

بعد از استفاده از گریدهای JQGrid, Kendo و Grid.mvc مشکلاتی که با هر کدام از آنها داشتم، در نهایت به WebGrid که به صورت توکار وجود دارد، برای استفاده جهت نمایش اطلاعات رسیدم؛ از این جهت که به کتابخانه‌ی جانبی نیازی ندارد و از نظر سرعت و لود شدن بهینه می‌باشد، البته با اضافه کردن یکسری کدهای css. برای آشنایی بیشتر با این helper توصیه می‌کنم ابتدا این [مقاله](#) را مطالعه نمایید. به صورت پیش فرض WebGrid صفحه بندی را به صورت خیلی ساده فقط با نمایش اعداد و جهت نماهای جلو و عقب، نشان می‌دهد که برای پروژه‌های رسمی تا حدودی جالب نیست.

در این مطلب قصد داریم از کتابخانه‌ی bootstrap جهت صفحه بندی استفاده کنیم و در نهایت به صفحه بندی زیر برسیم:



برای اینکار، ابتدا قبل از هر چیزی به یک متد الحاقی برای انجام صفحه بندی سفارشی سازی شده، نیاز داریم که کدهای این متد به صورت زیر خواهد بود:

```
public static class WebGridExtensions
{
    public static HelperResult PagerList(
        this WebGrid webGrid,
        WebGridPagerModes mode = WebGridPagerModes.NextPrevious | WebGridPagerModes.Numeric,
        string firstText = null,
        string previousText = null,
        string nextText = null,
        string lastText = null,
        int numericLinksCount = 5)
    {
        return PagerList(webGrid, mode, firstText, previousText, nextText, lastText, numericLinksCount,
            explicitlyCalled: true);
    }

    private static HelperResult PagerList(
        WebGrid webGrid,
        WebGridPagerModes mode,
        string firstText,
        string previousText,
        string nextText,
        string lastText,
        int numericLinksCount,
        bool explicitlyCalled)
    {
        int currentPage = webGrid.PageIndex;
        int totalPages = webGrid.PageCount;
        int lastPage = totalPages - 1;

        var ul = new TagBuilder("ul");
        var li = new List<TagBuilder>();

        if (ModeEnabled(mode, WebGridPagerModes.FirstLast)) {
            if (String.IsNullOrEmpty(firstText)) {
                firstText = "اولین";
            }

            var part = new TagBuilder("li") {
                InnerHtml = GridLink(webGrid, webGrid.GetPageUrl(0), firstText)
            };
        }
    }
}
```

```

        if (currentPage == 0) {
            part.MergeAttribute("class", "disabled");
        }
        li.Add(part);
    }

    if (ModeEnabled(mode, WebGridPagerModes.NextPrevious)) {
        if (String.IsNullOrEmpty(previousText)) {
            previousText = "قبلی";
        }

        int page = currentPage == 0 ? 0 : currentPage - 1;

        var part = new TagBuilder("li") {
            InnerHtml = GridLink(webGrid, webGrid.GetPageUrl(page), previousText)
        };

        if (currentPage == 0) {
            part.MergeAttribute("class", "disabled");
        }

        li.Add(part);
    }

    if (ModeEnabled(mode, WebGridPagerModes.Numeric) && (totalPages > 1)) {
        int last = currentPage + (numericLinksCount / 2);
        int first = last - numericLinksCount + 1;
        if (last > lastPage) {
            first -= last - lastPage;
            last = lastPage;
        }
        if (first < 0) {
            last = Math.Min(last + (0 - first), lastPage);
            first = 0;
        }
        for (int i = first; i <= last; i++) {
            var pageText = (i + 1).ToString(CultureInfo.InvariantCulture);
            var part = new TagBuilder("li") {
                InnerHtml = GridLink(webGrid, webGrid.GetPageUrl(i), pageText)
            };

            if (i == currentPage) {
                part.MergeAttribute("class", "active");
            }

            li.Add(part);
        }
    }

    if (ModeEnabled(mode, WebGridPagerModes.NextPrevious)) {
        if (String.IsNullOrEmpty(nextText)) {
            nextText = "بعدی";
        }

        int page = currentPage == lastPage ? lastPage : currentPage + 1;

        var part = new TagBuilder("li") {
            InnerHtml = GridLink(webGrid, webGrid.GetPageUrl(page), nextText)
        };

        if (currentPage == lastPage) {
            part.MergeAttribute("class", "disabled");
        }

        li.Add(part);
    }

    if (ModeEnabled(mode, WebGridPagerModes.FirstLast)) {
        if (String.IsNullOrEmpty(lastText)) {
            lastText = "آخرین";
        }

        var part = new TagBuilder("li") {

