

مدتی هست که در حال تهیه کتابخانه‌ی گزارش سازی مبتنی بر iTextSharp هستم و عمده‌ی استفاده از آن برای من تاکنون، تهیه گزارشات باکیفیت PDF فارسی تحت وب بوده؛ هر چند این کتابخانه وابستگی خاصی به فناوری مورد استفاده ندارد و با WinForms، WPF، مشتقات ASP.NET، سرویس‌های WCF و کلا هر جایی که دات نت فریم ورک کامل در دسترس باشد، قابل استفاده است. همچنین به منبع داده خاصی گره نخورده است و حتی می‌توانید از یک anonymously typed list عدم متصل به بانک اطلاعاتی خاصی نیز به عنوان منبع داده آن استفاده کنید.

کتابخانه PdfReport به عمد جهت دات نت 3.5 تهیه شده است تا بازه وسیعی از سیستم عامل‌ها را پوشش دهد. این کتابخانه علاوه بر تبدیل اطلاعات شما به گزارشات مبتنی بر PDF، امکان تهیه خروجی خودکار اکسل (2007 به بعد) را نیز دارد. فایل خروجی آن، به صورت پیوست درون فایل PDF تهیه شده قرار می‌گیرد و جزئی از آن می‌شود. بدیهی است اینبار با کتابخانه گزارش سازی روبرو هستید که با راست به چپ مشکلی ندارد! کتابخانه PdfReport بر پایه کتابخانه‌های معروف سورس باز iTextSharp و EPPlus تهیه شده است. حداقل مزیت استفاده از آن، صرفه جویی در وقت شما جهت آموختن ریزه کاری‌های مرتبط با هر کدام از کتابخانه‌های یاده شده است. برای نمونه جهت فراگیری کار با iTextSharp نیاز است یک کتاب 600 صفحه‌ای به نام iText in action را مطالعه و تمرین کنید. این مورد منهای مسایل و نکات متعدد مرتبط با زبان فارسی است که در این کتاب به آن‌ها اشاره‌ای نشده است. برای تهیه آن، مشکلات متداولی که کاربران مدام در انجمن‌های برنامه نویسی مطرح می‌کنند و ابزارهای موجود عاجز از ارائه راه حلی ساده برای حل آن‌ها هستند، مد نظر بوده و امید است نگارش یک این کتابخانه بتواند بسیاری از این دردها را کاهش دهد. کار با این کتابخانه صرفا با کدنویسی میسر است (code first) و همین مساله انعطاف پذیری قابل توجهی را ایجاد کرده که در طی روزهای آینده با جزئیات بیشتر آن آشنا خواهید شد.

اما چرا PDF؟

استفاده از قالب PDF برای تهیه گزارشات، این مزایا را به همراه دارد:

