

صورت مساله:

- می‌خواهیم footer پیش فرض PdfReport را که تاریخ را در یک سمت، و شماره صفحه را در سمتی دیگر نمایش می‌دهد، به عبارت «صفحه x از n» تغییر دهیم.

- می‌خواهیم در Header گزارش بجای Header پیش فرض PdfReport یکی از قالب‌های PDF تهیه شده توسط [Open Office](#) را نمایش دهیم (و یا هر ساختار دیگری را).

تمام اجزای PdfReport جهت امکان اعمال تغییرات کلی و توسعه آن‌ها طراحی شده‌اند؛ قالب‌ها، هدر، فوتر، منابع داده، قالب‌های نمایش سلول‌ها، تعریف توابع تجمعی سفرشی و غیره. جهت سهولت کار، به ازای هر یک از این موارد، پیاده سازی‌های پیش فرضی در PdfReport قرار دارند، امکان اگر مورد رضایت شما نیستند ... از بنیان تغییرشان دهید! (و همچنین اگر مورد جالبی را پیاده سازی کردید، می‌توانید به عنوان یک وصله جدید ارائه دهید تا به پروژه اضافه شود) ضمناً این مطالب سفرشی سازی نیاز به آشنایی با ساختار iTextSharp را نیز دارند؛ در حد ایجاد یک جدول ساده باید با iTextSharp [آشنا باشید](#).

مدل‌های مورد استفاده:

```
namespace PdfReportSamples.Models
{
    public class Task
    {
        public int Id { set; get; }
        public string Name { set; get; }
        public int PercentCompleted { set; get; }
        public bool IsActive { set; get; }
        public User Assignee { set; get; }
    }
}
```

```
using System;

namespace PdfReportSamples.Models
{
    public class User
    {
        public int Id { set; get; }
        public string Name { set; get; }
        public string LastName { set; get; }
        public long Balance { set; get; }
        public DateTime RegisterDate { set; get; }
    }
}
```

توسط این مدل‌ها قصد داریم تعدادی فعالیت (Task) را که به تعدادی کاربر انتساب یافته است، نمایش دهیم. همچنین نمایش مقادیر خواص تو در تو نیز در اینجا مد نظر است؛ برای مثال ستونی مانند این:

```
column.PropertyName<Task>(x => x.Assignee.Name)
```

کدهای کامل مثال را در ادامه ملاحظه خواهید نمود:

```
using System;
using System.Collections.Generic;
using System.Drawing;
using PdfReportSamples.Models;
using PdfRpt.Core.Contracts;
```

```

using PdfRpt.FluentInterface;

namespace PdfReportSamples.CustomHeaderFooter
{
    public class CustomHeaderFooterPdfReport
    {
        readonly CustomHeader _customHeader = new CustomHeader();
        public IPdfReportData CreatePdfReport()
        {
            return new PdfReport().DocumentPreferences(doc =>
            {
                doc.RunDirection(PdfRunDirection.LeftToRight);
                doc.Orientation(PageOrientation.Portrait);
                doc.PageSize(PdfPageSize.A4);
                doc.DocumentMetadata(new DocumentMetadata { Author = "Vahid", Application = "PdfRpt",
Keywords = "Test", Subject = "Test Rpt", Title = "Test" });
            })
            .DefaultFonts(fonts =>
            {
                fonts.Path(Environment.GetEnvironmentVariable("SystemRoot") + "\\fonts\\tahoma.ttf",
Environment.GetEnvironmentVariable("SystemRoot") +
"\\fonts\\verdana.ttf");
            })
            .PagesFooter(header =>
            {
                footer.CustomFooter(new CustomFooter(footer.PdfFont, PdfRunDirection.LeftToRight));
            })
            .PagesHeader(header =>
            {
                header.CustomHeader(_customHeader);
            })
            .MainTableTemplate(template =>
            {
                template.BasicTemplate(BasicTemplate.SilverTemplate);
            })
            .MainTablePreferences(table =>
            {
                table.ColumnsWidthsType(TableColumnWidthType.Relative);
                table.MultipleColumnsPerPage(new MultipleColumnsPerPage
                {
                    ColumnsGap = 22,
                    ColumnsPerPage = 2,
                    ColumnsWidth = 250,
                    IsRightToLeft = false,
                    TopMargin = 7
                });
            })
            .MainTableDataSource(dataSource =>
            {
                var rows = new List<Task>();
                var rnd = new Random();
                for (int i = 1; i < 210; i++)
                {
                    rows.Add(new Task
                    {
                        Assignee = new User
                        {
                            Id = i,
                            Name = "user-" + i
                        },
                        IsActive = rnd.Next(0, 2) == 1 ? true : false,
                        Name = "task-" + i
                    });
                }
                dataSource.StronglyTypedList(rows);
            })
            .MainTableColumns(columns =>
            {
                columns.AddColumn(column =>
                {
                    column.PropertyName("rowNo");
                    column.IsRowNumber(true);
                    column.CellsHorizontalAlignment(HorizontalAlignment.Center);
                    column.IsVisible(true);
                    column.Order(0);
                    column.Width(1);
                    column.HeaderCell("#");
                });

                columns.AddColumn(column =>
                {