```

```

        InnerHtml = GridLink(webGrid, webGrid.GetPageUrl(lastPage), lastText)
    };

    if (currentPage == lastPage) {
        part.MergeAttribute("class", "disabled");
    }

    li.Add(part);
}

ul.InnerHtml = string.Join("", li);

var html = "";
if (explicitlyCalled && webGrid.IsAjaxEnabled) {
    var span = new TagBuilder("span");
    span.MergeAttribute("data-swhgajax", "true");
    span.MergeAttribute("data-swhgcontainer", webGrid.AjaxUpdateContainerId);
    span.MergeAttribute("data-swhgcallback", webGrid.AjaxUpdateCallback);

    span.InnerHtml = ul.ToString();
    html = span.ToString();
} else {
    html = ul.ToString();
}

return new HelperResult(writer => {
    writer.Write(html);
});
}

private static String GridLink(WebGrid webGrid, string url, string text)
{
    TagBuilder builder = new TagBuilder("a");
    builder.SetInnerText(text);
    builder.MergeAttribute("href", url);
    if (webGrid.IsAjaxEnabled) {
        builder.MergeAttribute("data-swhglnk", "true");
    }
    return builder.ToString(TagRenderMode.Normal);
}

private static bool ModeEnabled(WebGridPagerModes mode, WebGridPagerModes modeCheck)
{
    return (mode & modeCheck) == modeCheck;
}
}

```

کلاس فوق باید در پوشه‌ی App_Code قرار گیرد.

پس از آن در View یی که اطلاعات را نمایش می‌دهید، فقط لازم است کد زیر را اضافه نمایید:

```

<div>
@grid.PagerList(mode: WebGridPagerModes.All)
</div>

```

تا اینجا متد مورد نظر برای انجام صفحه بندی گرید پیاده سازی شد. ادامه‌ی کار هم مشخص است؛ داشتن یک PartialView جهت نمایش لیست اطلاعات، پاس دادن دیتا به Partial و تمام.

در ادامه برای تکمیل بحث مثالی را از نحوه‌ی نمایش اطلاعات و صفحه بندی سفارشی نشان خواهیم داد:

PartialView لازم برای نمایش اطلاعات

تنظیمات لازم گرید :

```

@{
    WebGrid grid = new WebGrid(Model,
        rowsPerPage: 10,
        ajaxUpdateContainerId: "grid");
}

```

```
var rowIndex = ((grid.PageIndex + 1) * grid.RowsPerPage) - (grid.RowsPerPage - 1);
}
```

تعیین فیلدهای گرید :

```
@grid.Table(
    tableStyle: "table table-striped table-hover",
    headerStyle: "webgrid-header",
    alternatingRowStyle: "webgrid-alternating-row",
    selectedRowStyle: "webgrid-selected-row",
    rowStyle: "webgrid-row-style",
    columns: grid.Columns(
        grid.Column(columnName: "Name", header: "نام استان", style: "myfont"),
        grid.Column(columnName: "NameEn", header: "نام استان ( انگلیسی)", style: "myfont"),
        grid.Column(header: "", format: item => @Html.ActionLink("مدیریت شهرها", actionName:
MVC.Admin.City.ActionNames.Index, controllerName: MVC.Admin.City.Name, routeValues: new {Code=item.Code
}),htmlAttributes:null)),
        grid.Column(header: "",
            style: "text-align-center-col smallcell",
            format: item => @Html.ActionLink(linkText: "ویرایش", actionName: "Edit",
                controllerName: "Province", routeValues: new { area = "Admin", code =
item.Code },
                htmlAttributes: new { @class = "btn-sm btn-info vertical-center" })))
    ) )

<div>
    @grid.PagerList(mode: WebGridPagerModes.All)
</div>
```

خروجی حاصل به صورت زیر خواهد بود :

#	نام استان	نام استان (انگلیسی)		
۱	تهران	Tehan	مدیریت شهرها	ویرایش حذف
۲	کردستان	Kordistan	مدیریت شهرها	ویرایش حذف
۳	کرمانشاه	kermanshah	مدیریت شهرها	ویرایش حذف

اولین	قبلی	۱	۲	۳	۴	بعدی	آخرین
-------	------	---	---	---	---	------	-------

اگر طبق توضیحات بالا عمل کرده باشید، در نهایت صفحه بندی شما به صورت عمودی نمایش داده می شود؛ یعنی هر کدام از شماره صفحات در یک سطر. دلیل آن هم این است که تگ ul، کلاس pagination را ندارد. در کدهای بوت استراپ تعریف شده است که تمام li هایی که به صورت مستقیم داخل کلاس pagination هستند خصوصیات مورد نظر را بگیرند.