- دقیقا همان چیزی که مشاهده می‌شود، در هر مکانی قابل چاپ است. با همان کیفیت، همان اندازه صفحه، همان فونت و غیره. به این ترتیب به صفحه بندی بسیار مناسب و بهینه‌ای می‌توان رسید و مشکلات گزارشات HTML ایی وب را ندارد.
- امکان استفاد از فونت‌های شکیل‌تر در آن بدون مشکل و بدون نیاز به نصب بر روی کامپیوتری میسر است؛ چون فونت را می‌توان در فایل PDF نیز قرار داد.
- این فایل در تمام سیستم عامل‌ها پشتیبانی می‌شود. خصوصا اینکه فایل نهایی در تمام کامپیوترها و در انواع و اقسام سیستم عامل‌ها به یک شکل و اندازه نمایش داده خواهد شد. برای مثال اینطور نیست که در ویندوز XP، اعداد آن فارسی نمایش داده شوند و در ویندوز 7 با تنظیمات محلی خاصی، دیگر اینطور نباشد. حتی تحت لینوکس هم اعداد آن فارسی نمایش داده خواهد شد چون فونت مخصوص بکار رفته، در خود فایل PDF قابل قرار دادن است.
- برنامه معروف و رایگان Adobe reader برای خواندن و مشاهده آن کفایت می‌کند و البته کلاینت یکبار باید این برنامه را نصب کند. همچنین از این نوع برنامه‌های رایگان برای مشاهده فایل‌های PDF زیاد است.
- تمام صفحات گزارش را در یک فایل می‌توان داشت و به یکباره تمام آن نیز به سادگی قابل چاپ است. این مشکلی است که با گزارشات تحت وب وجود دارد که نمی‌شود مثلا یک گزارش 100 صفحه‌ای را به یکباره به چاپگر ارسال کرد. به همین جهت عموما کاربران درخواست می‌دهند تا کل گزارش را در یک صفحه HTML نمایش دهید تا ما راحت آنرا چاپ کنیم یا راحت از آن خروجی بگیریم. اما زمانیکه فایل PDF تهیه می‌شود این مشکلات وجود نخواهد داشت و جهت Print بسیار بهینه سازی شده است. تا حدی که الان فرمت برگزیده تهیه کتاب‌های الکترونیکی نیز PDF است. مثلا سایت معروف آمازون امکان فروش نسخه PDF کتاب‌ها را هم پیش بینی کرده است.
- امکان صفحه بندی دقیق به همراه مشخص سازی landscape یا portrait بودن صفحه نهایی میسر است. چیزی که در گزارشات صفحات وب آنچنان معنایی ندارد.
- امکان رمزنگاری اطلاعات در آن پیش بینی شده است. همچنین می‌توان به فایل‌های PDF امضای دیجیتال نیز اضافه کرد. به این ترتیب هرگونه تغییری در محتوای فایل توسط برنامه‌های PDF خوان معتبر گزارش داده شده و می‌توان از صحت اطلاعات ارائه

شده توسط آن اطمینان حاصل کرد.

- از فشرده سازی مطالب، فایل ها و تصاویر قرار داده شده در آن پشتیبانی می کند.

- از گرافیک برداری پشتیبانی می کند.

مجوز استفاده از این کتابخانه:

کار من مبتنی بر LGPL است. به این معنا که به صورت باینری (فایل dll) در هر نوع پروژه ای قابل استفاده است.

اما ... PdfReport از دو کتابخانه دیگر نیز استفاده می کند:

- کتابخانه iTextSharp که دارای مجوز AGPL است. این مجوز رایگان نیست.

- کتابخانه EPPPlus برای تولید فایل های اکسل با کیفیت. مجوز استفاده از این کتابخانه LGPL است و تا زمانیکه به صورت باینری از آن استفاده می کنید، محدودیتی را برای شما ایجاد نخواهد کرد.

کتابخانه PdfReport به صورت سورس باز در CodePlex [قرار گرفته](#) ؛ اما جهت پرسیدن سؤالات، پیشنهادات، ارائه بهبود و غیره می توانید (و بهتر است) از قسمت [مدیریت پروژه مرتبط در سایت جاری](#) نیز استفاده کنید.

نحوه تهیه اولین گزارش، با کتابخانه PdfReport

الف) یک پروژه Class library جدید را شروع کنید. از این جهت که گزارشات PdfReport در انواع و اقسام پروژه های VS.NET قابل استفاده است، می توان از این پروژه Class library به عنوان کلاس های پایه قابل استفاده در انواع و اقسام پروژه های مختلف، بدون نیاز به تغییری در کدهای آن استفاده کرد.

ب) آخرین نگارش فایل های مرتبط با PdfReport را [از اینجا](#) دریافت کنید و سپس ارجاعاتی را به اسمبلی های موجود در بسته آن به پروژه خود اضافه نمایید (ارجاعاتی به PdfReport، iTextSharp و EPPPlus). فایل XML راهنمای کتابخانه نیز به همراه بسته آن می باشد که در حین استفاده از متدها و خواص PdfReport کمک بزرگی خواهد بود.

ج) کلاس های زیر را به آن اضافه کنید:

```
using System.Web;
using System.Windows.Forms;

namespace PdfReportSamples
{
    public static class AppPath
    {
        public static string ApplicationPath
        {
            get
            {
                if (isInWeb)
                    return HttpRuntime.AppDomainAppPath;

                return Application.StartupPath;
            }
        }

        private static bool isInWeb
        {
            get
            {
                return HttpContext.Current != null;
            }
        }
    }
}
```

از این کلاس برای مشخص سازی محل ذخیره سازی فایل های نهایی PDF تولیدی استفاده خواهیم کرد. همانطور که مشاهده می کنید ارجاعاتی را به System.Web.dll و System.Windows.Forms.dll نیاز دارد.