```

```

        column.PropertyName<Task>(x => x.Name);
        column.CellsHorizontalAlignment(HorizontalAlignment.Center);
        column.IsVisible(true);
        column.Order(1);
        column.Width(3);
        column.HeaderCell("Task Name");
    });

    columns.AddColumn(column =>
    {
        column.PropertyName<Task>(x => x.Assignee.Name); // nested property support
        column.CellsHorizontalAlignment(HorizontalAlignment.Center);
        column.IsVisible(true);
        column.Order(2);
        column.Width(3);
        column.HeaderCell("Assignee");
    });

    columns.AddColumn(column =>
    {
        column.PropertyName<Task>(x => x.IsActive);
        column.CellsHorizontalAlignment(HorizontalAlignment.Center);
        column.IsVisible(true);
        column.Order(3);
        column.Width(2);
        column.HeaderCell("Active");
        column.ColumnItemsTemplate(template =>
        {
            template.Checkmark(checkmarkFillColor: Color.Green, crossSignFillColor:
Color.DarkRed);
        });
    });
    })
    .MainTableEvents(events =>
    {
        events.DataSourceIsEmpty(message: "There is no data available to display.");
    })
    .Export(export =>
    {
        export.ToExcel();
    })
    .Generate(data => data.AsPdfFile(AppPath.ApplicationPath +
"\\Pdf\\CustomHeaderFooterPdfReportSample.pdf"));
    }
}

```

به همراه Header سفرارشی:

```

using System.Collections.Generic;
using iTextSharp.text;
using iTextSharp.text.pdf;
using PdfRpt.Core.Contracts;
using PdfRpt.Core.Helper;

namespace PdfReportSamples.CustomHeaderFooter
{
    public class CustomHeader : IPageHeader
    {
        public PdfPTable RenderingGroupHeader(Document pdfDoc, PdfWriter pdfWriter, IList<CellData>
rowdata, IList<SummaryCellData> summaryData)
        {
            return null;
        }

        Image _image;
        public PdfPTable RenderingReportHeader(Document pdfDoc, PdfWriter pdfWriter,
IList<SummaryCellData> summaryData)
        {
            if (_image == null) //cache is empty
            {
                var templatePath = AppPath.ApplicationPath + "\\data\\PdfHeaderTemplate.pdf";
                _image = PdfImageHelper.GetITextSharpImageFromPdfTemplate(pdfWriter, templatePath);
            }

            var table = new PdfPTable(1);
            var cell = new PdfPCell(_image, true) { Border = 0 };
            table.AddCell(cell);
        }
    }
}

