultApi", new { httproute = "", controller = "products" })'
```

و البته فرض بر این است که مسیریابی `DefaultApi` پیشتر در برنامه‌ی شما ثبت شده‌است:

```
routes.MapHttpRoute(
    name: "DefaultApi",
    routeTemplate: "api/{controller}/{id}",
    defaults: new { id = RouteParameter.Optional }
);
```

نویسنده: ژوپیتز

تاریخ: ۱۱:۸ ۱۳۹۳/۱۱/۱۱

- ممنون؛ از وب API استفاده شده بود که با راهنمایی شما حل شد. ولی استفاده از خروجی `Json` کنترلر به‌نظرم بهتر و ساده‌تر اومد. آیا تفاوتی محسوسی بین این دو روش وجود داره؟

- آیا امکان استفاده مستقیم اشیا `Strongly Typed` هم در توابع این کتابخانه وجود داره؟ (منظورم همون `@model` به صورت مستقیم یا با واسطه است).

نویسنده: وحید نصیری

تاریخ: ۱۱:۲۰ ۱۳۹۳/۱۱/۱۱

- خیر. اگر هدف صرفاً بازگشت جی‌سون است، تفاوت خاصی ندارند. فقط `Web API` به صورت پیش فرض از `JSON.NET` استفاده می‌کند؛ [اطلاعات بیشتر](#) و همچنین با وب فرم‌ها هم سازگار است.

- بله: [استفاده از Expressionها جهت ایجاد Strongly typed view در ASP.NET MVC](#)

نویسنده: رضا نادری

تاریخ: ۲۲:۳۵ ۱۳۹۳/۱۲/۲۶

سلام؛ من می‌خواهم در متد `read`، خواندنم همراه با پارامتر باشد. به این معنا که آن رکوردهایی را بخوان که `Id=12` هست. چگونه میتوانم این کار را بکنم؟

نویسنده: وحید نصیری

تاریخ: ۰:۸ ۱۳۹۳/۱۲/۲۷

- مثال بحث جاری جهت اضافه شدن پارامترهای سفارشی به روز شد. با این [View](#) و این [کنترلر](#).
- مقایسه‌ی تغییرات انجام شده با نمونه‌ی قبلی [در اینجا](#).

نویسنده: رضا نادری

تاریخ: ۰:۳۱ ۱۳۹۳/۱۲/۲۷

من از `mvc` استفاده میکنم و فرمایشات شما را هم انجام دادم ولی پارامترها `null` هستند. آیا در `mvc` نحوه کار متفاوت است.

نویسنده: وحید نصیری

تاریخ: ۱۳۹۳/۱۲/۲۷ ۱:۱۰

خیر متفاوت نیست. متد Get در ASP.NET Web API معادل متد دارای ویژگی HttpGet در ASP.NET MVC است. این Get هم بر اساس 'GET' type: سمت کلاینت مشخص می‌شود. اگر مقدار آن به Post تنظیم شده، باید به متد Post سمت سرور یا اکشن متد HttpPost دار ارسال شود.

نویسنده: رضا نادری
تاریخ: ۱۳۹۴/۰۱/۰۴ ۱۵:۱۵

سلام من پروژه شما را دانلود کردم و هنگام تست وقتی پارامتر داشته باشد پیغام undefined می‌دهد ولی هنگامی که پارامتر نداشته باشد درست جواب می‌دهد.
می‌خواستم بدونم که آیا نیاز به افزونه ویا ... چیز دیگری هست که رو سیستم من نصب باشه؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۴/۰۱/۰۴ ۱۶:۴۷

نسخه‌ی تکمیلی ASP.NET MVC بحث جاری (^):

```

15 public ActionResult GetProducts(string param1, string param2)
16 {
17     var products = new List<Product>();
18     for (var i = 1; i <= 100; i++)
19     {
20         products.Add(new Product { Id = i, Name = "Product " + i });
21     }
22     return Json(products, JsonRequestBehavior.AllowGet);
23 }

```

نسخه‌ی تکمیلی ASP.NET Web API بحث جاری (^):