در ادامه کلاس User را جهت ساخت یک منبع داده درون حافظه‌ای تعریف خواهیم کرد:

```
using System;

namespace PdfReportSamples.IList
{
    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; }
    }
}
```

اکنون کلاس اصلی گزارش ما به صورت زیر خواهد بود:

```
using System;
using System.Collections.Generic;
using PdfRpt.Core.Contracts;
using PdfRpt.FluentInterface;

namespace PdfReportSamples.IList
{
    public class IListPdfReport
    {
        public IPdfReportData CreatePdfReport()
        {
            return new PdfReport().DocumentPreferences(doc =>
            {
                doc.RunDirection(PdfRunDirection.RightToLeft);
                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(footer =>
            {
                footer.DefaultFooter(DateTime.Now.ToString("MM/dd/yyyy"));
            })
            .PagesHeader(header =>
            {
                header.DefaultHeader(defaultHeader =>
                {
                    defaultHeader.ImagePath(AppPath.ApplicationPath + "\\Images\\01.png");
                    defaultHeader.Message("گزارش جدید ما");
                });
            })
            .MainTableTemplate(template =>
            {
                template.BasicTemplate(BasicTemplate.ClassicTemplate);
            })
            .MainTablePreferences(table =>
            {
                table.ColumnsWidthsType(TableColumnWidthType.Relative);
                table.NumberOfDataRowsPerPage(5);
            })
            .MainTableDataSource(dataSource =>
            {
                var listOfRows = new List<User>();
                for (int i = 0; i < 200; i++)
                {
                    listOfRows.Add(new User { Id = i, LastName = "نام خانوادگی" + i, Name = "نام " + i,
Balance = i + 1000 });
                }
                dataSource.StronglyTypedList<User>(listOfRows);
            })
            .MainTableSummarySettings(summarySettings =>
            {
                summarySettings.OverallSummarySettings("جمع کل");
                summarySettings.PerviousPageSummarySettings("نقل از صفحه قبل");
            })
        }
    }
}
```

```

summarySettings.PageSummarySettings("جمع صفحه");
})
.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<User>(x => x.Id);
        column.CellsHorizontalAlignment(HorizontalAlignment.Center);
        column.IsVisible(true);
        column.Order(1);
        column.Width(2);
        column.HeaderCell("شماره");
    });

    columns.AddColumn(column =>
    {
        column.PropertyName<User>(x => x.Name);
        column.CellsHorizontalAlignment(HorizontalAlignment.Center);
        column.IsVisible(true);
        column.Order(2);
        column.Width(3);
        column.HeaderCell("نام");
    });

    columns.AddColumn(column =>
    {
        column.PropertyName<User>(x => x.LastName);
        column.CellsHorizontalAlignment(HorizontalAlignment.Center);
        column.IsVisible(true);
        column.Order(3);
        column.Width(3);
        column.HeaderCell("نام خانوادگی");
    });

    columns.AddColumn(column =>
    {
        column.PropertyName<User>(x => x.Balance);
        column.CellsHorizontalAlignment(HorizontalAlignment.Center);
        column.IsVisible(true);
        column.Order(4);
        column.Width(2);
        column.HeaderCell("موجودی");
        column.ColumnItemsTemplate(template =>
        {
            template.TextBlock();
            template.DisplayFormatFormula(obj => obj == null ? string.Empty :
string.Format("{0:n0}", obj));
        });
        column.AggregateFunction(aggregateFunction =>
        {
            aggregateFunction.NumericAggregateFunction(AggregateFunction.Sum);
            aggregateFunction.DisplayFormatFormula(obj => obj == null ? string.Empty :
string.Format("{0:n0}", obj));
        });
    });

})
.MainTableEvents(events =>
{
    events.DataSourceIsEmpty(message: "رکوردی یافت نشد.");
})
.Export(export =>
{
    export.ToExcel();
    export.ToCsv();
    export.ToXml();
})
.Generate(data => data.AsPdfFile(AppPath.ApplicationPath + "\\Pdf\\RptIListSample.pdf"));
}