```

```

        return table;
    }
}

```

و Footer سفرارشی استفاده شده:

```

using System.Collections.Generic;
using iTextSharp.text;
using iTextSharp.text.pdf;
using PdfRpt.Core.Contracts;

namespace PdfReportSamples.CustomHeaderFooter
{
    public class CustomFooter : IPageFooter
    {
        PdfContentByte _pdfContentByte;
        readonly IPdfFont _pdfRptFont;
        readonly Font _font;
        readonly PdfRunDirection _direction;
        PdfTemplate _template;

        public CustomFooter(IPdfFont pdfRptFont, PdfRunDirection direction)
        {
            _direction = direction;
            _pdfRptFont = pdfRptFont;
            _font = _pdfRptFont.Fonts[0];
        }

        public void ClosingDocument(PdfWriter writer, Document document, IList<SummaryCellData>
columnCellsSummaryData)
        {
            _template.BeginText();
            _template.SetFontAndSize(_pdfRptFont.Fonts[0].BaseFont, 8);
            _template.SetTextMatrix(0, 0);
            _template.ShowText((writer.PageNumber - 1).ToString());
            _template.EndText();
        }

        public void PageFinished(PdfWriter writer, Document document, IList<SummaryCellData>
columnCellsSummaryData)
        {
            var pageSize = document.PageSize;
            var text = "Page " + writer.PageNumber + " / ";
            var textLen = _font.BaseFont.GetWidthPoint(text, _font.Size);
            var center = (pageSize.Left + pageSize.Right) / 2;
            var align = _direction == PdfRunDirection.RightToLeft ? Element.ALIGN_RIGHT :
Element.ALIGN_LEFT;

            ColumnText.ShowTextAligned(
                canvas: _pdfContentByte,
                alignment: align,
                phrase: new Phrase(text, _font),
                x: center,
                y: pageSize.GetBottom(25),
                rotation: 0,
                runDirection: (int)_direction,
                arabicOptions: 0);

            var x = _direction == PdfRunDirection.RightToLeft ? center - textLen : center + textLen;
            _pdfContentByte.AddTemplate(_template, x, pageSize.GetBottom(25));
        }

        public void DocumentOpened(PdfWriter writer, IList<SummaryCellData> columnCellsSummaryData)
        {
            _pdfContentByte = writer.DirectContent;
            _template = _pdfContentByte.CreateTemplate(50, 50);
        }
    }
}

```

البته لازم به ذکر است که تمام این کدها به پوشه Samples سورس پروژه نیز جهت سهولت دسترسی، [اضافه شده‌اند](#).

توضیحات:

برای پیاده سازی Header و Footer سفارشی در PdfReport نیاز خواهید داشت تا دو اینترفیس IPageHeader و IPageFooter را پیاده سازی کنید.

ساختار IPageHeader را در ذیل ملاحظه می کنید:

```
using System.Collections.Generic;
using iTextSharp.text;
using iTextSharp.text.pdf;

namespace PdfRpt.Core.Contracts
{
    public interface IPageHeader
    {
        PdfPTable RenderingGroupHeader(Document pdfDoc, PdfWriter pdfWriter, IList<CellData>
newGroupInfo, IList<SummaryCellData> summaryData);

        PdfPTable RenderingReportHeader(Document pdfDoc, PdfWriter pdfWriter, IList<SummaryCellData>
summaryData);
    }
}
```

RenderingGroupHeader مرتبط است به مباحث گروه بندی اطلاعات و گزارشات master-detail که در قسمت های بعد به آنها اشاره خواهد شد. چون در اینجا به آن نیازی نداشتیم، تنها کافی است متد متناظر با آن، null بر گرداند که در کلاس CustomHeader فوق قابل مشاهده است.

متد RenderingReportHeader به ازای تولید هر صفحه جدید، فراخوانی خواهد شد. به عبارتی می توانید در صفحات مختلف، هدرهای مختلفی را نمایش دهید.

خروجی هر دو متد در اینجا یک جدول از نوع PdfPTable است. بنابراین هر نوع ساختار دلخواهی را که علاقمند هستید به شکل یک PdfPTable ایجاد کرده و بازگشت دهید. این جدول در هدر صفحات ظاهر خواهد شد.

برای نمونه در کلاس CustomHeader، یک قالب تهیه شده توسط Open Office توسط متد توکار PdfImageHelper.GetITextSharpImageFromPdfTemplate دریافت و تبدیل به تصویر می شود. این تصویر از نوع تصاویر قابل درک توسط iTextSharp است و نه اینکه واقعا تبدیل به یک تصویر معمولی مثلا از نوع bmp شود. سپس این تصویر، در یک ردیف از جدولی قرار داده شده و این جدول بازگشت داده می شود.

در کل یا توسط کار با PdfPTable می توانید یک هدر غیرپیش فرض را طراحی کنید و یا می توانید توسط ابزارهای بصری مانند Open Office یک قالب خاص را برای آن تهیه کرده و به روشی که ذکر شد و کدهای آنرا ملاحظه می کنید، بارگذاری و استفاده کنید. این قالب ها در مسیر Bin\Data سورس های پروژه قرار داده شده اند.

