

صورت مساله:

- لیستی از حقوق کارکنان را داریم. در گزارش نهایی آن نیاز است عدد حقوق کارکنانی با مبلغ کمتر از 1000، با رنگی دیگر نمایش داده شوند.

همچنین در این گزارش هر ردیفی که در ماه 7 واقع شده نیز ظاهر عدد سلول مربوط به آن ماه، به رنگ قهوه‌ای و زمینه زرد تغییر یابد.

- در ستون مشخصات افراد این گزارش، نیاز است تصویر کارمند به همراه نام او در ذیل این تصویر (داخل یک سلول) نمایش داده شوند.

چیزی شبیه به این گزارش!

 <p>گزارش جدید ما</p>			
مبلغ	ماه	شخص	ردیف
۲۵,۴۶۰	نقل از ستون قبل		
۱,۵۳۷	۷	 <p>شخص ۲۴</p>	۲۵
۱,۹۵۹	۷	 <p>شخص ۲۵</p>	۲۶
۱,۳۹۰	۱۰	 <p>شخص ۲۶</p>	۲۷
			
مبلغ	ماه	شخص	ردیف
۱۵,۴۶۳	نقل از ستون قبل		
۴۹۱	۱۰	 <p>شخص ۱۶</p>	۱۷
۱,۹۴۳	۷	 <p>شخص ۱۷</p>	۱۸
۱,۷۸۷	۴	 <p>شخص ۱۸</p>	۱۹

مورد اول در گزارشات، اصطلاحا به conditional formatting معروف است و مورد دوم مرتبط است به تهیه قالب‌های سفارشی، بجای استفاده از قالب‌های سلول‌های پیش فرض PdfReport؛ که در ادامه نحوه انجام این موارد را بررسی خواهیم کرد.

ابتدا سورس کامل این مثال را ملاحظه نمائید:

```
using System;
```

```

using iTextSharp.text;
using PdfRpt.Core.Contracts;
using PdfRpt.Core.Helper;
using PdfRpt.FluentInterface;

namespace PdfReportSamples.CustomCellTemplate
{
    public class CustomCellTemplatePdfReport
    {
        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" });
                doc.Compression(new CompressionSettings
                {
                    CompressionLevel = CompressionLevel.BestCompression,
                    EnableCompression = true
                });
            });
            .DefaultFonts(fonts =>
            {
                fonts.Path(AppPath.ApplicationPath + "\\fonts\\irsans.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.SnowyPineTemplate);
            })
            .MainTablePreferences(table =>
            {
                table.ColumnsWidthsType(TableColumnWidthType.Relative);
                table.MultipleColumnsPerPage(new MultipleColumnsPerPage
                {
                    ColumnsGap = 20,
                    ColumnsPerPage = 2,
                    ColumnsWidth = 250,
                    IsRightToLeft = true,
                    TopMargin = 7
                });
            })
            .MainTableDataSource(dataSource =>
            {
                var table = new System.Data.DataTable("لیست حقوق");
                table.Columns.Add("شخص", typeof(string));
                table.Columns.Add("ماه", typeof(int));
                table.Columns.Add("مبلغ", typeof(decimal));

                var rnd = new Random();
                for (int i = 0; i < 200; i++)
                    table.Rows.Add("شخص " + i, rnd.Next(1, 12), rnd.Next(400, 2000));

                dataSource.DataTable(table);
            })
            .MainTableEvents(events =>
            {
                events.DataSourceIsEmpty(message: "There is no data available to display.");
                events.CellCreated(args =>
                {
                    //change the background color of the cell based on the value
                    if (args.RowType == RowType.DataTableRow && args.Cell.RowData.Value != null &&
args.Cell.RowData.Value is decimal)
                    {
                        if ((decimal)args.Cell.RowData.Value <= 1000)
                            args.Cell.BasicProperties.BackgroundColor = BaseColor.CYAN;
                    }
                });
            });
        }
    }
}