```

}

و برای استفاده از آن:

```
var rpt = new IListPdfReport().CreatePdfReport();
// rpt.FileName
```



گزارش جدید ما

ردیف	شماره	نام	نام خانوادگی	موجودی
			نقل از صفحه قبل	۲۱۳,۹۱۵
۱۹۶	۱۹۵	نام ۱۹۵	نام خانوادگی ۱۹۵	۱,۱۹۵
۱۹۷	۱۹۶	نام ۱۹۶	نام خانوادگی ۱۹۶	۱,۱۹۶
۱۹۸	۱۹۷	نام ۱۹۷	نام خانوادگی ۱۹۷	۱,۱۹۷
۱۹۹	۱۹۸	نام ۱۹۸	نام خانوادگی ۱۹۸	۱,۱۹۸
۲۰۰	۱۹۹	نام ۱۹۹	نام خانوادگی ۱۹۹	۱,۱۹۹
			جمع صفحه	۵,۹۸۵
			جمع کل	۲۱۹,۹۰۰

برای نمونه، جهت مشاهده نمایش این خروجی در یک برنامه ویندوزی، به مثال‌های همراه سورس پروژه [در مخزن کد آن](#) مراجعه نمایید.

توضیحات بیشتر:

- در قسمت DocumentPreferences، جهت راست به چپ (PdfRunDirection)، اندازه صفحه (PdfPageSize)، جهت صفحه (PageOrientation) و امثال آن تنظیم می‌شوند.
- سپس نیاز است قلم‌های مورد استفاده در گزارش مشخص شوند. در متد DefaultFonts باید دو قلم را معرفی کنید. قلم اول، قلم پیش فرض خواهد بود و قلم دوم برای رفع نواقص قلم اول مورد استفاده قرار می‌گیرد. برای مثال اگر قلم اول فاقد حروف انگلیسی است، به صورت خودکار به قلم دوم رجوع خواهد شد.
- در ادامه در متد PagesFooter، تاریخ درج شده در پایین تمام صفحات مشخص می‌شود. در مورد ساخت Footer سفارشی در قسمت‌های بعدی بحث خواهد شد.
- در متد PagesHeader، متن و تصویر قرار گرفته در Header تمام صفحات گزارش قابل تنظیم است. این مورد نیز قابل سفارشی سازی است که در قسمت‌های بعد به آن خواهیم پرداخت.
- توسط MainTableTemplate، قالب ظاهری ردیف‌های گزارش مشخص می‌شود. یک سری قالب پیش فرض در کتابخانه PdfReport موجود است که توسط متد BasicTemplate آن قابل دسترسی است. در مورد نحوه تعریف قالب‌های سفارشی به مرور در قسمت‌های بعد، بحث خواهد شد.
- در قسمت MainTablePreferences تنظیمات جدول اصلی گزارش تعیین می‌شود. برای مثال چه تعداد ردیف در صفحه نمایش داده شود. اگر این مورد را تنظیم نکنید، به صورت خودکار محاسبه خواهد شد. نحوه تعیین عرض ستون‌های گزارش به کمک متد ColumnsWidthsType مشخص می‌شود که در اینجا حالت نسبی در نظر گرفته شده است.
- منبع داده مورد استفاده توسط متد MainTableDataSource مشخص می‌شود که در اینجا یک لیست جنریک تعیین شده و سپس

توسط متد `StronglyTypedList` در اختیار گزارش ساز جاری قرار می‌گیرد. تعدادی منبع داده پیش فرض در `PdfReport` وجود دارند که هر کدام را در قسمت‌های بعدی بررسی خواهیم کرد. همچنین امکان تعریف منابع داده سفارشی نیز وجود دارد.