ساختار IPageFooter به صورت زیر است:

```
using iTextSharp.text;
using iTextSharp.text.pdf;
using System.Collections.Generic;

namespace PdfRpt.Core.Contracts
{
    public interface IPageFooter
    {
        void DocumentOpened(PdfWriter writer, IList<SummaryCellData> columnCellsSummaryData);

        void PageFinished(PdfWriter writer, Document document, IList<SummaryCellData>
columnCellsSummaryData);

        void ClosingDocument(PdfWriter writer, Document document, IList<SummaryCellData>
columnCellsSummaryData);
    }
}
```

برای طراحی یک Footer سفارشی کافی است اینترفیس فوق را پیاده سازی کنید که نمونه ای از آنرا در کدهای کلاس CustomFooter ملاحظه می نمائید.

متد DocumentOpened، با وهله سازی شیء Document فراخوانی می شود.

متد PageFinished هر بار پیش از اتمام کار صفحه جاری و افزوده شدن آن به Document فراخوانی می‌گردد. متد ClosingDocument، در زمان بسته شدن شیء Document فراخوانی خواهد شد.

اگر به امضای این متدها دقت کنید، شیء PdfWriter در اختیار شما قرار گرفته است که توسط آن می‌توان مستقیماً بر روی فایل PDF، محتوایی را قرار داد. شیء Document نیز در دسترس است. مثلاً توسط آن می‌توان اندازه دقیق صفحه را بدست آورد. به علاوه پارامتر columnCellsSummaryData نیز امکان دسترسی به مقادیر ردیف‌های قبلی را در اختیار شما قرار می‌دهد. برای مثال اگر نیاز دارید تا بر اساس مقادیر ستون‌ها و ردیف‌های قبلی، محاسباتی را انجام داده و در پایین صفحات درج کنید، به این ترتیب دسترسی کاملی به آن‌ها، خواهید داشت.

استفاده از این کلاس‌های سفرارشی نیز همواره به شکل زیر خواهد بود:

```
readonly CustomHeader _customHeader = new CustomHeader();
//...
.PagesFooter(footer =>
{
    footer.CustomFooter(new CustomFooter(footer.PdfFont, PdfRunDirection.LeftToRight));
})
.PagesHeader(header =>
{
    header.CustomHeader(_customHeader);
})
```

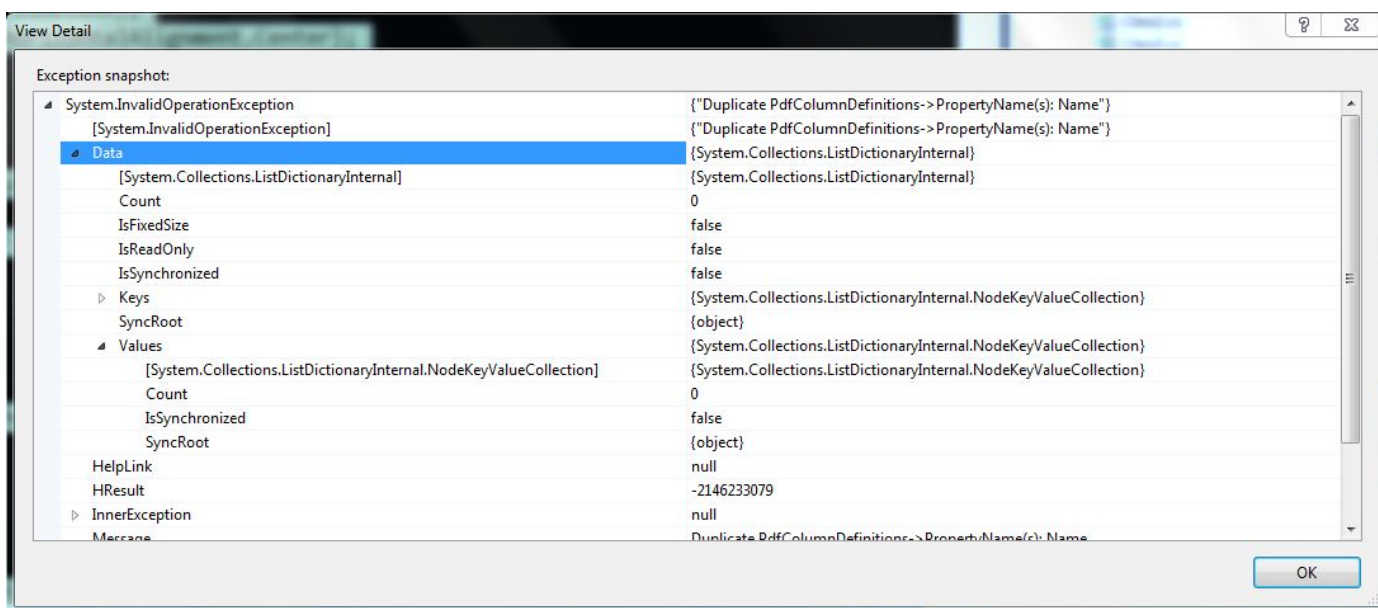
کلاً در PdfReport هر جایی متدی به نام CustomXYZ را مشاهده کردید، این متد یک اینترفیس را دریافت می‌کند. به عبارتی این امکان را خواهید داشت تا از متدهای پیش فرض مهیا صرف‌نظر کرده و مطابق نیاز، نسبت به پیاده سازی و استفاده از وهله جدیدی از این اینترفیس تعریف شده، اقدام کنید.