```

12 public class ProductsController : ApiController
13 {
14     public IEnumerable<Product> Get(string param1, string param2)
15     {
16         var products = new List<Product>();
17         for (var i = 1; i <= 100; i++)
18         {
19             products.Add(new Product { Id = i, Name = "Product " + i });
20         }
21         return products;
22     }
23 }

```

هر دو مورد پارامترهای ارسالی را بدون مشکل دریافت می‌کنند.

نویسنده: رضا نادری
تاریخ: ۱۳۹۴/۰۱/۰۴ ۲۱:۰۸

سلام؛ من در حل این مشکل به یک نکته برخورددم و آن اینه که وقتی view من دقیقاً مشابه آن چیزی هست که شما در جواب فوق

آدرس دادید من موفق به دریافت پارامترها میگردم. اما اگر ویو من به شکل زیر باشد پارامترها را نال بر میگرداند.

```
<script type="text/javascript">
$(function () {
    // var r = "12";
    var productsDataSource = new kendo.data.DataSource({
        transport: {
            read: {
                url: "@Url.Action('GetProducts', 'Home')",
                dataType: "json",
                contentType: 'application/json; charset=utf-8',
                type: 'GET',
                data: { param1: "dfvdf", param2: "val2" } // ارسال اطلاعات اضافی و سفارشی به
            },
            create: {
                url: "@Url.Action('PostProduct','Home')",
                contentType: 'application/json; charset=utf-8',
                type: "POST"
            },
            update: {
                url: // function (product) {
                //     "@Url.Action('UpdateProduct','Home')", //, +product.Id;
                // },
                contentType: 'application/json; charset=utf-8',
                type: "PUT"
            },
            destroy: {
                url: function (p) {
                    return "@Url.Action('DeleteProduct','Home')/" + p.Id;
                },
                contentType: 'application/json; charset=utf-8',
                type: "DELETE"
            },
            parameterMap: function (options) {
                return kendo.stringify(options);
            }
        },
        schema: {
            parse: function (data) {
                return data;
            },
            data: "Data",
            total: "Total",
            model: {
                id: "Id", // define the model of the data source. Required for validation and
                fields: {
                    "Id": { type: "number", editable: false }, // تعیین نوع فیلد برای جستجوی پویا
                    "Name": { type: "string", validation: { required: true }, editable: true },
                    "Discription": { type: "string", },
                    "Title": { type: "string", editable: false },
                    "GroupName": { type: "string", },
                    "Link": { type: "string" }
                }
            }
        },
        batch: false,

        error: function (e) {
            alert(e.errorThrown.stack);
        },
        pageSize: 5,
        sort: { field: "Id", dir: "desc" }
    });
    $("#report-grid").kendoGrid({
        dataSource: productsDataSource,
        autoBind: true,
        scrollable: false,
        pageable: true,
        sortable: true,
        columns: [
            { field: "Id", title: "#" },
            { field: "Name", title: "Product" }
        ]
    });
}
```

سرور در حین درخواست

مهم است

property types.


```
});
});
</script>
```

نویسنده: وحید نصیری
تاریخ: ۲۲:۲۷ ۱۳۹۴/۰۱/۰۴

کدهای ASP.NET MVC مطلب « [فعال سازی عملیات CRUD در Kendo UI Grid](#) » را جهت دریافت پارامتر سفارشی به روز کردم .
زمانیکه [صفحه بندی](#) فعال است، تمام پارامترها داخل یک کوئری استرینگ با فرمت جی-سون قرار می‌گیرند. به این شکل:

```
{"param1":"val1","param2":"val2","take":10,"skip":0,"page":1,"pageSize":10,"sort":[{"field":"Id","dir":"desc"}]}
```

برای خواندن آن‌ها فقط کافی است یک کلاس سفارشی ایجاد کرد:

```
// با ارث بری، خواص اضافی و سفارشی را به کلاس پایه اضافه می‌کنیم
public class CustomDataSourceRequest : DataSourceRequest
{
    public string Param1 { set; get; }
    public string Param2 { set; get; }
}
```

بعد بجای DataSourceRequest اصلی، از کلاس سفارشی حاوی پارامترهای اضافی استفاده خواهیم کرد:

```
var request = JsonConvert.DeserializeObject<CustomDataSourceRequest>(queryString);
```

نویسنده: امیر نوروزیان
تاریخ: ۱۱:۳۴ ۱۳۹۴/۰۲/۰۷

با سلام؛ من از api استفاده نمیکنم دقیقاً روش شما رو استفاده کردم برنامه رو هم دیباگ کردم اطلاعات درست به سمت سرور میره بر میگردد ولی نمیدونم چرا نشون نمیده و هیچ خطاییم نشون نمیده

The screenshot shows a web browser with a Kendo UI Grid titled "Product". The grid is empty, displaying "No items to display". The developer console is open, showing the "Net" panel with a list of network requests. The last request is "GET GridData?()", which returned a 200 OK status. The response is shown in the "Response" tab, displaying a JSON array of four objects, each representing a product with an Id and a Name.

| URL | Status | Domain | Size | Remote IP | Timeline |
|-------------------------------|------------------|-----------------|------|-------------|----------|
| GET Index | 200 OK | localhost:38019 | KB | [::1]:38019 | باعد |
| GET kendo.common.min.css | 200 OK | localhost:38019 | KB | [::1]:38019 | باعد |
| GET kendo.common.core.min.css | 200 OK | localhost:38019 | KB | [::1]:38019 | باعد |
| GET kendo.default.min.css | 200 OK | localhost:38019 | KB | [::1]:38019 | باعد |
| GET jquery.min.js | 304 Not Modified | localhost:38019 | KB | [::1]:38019 | باعد |
| GET kendo.web.min.js | 304 Not Modified | localhost:38019 | MB | [::1]:38019 | باعد |
| GET kendo.all.min.js | 304 Not Modified | localhost:38019 | MB | [::1]:38019 | باعد |
| GET sprite.png | 304 Not Modified | localhost:38019 | KB | [::1]:38019 | باعد |
| GET GridData?() | 200 OK | localhost:38019 | KB | [::1]:38019 | باعد |

The response for "GET GridData?()" is shown in the "Response" tab, displaying a JSON array of four objects, each representing a product with an Id and a Name.

```
Object { Id=1, Name="Product 1" }
Object { Id=2, Name="Product 2" }
Object { Id=3, Name="Product 3" }
Object { Id=4, Name="Product 4" }
```

نویسنده: وحید نصیری
تاریخ: ۱۲:۳۲ ۱۳۹۴/۰۲/۰۷

کدهای کامل بحث جاری را با فرمت ASP.NET MVC، [از اینجا](#) دریافت کنید و با کدهای خودتان انطباق دهید؛ با این [کنترلر](#) و با این

نویسنده: امیر نوروزیان
تاریخ: ۱۳۹۴/۰۲/۰۷ ۱۲:۵۷

ممنون
مشکلم حل شد فراموش کردن <body> بود

نویسنده: محمد رحیم پورابراهیم
تاریخ: ۱۳۹۴/۰۳/۰۷ ۱۴:۳۸

سلام
من طبق مقاله یه پروژه MVC رو ساختم اما زمانی که پروژه رو اجرا میکنم در خطی که

```
$("#report-grid").kendoGrid
```

رو تعریف کردم خطای زیر رو دریافت میکنم

```
$(...).kendoGrid is not a function
```

در layout به فایلهای css و js ارجاع داده شده و view که ساخته ام از layout استفاده میکنم.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۴/۰۳/۰۷ ۱۵:۳۴

همین سؤال [در نظرات مطلب معرفی Kendo Ui](#) بحث شده.