- با کمک متد `MainTableSummarySettings`، برچسب‌های جمع‌های پایین صفحات مشخص می‌شود.
- در قسمت `MainTableColumns`، ستون‌هایی را که علاقمندیم در گزارش ظاهر شوند، قید می‌کنیم. هر ستون باید با یک فیلد یا خاصیت منبع داده متناظر باشد. همچنین همانطور که مشاهده می‌کنید امکان تعیین `Visibility`، عرض و غیره آن نیز مهیا است (قابلیت ساخت گزارشاتی که به انتخاب کاربر، ستون‌های آن ظاهر یا مخفی شوند). در اینجا توسط `callback`‌هایی که در متد `ColumnItemsTemplate` قابل دسترسی هستند، می‌توان اطلاعات را پیش از نمایش فرمت کرد. برای مثال سه رقم جدا کننده به اعداد اضافه کرد (برای نمونه در خاصیت موجودی فوق) و یا توسط متد `AggregateFunction`، می‌توان متد تجمعی مناسبی را جهت ستون جاری مشخص کرد.
- توسط متد `MainTableEvents` به بسیاری از رخدادهای داخلی `PdfReport` دسترسی خواهیم یافت. برای مثال اگر در اینجا رکوردی موجود نباشد، رخداد `DataSourceIsEmpty` صادر خواهد شد.
- به کمک متد `Export`، خروجی‌های دلخواه مورد نظر را می‌توان مشخص کرد. تعدادی خروجی، مانند اکسل، XML و CSV در این کتابخانه موجود است. امکان سفارشی سازی آن‌ها نیز پیش بینی شده است.
- و نهایتاً توسط متد `Generate` مشخص خواهیم کرد که فایل گزارش کجا ذخیره شود.

لطفاً برای طرح مشکلات و سؤالات خود در رابطه با کتابخانه `PdfReport` [از این قسمت سایت](#) استفاده کنید.

نظرات خوانندگان

نویسنده: محمد صافدل
تاریخ: ۸:۴۲ ۱۳۹۱/۰۷/۱۳

ممنون آقای نصیری. من با توجه به مقالات گذشته شما در زمینه iTextSharp یک کتابخانه کوچک برای کارهای خودم تهیه کرده بودم ولی محدودیت‌ها و مشکلات زیادی داشت. با توجه به اینکه کیفیت pdfهای سایت شما به نظرم خیلی خوب هستن تصمیم داشتم از شما درباره نحوه ساخت اونها سوال کنم که خودتون زحمتشو کشیدید. قصد دارم با اجازه شما این کتابخانه را جایگزین کنم. باز هم از شما ممنونم

نویسنده: محمد
تاریخ: ۹:۱۲ ۱۳۹۱/۰۷/۱۳

بسیار بسیار ممنون و متشکر
جدا برای گزارشات خصوصاً در وب مشکل داشتم

نویسنده: امیرحسین جلوداری
تاریخ: ۱۱:۴۸ ۱۳۹۱/۰۷/۱۳

ایول دارین آقای نصیری دی ... از این حرکتتون لذت بردم دی ... خدا اجرت بده دی

نویسنده: mohsen
تاریخ: ۱۳:۸ ۱۳۹۱/۰۷/۱۳

ببخشید آقای نصیری آیا با استفاده از این کتابخانه می‌توان هر گزارشی را با هر شکلی طراحی کرد؟

ممنونم

نویسنده: وحید نصیری
تاریخ: ۱۳:۲۳ ۱۳۹۱/۰۷/۱۳

کلمات «هر گزارشی» و «هر شکلی» یعنی چی؟
اگر «هر شکلی» منظور ساخت گزارشات نامنظم است، شاید [این روش](#) استفاده از قالب‌های open office بهتر باشد.

نویسنده: mohsen
تاریخ: ۱۳:۳۴ ۱۳۹۱/۰۷/۱۳

مثلاً بخواهیم هم از این مثالی که فرمودید با استفاده از open office پیاده سازی کردید استفاده کنم و در پایین صفحه یه جدول که به صورت یک Grid اطلاعات را نمایش دهد. مثلاً بخواهیم اطلاعات یک شخص را در بالای صفحه نمایش دهیم و در پایین صفحه تمام درخواست‌های آن کاربر را باید لیست کنیم در این حالت از کدام روش استفاده کنیم؟

نویسنده: وحید نصیری
تاریخ: ۱۳:۳۸ ۱۳۹۱/۰۷/۱۳

- می‌تونید یک هدر سفارشی در PdfReport درست کنید. (چیزی که به همراه آن است یک هدر ساده همه منظوره است)
- می‌تونید از قالب‌های AcroForm مربوط به Open Office در هدر Pdf Report استفاده کنید.
این موارد رو در قسمت‌های بعدی توضیح می‌دم.