نظرات خوانندگان

نویسنده: mohsen

تاریخ: ۱۳۹۱/۰۷/۲۷ ۰:۴۸

با سلام؛ من برای پیاده سازی این مثال ابتدا خواستم که دقیقاً مثالی که شما اینجا ذکر کردید را بنویسم ولی متاسفانه به خطای زیر برخورددم ممنون میشم اگر راهنماییم کنید



نویسنده: وحید نصیری

تاریخ: ۱۳۹۱/۰۷/۲۷ ۱:۳

این مورد رو اخیرا اضافه کردم. لطفاً [نگارش 1.2](#) رو دریافت کنید تا خواص تو در تو را بدون مشکل بتوانید استفاده کنید. همچنین بهتر است از [NuGet استفاده کنید](#) تا از به روز رسانی‌ها بهتر مطلع شوید.

نویسنده: پویا امینی

تاریخ: ۱۳۹۱/۰۷/۲۹ ۰:۴۰

با سلام خدمت جناب نصیری، ببخشید شما فرمودید

می‌خواهیم در Header گزارش بجای Header پیش فرض PdfReport یکی از قالب‌های PDF تهیه شده توسط [Open Office](#) را نمایش دهیم (و یا هر ساختار دیگری را).

ولی در مثالی که در اینجا زدید در قسمت Header یک جدول ایجاد کردید حال اگر من بخواهم واقعاً از فایلی که با استفاده از OpenOffice ایجاد کرده‌ام و با استفاده از این روش مقدار TextBox‌های اون رو پر کرده‌ام استفاده کنم باید چه تغییری ایجاد کنم. ممنونم

نویسنده: وحید نصیری

تاریخ: ۱۳۹۱/۰۷/۲۹ ۰:۵۱

خروجی نهایی متد `public PdfTable RenderingReportHeader` یک جدول است. به همین جهت تعریف یک جدول ساده رو مشاهده کردید (که داخل آن این قالب قرار گرفته). اما فایل [PdfHeaderTemplate.pdf](#) ذکر شده در آن، واقعا یک فایل قالب Open Office است. فایل odt آن هم در پوشه [Bin/Data](#) سورسها موجود است.

نویسنده: پویا امینی
تاریخ: ۱۳۹۱/۰۷/۲۹ ۱:۳۱

جناب نصیری من زمانی که فایل [25817](#) رو از این [آدرس](#) دانلود می‌کنم و کل Solution رو اجرا می‌کنم بهم خطای زیر رو میده

Error List			
7 Errors 0 Warnings 0 Messages			
Description	File	L	C
2 Metadata file 'C:\Users\Administrator\Desktop\vhaid\Samples\PdfReportSamples\bin\Debug\PdfReportSamples.dll' could not be found			WebAppTests
6 Metadata file 'C:\Users\Administrator\Desktop\vhaid\Samples\PdfReportSamples\bin\Debug\PdfReportSamples.dll' could not be found	CSC		DemosBrowser
1 Metadata file 'C:\Users\Administrator\Desktop\vhaid\Lib\bin\Debug\PdfRpt.dll' could not be found			PdfReportSamples
3 Metadata file 'C:\Users\Administrator\Desktop\vhaid\Lib\bin\Debug\PdfRpt.dll' could not be found			WebAppTests
7 Metadata file 'C:\Users\Administrator\Desktop\vhaid\Lib\bin\Debug\PdfRpt.dll' could not be found	CSC		DemosBrowser
5 Importing key file "key.pfx" was canceled.			PdfRpt
4 Cannot import the following key file: key.pfx. The key file may be password protected. To correct this, try to import the certificate again or manually install the certificate to the Strong Name CSP with the following key container name: VS_KEY_5705A10B541BBA03			PdfRpt