برای این کار دو راه حل وجود دارد :

راه حل اول: تغییر کدهای css

کدهای نوشته شده برای صفحه بندی در بوت استراپ را از حالت زیر:

```
.pagination > li
```

به حالت زیر تغییر دهید:

```
.pagination li
```

یادآوری: علامت > در CSS یعنی به صورت مستقیم و در شاخه‌ی اول.

راه حل دوم - افزودن کلاس .pagination به تگ ul:
ابتدا کلاس .pagination را از تگ div حذف نمایید:

```
<div >  
    @grid.PagedList(mode: WebGridPagerModes.All)  
</div>
```

و در کدهایی کلاس WebGridExtensions، در قسمتی که تگ ul اضافه می‌شود، کلاس مورد نظر را به آن اضافه می‌کنیم:

```
var ul = new TagBuilder("ul");  
ul.AddCssClass("pagination");
```

[دانلود کدهای این مثال](#)

نکته‌ای در مورد Webgrid

اگر نیاز داشتید به یکباره تمام اطلاعات را در گرید لود نکنید و به صورت n تا n رکوردها را نمایش دهید، در این حالت پس از پاس دادن لیستی از اطلاعات به View مورد نظر لازم است تعداد کل رکوردها را در یک متغیر به سمت View بفرستید. این کار به این دلیل می‌باشد که بتوان صفحه بندی را تولید کرد. برای این کار در بخش تنظیمات Webgrid مقدار source را برابر null قرار دهید و از قطعه کد زیر جهت بایند کردن گرید، بعد از کدهای تنظیمات WebGrid استفاده نمایید:

```
grid.Bind(Model, rowCount: (int)ViewBag.PageCount);
```

نظرات خوانندگان

نویسنده: امیر نوروزیان
تاریخ: ۱۴:۲۷ ۱۳۹۴/۰۸/۰۳

با سلام و تشکر از شما
هنگام استفاده از custom paging صفحه بندی مربوط به خود webgrid رو هم نشون میده که با غیر فعال کردن آن
canpage=false کلا صفحه بندی غیر فعال می‌شود.

<54321>

اولین	قبلی	2	3	4	بعدی	آخرین
-------	------	---	---	---	------	-------

نویسنده: عثمان رحیمی
تاریخ: ۱۵:۴۳ ۱۳۹۴/۰۸/۰۳

سلام .
لطفا از تابع Table به جای GetHtml استفاده کنید و خط زیر را هم از آن حذف کنید :

```
mode: WebGridPagerModes.All,
```

نمونه :

```
@{
    var grid = new WebGrid(
        source: null,
        canPage: true,
        rowsPerPage: 10,
        canSort: true,
        defaultSort: "Title"
    );
    grid.Bind(Model, rowCount: (int)ViewBag.PageCount, autoSortAndPage: false);
    var rowIndex = ((grid.PageIndex + 1) * grid.RowsPerPage) - (grid.RowsPerPage - 1);
}
@grid.Table(
    tableStyle: "table table-striped table-hover",
    headerStyle: "webgrid-header",
    footerStyle: "webgrid-footer",
    alternatingRowStyle: "webgrid-alternating-row",
    selectedRowStyle: "webgrid-selected-row",
    rowStyle: "webgrid-row-style",

    columns: grid.Columns(
        grid.Column(header: "#",
            style: "text-align-center-col",
            format: @<text>@(rowIndex++)</text>),
        grid.Column(columnName: "Title", header: "عنوان", style: "myfont"),
        grid.Column(columnName: "URL", header: "آدرس", style: "myfont"),

        grid.Column(header: "",
            style: "text-align-center-col smallcell",
            format: item => @Html.ActionLink(linkText: "ویرایش", actionName: "Edit",
                controllerName: "Link", routeValues: new { area = "Admin", id = item.Code
```

```

},
        htmlAttributes: new { @class = "btn-sm btn-info vertical-center" })),
    grid.Column(header: "",
        format: @<form action="Link/Delete/@item.Code" method="post">
            <input type="submit" class="btn-sm btn-danger submitlink"
                onclick="return confirm('آیا از حذف این آیتم مطمئن هستید ؟');"
                value="حذف" />
        </form>)))
<div class="text-center">
    @grid.PagedList(mode: WebGridPagerModes.All)
</div>

```

اگر شما تمام دیتا را به یکباره می‌خواهید نمایش دهید باید به صورت زیر تغییر دهید :

```

@{
    var grid = new WebGrid(
        source: Model,
        canPage: true,
        rowsPerPage: 10,
        canSort: true,
        defaultSort: "Title"
    );
    // grid.Bind(Model, rowCount: (int)ViewBag.PageCount, autoSortAndPage: false); // Delete this Line
    var rowIndex = ((grid.PageIndex + 1) * grid.RowsPerPage) - (grid.RowsPerPage - 1);
}

```