نویسنده: mohsen
تاریخ: ۱۳:۴۰ ۱۳۹۱/۰۷/۱۳

ممنونم استاد. پس من مشتاقانه منتظر این مطلبتون هستم. باز هم ممنونم

نویسنده: مجتبی حاجی وندیان
تاریخ: ۱۴:۰ ۱۳۹۱/۰۷/۱۳

استاد نصیری ممنون از تمام زحماتی که می‌کشید؛ در حال حاضر جز تشکر چیزی از دستم بر نیامد.

نویسنده: مسعود زیانی
تاریخ: ۱۴:۵ ۱۳۹۱/۰۷/۱۳

واقعا از زحماتتون تشکر میکنم

خیلی مفید بود.

نویسنده: سیروان عقیفی
تاریخ: ۱۶:۱۷ ۱۳۹۱/۰۷/۱۳

استاد خیلی ممنون واقعا شاهکاره.

نویسنده: حسن زاده
تاریخ: ۱۲:۳۱ ۱۳۹۱/۰۷/۱۴

ممنون مهندس واقعا کار جالبی هست
به نظرتون طراحی یه محیط Designer برای این کتابخانه هم میتونه خوب باشه
اگه در این مورد موافق بودین من هم می‌تونم کمک کنم

نویسنده: وحید نصیری
تاریخ: ۱۳:۱۸ ۱۳۹۱/۰۷/۱۴

یکی از خوبی‌های کارهای سورس باز امکان مشارکت عموم برنامه نویسی‌ها است. اگر علاقمند به مشارکت در آن بودید، می‌تونید
وصله‌های خودتون رو [در اینجا](#) ارسال نمائید تا به پروژه یا مثال‌های آن اضافه شوند. با تشکر

نویسنده: شاهین کیاست
تاریخ: ۱۸:۴۵ ۱۳۹۱/۰۷/۱۵

فوق العادست .

خسته نباشید.

نویسنده: ناصر طاهری
تاریخ: ۲۲:۴۰ ۱۳۹۱/۰۷/۱۵

مثل همیشه عالی
تشکر فراوان دارم از شما

نویسنده: حسین مرادی نیا
تاریخ: ۲۳:۵۱ ۱۳۹۱/۰۷/۱۵

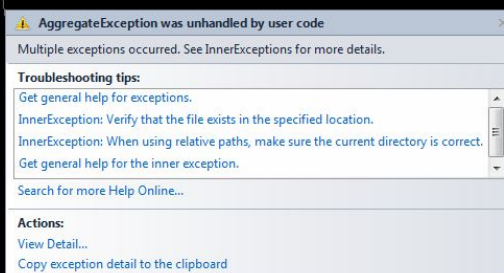
ممنون آقای نصیری
واقعا عالیه

نویسنده: M.B

تاریخ: ۱۰:۴۴ ۱۳۹۱/۰۷/۲۰

استاد من یک پروژه از نوع WebForms ایجاد کردم و سه کلاسی که شما در اینجا ایجاد کردید را عیناً در پروژه ایجاد کردم ولی زمانی که پروژه را اجرا می‌کنم به خطای زیر بر می‌خورم

```
column.ColumnItemsTemplate(template =>
{
    template.TextBlock();
    template.DisplayFormatFormula(obj => obj == null ? string.Empty : string.Format("{0:n0}", obj));
});
column.AggregateFunction(aggregateFunction =>
{
    aggregateFunction.NumericAggregateFunction(AggregateFunction.Sum);
    aggregateFunction.DisplayFormatFormula(obj => obj == null ? string.Empty : string.Format("{0:n0}", obj));
});
});
});
})
.MainTableEvents(events =>
{
    events.DataSourceIsEmpty(message: "رکوردی یافت نشد.");
})
.Export(export =>
{
    export.ToExcel();
    export.ToCsv();
    export.ToXml();
})
.Generate(data => data.AsPdfFile(AppPath.ApplicationPath + "Pdf\\RptIListSample.pdf"));
```