و وقتی که از پوشه Samples فایل مربوط به WebApp رو باز می‌کنم باز هم قادر به اجرای اون نیستم.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۰۷/۲۹ ۹:۳۰

[اینجا پاسخ دادم](#) به این سؤال

نویسنده: a.g
تاریخ: ۱۳۹۲/۰۳/۱۵ ۱۵:۱۵

سلام
من نیاز دارم تا در Page Header و Group Header از قالب تهیه شده توسط Open Office استفاده کنم. قالب هایی که تهیه شدن، یه سری فیلد دارن که موقع ساخت گزارش باید پر بشن.
چطور باید فیلدهای موجود در قالب رو بعد از لود مقدار دهی کرد؟

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۳/۱۵ ۱۶:۵۴

در مورد جزئیات نحوه مقدار دهی فیلدهای این نوع قالبها مراجعه کنید به مطلب « [ساخت یک گزارش ساز به کمک iTextSharp](#) و [Open Office](#) ».

بعد از آشنایی، متد `GetITextSharpImageFromAcroForm` تعریف شده در PdfReport هم راه ساده‌تر پر کردن این نوع فیلدها است.

```
public static iTextSharp.text.Image GetITextSharpImageFromAcroForm(
    this PdfWriter pdfWriter,
    string pdfTemplateFilePath,
    IList<CellData> data,
    Action<IList<CellData>, AcroFields, PdfStamper> onFillAcroForm,
    IList<iTextSharp.text.Font> fonts,
```



```
int pageNumber = 1)
```

یک چنین امضایی داره تعریف شده در فضای نام [PdfRpt.Core.Helper](#).

نویسنده: عباس قربانی
تاریخ: ۲۳:۴۴ ۱۳۹۲/۰۳/۱۵

ممنون، واقعا لطف کردید

نویسنده: ali
تاریخ: ۱۸:۲۲ ۱۳۹۲/۰۳/۲۵

سلام

اول باید تشکر کنم بابت این ابزار که زحمتشو کشیدید.
میخواستم ببینم چطور میشه به AcroForm رو داخل footer گذاشت؟

نویسنده: وحید نصیری
تاریخ: ۱۸:۳۹ ۱۳۹۲/۰۳/۲۵

[در مثال هدر و فوتر سفارشی](#) یک نمونه استفاده از AcroForm به عنوان header هست. نکته مهم آن نحوه بازگشت این قالب به فرمت تصویر برداری قابل استفاده در iTextSharph است. سپس [در مثال InlineProviders](#) یک روش ساده تر افزودن محتویات دلخواه به فوتر صفحه معرفی شده در متد `inlineFooter.AddPageFooter`.

نویسنده: مهرداد
تاریخ: ۰:۲۷ ۱۳۹۲/۰۵/۰۵

با تشکر از شما
میشه نمونه دیگری برای استفاده سفارشی از فوتر هم بگذارید
چون مثلا من میخوام چند جمله در صفحه آخر فقط نمایش بدم
مثل امضا مدیر و امضا کاربر
ولی هر روشی انجام میدم باز جواب نمیده بهم و این کار را انجام نمیده
باید چه کاری بکنم
متشکرم

نویسنده: وحید نصیری
تاریخ: ۰:۵۹ ۱۳۹۲/۰۵/۰۵

[از رخدادهای باید استفاده کنید](#).

نویسنده: مهرداد
تاریخ: ۱:۲ ۱۳۹۲/۰۵/۰۵

متشکرم
و یک سوال دیگه
من در قسمت هدر سایت گرید و تبیل طراحی میکنم
میخوام برادر هیچ کدوم از گریدها نداشته باشه اما گرید اصلی میشه
اما گریدهای داخلی نمیشه؟!
آیا باید کار دیگه ای کنم؟!

نویسنده: وحید نصیری
تاریخ: ۱۳۹۲/۰۵/۰۵

- برای پرسش و پاسخ‌های متفرقه در مورد این کتابخانه لطفاً از [قسمت مخصوص آن در سایت](#) استفاده کنید.
- یک قسمت به [طراحی رنگ گرید اصلی](#) اختصاص دارد.
- پس از آن می‌تونید [قالب شفاف](#) هم مثلاً ایجاد کنید (یا هر حالت دلخواه دیگری). روش استفاده:

```
.MainTableTemplate(template =>
{
    template.CustomTemplate(new TransparentTemplate());
})
```