نویسنده: وحید نصیری

تاریخ: ۱۰:۵۰ ۱۳۹۱/۰۷/۲۰

- لطفاً برای رفع مشکلات مرتبط با PdfReport [از این قسمت](#) سایت استفاده نمایید.
 - برای مشاهده خطای واقعی بر روی لینک view details (ذیل قسمت Actions تصویر فوق) کلیک کنید. در اینجا بهتر می‌توان بررسی کرد که مشکل اصلی چه چیزی بوده است. (ممکن است فونت مورد استفاده در مسیر برنامه شما نباشد، یا دسترسی write نداشته باشید، یا پوشه خروجی pdf در مسیر و ریشه برنامه شما ایجاد نشده (مطابق تنظیمات AppPath انتهای گزارش)، یا هر خطای دیگری که ریز آن در قسمت view details یاد شده، ذکر می‌شود)

نویسنده: پویا امینی

تاریخ: ۱۶:۲۱ ۱۳۹۱/۰۸/۲۹

اگر بخواهیم در قسمت Footer گزارش (هر صفحه)، فقط از </hr> و یک نوشته استفاده کنیم باید از چه تنظیماتی استفاده کنیم؟
 ممنون

نویسنده: وحید نصیری

تاریخ: ۱۶:۳۲ ۱۳۹۱/۰۸/۲۹

لطفاً برجسب PdfReport را در سایت بررسی کنید. به این موضوع پرداخته شده: ([^](#))

نویسنده: molana11

تاریخ: ۱۱:۵۴ ۱۳۹۱/۱۰/۱۶

با سلام. آقای نصیری آیا می‌توان از pdf report در سیلورلایت استفاده کرد؟ با تشکر.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۰/۱۶ ۱۲:۳۷

بله (با استفاده از یک سرویس WCF). مثال‌های آن را [دریافت کنید](#) . مراجعه کنید به پوشه [SLPdf](#) آن جهت مشاهده یک نمونه آماده.

نویسنده: molana11
تاریخ: ۱۳۹۱/۱۰/۱۸ ۰:۱۹

آقای نصیری .

با سلام. من قصد دارم گزارشاتم را براساس pdf report ایجاد کنم ولی چند تا مشکل دارم. در صورت امکان راهنمایی کنید:

- 1- آیا می‌شود محتوای فایل پی دی اف را درون سیلورلایت مثلاً در یک گرید یا یک بوردر نمایش داد ؟
 - 2- حجم فایل خروجی در مثالی که شما برای دانلود قرار دادید 400 کیلوبایت بود. آیا می‌توان حجم آنرا پایین آورد؟
 - 3- در مثال شما حدود 20 ثانیه طول کشید تا گزارش ساخته شود. آیا می‌شود این زمان را کمتر کرد؟
- متشکرم از زحمات شما.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۰/۱۸ ۰:۲۹

- نیاز به نمایش دهنده PDF نوشته شده با سیلورلایت دارید. یک سری کار تجاری [از تلریک](#) و امثال آن ([^](#) , [^](#)) برای اینکار هست.

- حجم فایل نهایی به اندازه کافی فشرده شده است. استفاده از تصاویر یا تعداد صفحات بالا، حجم را بیشتر خواهند کرد به همراه بالا بردن مدت زمان تولید فایل. همچنین یک سری پیوست/خروجی جانبی نیز به فایل اضافه می‌شوند، مانند خروجی اکسل، xml و csv. هر کدام از این‌ها را که مورد نیاز نیستند، در حین تهیه گزارش ذکر نکنید تا فایل نهایی حجم کمتری داشته باشد. بدیهی است تولید هر کدام نیز زمانی را به خود اختصاص خواهند داد.

- مثالی که موجود بود را تست کردم حدود 1 ثانیه بیشتر طول نکشید؛ نه 20 ثانیه.

روشی وجود دارد به نام warmup برای خیلی از کارهای دات نت. در پشت صحنه سیستم، حین اجرای اولیه برنامه یک گزارش خالی را تولید کنید. به این صورت سیستم JIT دات نت مجبور خواهد شد سریعتر وارد عمل شود (نه در زمان نیاز). در دفعه بعد فراخوانی گزارشات، نتیجه کار بسیار سریع خواهد بود.

نویسنده: molana11
تاریخ: ۱۳۹۱/۱۰/۱۸ ۰:۴۱

متشکرم از پاسخ شما.

هنگام ساخته شدن گزارش بلافاصله پنجره دانلود باز می‌شود و درون فریم چیزی نمایش داده نمی‌شود، مشکل از کجاست؟ با تشکر.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۱/۱۰/۱۸ ۰:۴۳

احتمالاً دانلود منیجر شما طوری تنظیم شده که لینک‌های PDF را پیش از نمایش در مرورگر، ردیابی می‌کند.

نویسنده: ایلیا اکبری فرد
تاریخ: ۱۳۹۱/۱۰/۲۱ ۱۷:۳۰

مهندس نصیری با سلام.

می‌خواستم بابت تهیه pdfReport از شما تشکر کنم.

من در پروژه خودم از pdfReport استفاده کردم. بسیار هم راضی هستم. هم در سیلورلایت و هم در mvc. مزایای code first در ابتدا برایم غیرقابل قبول بود ولی الان code first انتخاب اول منه. همه جوره این کتابخانه عالیه. بسیار بسیار سپاس مهندس جان. یاعلی.

نویسنده: ایلیا اکبری فرد
تاریخ: ۱۵:۴۱ ۱۳۹۱/۱۰/۲۳

درون سیلورلایت ، سورس فریم را به مسیر فایل پی دی اف (که توسط دستور AsPdfFile تولید می شود بوسیله WCF) ست میکنم. امکان این وجود دارد که فایل بطور مستقیم درون فریم نمایش داده شود و در جایی ذخیره نشود؟

نویسنده: وحید نصیری
تاریخ: ۱۷:۱۱ ۱۳۹۱/۱۰/۲۳

- لطفا از قسمت پرسش و پاسخ مخصوص این پروژه در سایت برای طرح سؤالات خودتون استفاده کنید.
- شبیه به سؤال شما قبلا مطرح شده.

نویسنده: ایلیا اکبری فرد
تاریخ: ۱۷:۲۲ ۱۳۹۱/۱۰/۲۳

متوجه شدم.
متشکرم.

نویسنده: امیر هاشم زاده
تاریخ: ۲۳:۱۳ ۱۳۹۳/۰۵/۲۰

[اینجا](#) راه حلش رو آقای نصیری رو توضیح دادند.

نویسنده: امیر هاشم زاده
تاریخ: ۲۰:۳ ۱۳۹۳/۰۵/۲۲

برچسب های summary، چگونه مقداردهی می شوند؟ و محل آنها چگونه تعیین می شود؟ به عبارت دیگر ارتباط متد MainTableSummarySettings با فیلدهای منبع داده متناظر چگونه است؟

نویسنده: وحید نصیری
تاریخ: ۹:۳۸ ۱۳۹۳/۰۵/۲۲

- به صورت خودکار قبل از اولین ستونی که دارای AggregateFunction است ظاهر خواهند شد.
- همچنین متدهای داخل MainTableSummarySettings دارای پارامترهای دیگری هم هستند (تک پارامتری نیستند) که بتوان توسط آنها محل ظاهر شدن برچسب ها را صریحا مشخص کرد.

نویسنده: امیر هاشم زاده
تاریخ: ۱۰:۱۹ ۱۳۹۳/۰۵/۲۲

1- در صورتیکه چندین ستون عددی داشته باشیم و بخواهیم برای همه آنها برچسب های "نقل از صفحه قبل"، "جمع صفحه"، "جمع کل" داشته باشیم، چگونه امکان پذیر است؟
2- آیا درست متوجه شدم که: جمع ستون دارای AggregateFunction، مقابل برچسب "جمع صفحه" درج می شود؟

نویسنده: وحید نصیری
تاریخ: ۱۰:۳۳ ۱۳۹۳/۰۵/۲۲

- هر ستون عددی که در تعریف آن AggregateFunction درج و مشخص شود، به صورت خودکار دارای فیلدهای ته جمع خواهد بود.
- برچسب ها هم فقط یکبار در هر ردیف مشخص می شوند به صورت خودکار (پیش فرض) یا حالت سفارشی که عنوان شد (مرسوم تمام گزارشات متداول به همین نحو است).