```

```

    });
    });
    .MainTableSummarySettings(summary =>
    {
        summary.OverallSummarySettings("جمع کل");
        summary.PageSummarySettings("جمع صفحه");
        summary.PreviousPageSummarySettings("نقل از ستون قبل");
    })
    .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("ردیف");
        });

        columns.AddColumn(column =>
        {
            column.PropertyName("شخص");
            column.CellsHorizontalAlignment(HorizontalAlignment.Center);
            column.IsVisible(true);
            column.Order(1);
            column.Width(3);
            column.HeaderCell("شخص");
            column.ColumnItemsTemplate(t => t.CustomTemplate(new MyCustomCellTemplate()));
        });

        columns.AddColumn(column =>
        {
            column.PropertyName("ماه");
            column.CellsHorizontalAlignment(HorizontalAlignment.Center);
            column.IsVisible(true);
            column.Order(2);
            column.Width(2);
            column.HeaderCell("ماه");
            column.ColumnItemsTemplate(template =>
            {
                template.TextBlock();
                template.ConditionalFormatFormula(list =>
                {
                    var cellValue = int.Parse(list.GetSafeStringValueOf("ماه", nullValue:
"0"));
                    if (cellValue == 7)
                    {
                        return new CellBasicProperties
                        {
                            PdfFontStyle = DocumentFontStyle.Bold |
DocumentFontStyle.Underline,
                            FontColor = new BaseColor(System.Drawing.Color.Brown),
                            BackgroundColor = new BaseColor(System.Drawing.Color.Yellow)
                        };
                    }
                    return new CellBasicProperties { PdfFontStyle = DocumentFontStyle.Normal
};
                });
            });
        });

        columns.AddColumn(column =>
        {
            column.PropertyName("مبلغ");
            column.CellsHorizontalAlignment(HorizontalAlignment.Center);
            column.IsVisible(true);
            column.Order(3);
            column.Width(2);
            column.HeaderCell("مبلغ");
            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));
    });
    });
    .Export(export =>
    {
        export.ToXml();
        export.ToExcel();
    })
    .Generate(data => data.AsPdfFile(AppPath.ApplicationPath +
"\\Pdf\\RptDataTableSample.pdf"));
}
}
}

```

به همراه قالب سلول سفارشی آن:

```

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

namespace PdfReportSamples.CustomCellTemplate
{
    public class MyCustomCellTemplate : IColumnItemsTemplate
    {
        Random _rnd = new Random();

        public void CellRendered(PdfPCell cell, Rectangle position, PdfContentByte[] canvases,
CellAttributes attributes)
        {
        }

        public CellBasicProperties BasicProperties { set; get; }
        public Func<IList<CellData>, CellBasicProperties> ConditionalFormatFormula { set; get; }

        public PdfPCell RenderingCell(CellAttributes attributes)
        {
            var pdfCell = new PdfPCell();
            var table = new PdfPTable(1) { RunDirection = PdfWriter.RUN_DIRECTION_RTL };

            var filePath = AppPath.ApplicationPath + "\\Images\\" + _rnd.Next(1, 5).ToString("00") +
".png";
            var photo = PdfImageHelper.GetITextSharpImageFromImageFile(filePath);
            table.AddCell(new PdfPCell(photo, fit: false)
            {
                Border = 0,
                VerticalAlignment = Element.ALIGN_BOTTOM,
                HorizontalAlignment = Element.ALIGN_CENTER
            });

            var name = attributes.RowData.TableRowData.GetSafeStringValueOf("شخص");
            table.AddCell(new PdfPCell(attributes.BasicProperties.PdfFont.FontSelector.Process(name))
            {
                Border = 0,
                HorizontalAlignment = Element.ALIGN_CENTER
            });

            pdfCell.AddElement(table);

            return pdfCell;
        }
    }
}

```

توضیحات:

- در این مثال از منبع داده‌ای از نوع DataTable استفاده شده است؛ که نحوه بکارگیری آن‌را در متد MainTableDataSource
- ملاحظه می‌کنید. ستون‌های تعریف شده در MainTableColumns نیز بر اساس ستون‌های DataTable مشخص شده‌اند.
- در متد DocumentPreferences، نحوه مشخص سازی فشرده سازی نهایی فایل PDF را ملاحظه می‌کنید. این مورد از مزایای

استفاده از فایل‌های PDF است.

- برای اعمال فرمت شرطی اطلاعات در PdfReport دو روش وجود دارد.
الف) استفاده از متد MainTableEvents و کار کردن با رخدادهای تعریف شده در آن مانند CellCreated. در اینجا می‌توان در نحوه رندر شدن یک سلول دخالت کرد:

```
events.CellCreated(args =>
{
    //change the background color of the cell based on the value
    if (args.RowType == RowType.DataTableRow && args.Cell.RowData.Value != null &&
args.Cell.RowData.Value is decimal)
    {
        if ((decimal)args.Cell.RowData.Value <= 1000)
            args.Cell.BasicProperties.BackgroundColor = BaseColor.CYAN;
    }
});
```

برای مثال در تعاریف فوق، اگر نوع ردیف، از نوع ردیف‌های اطلاعاتی جدول باشد، مقدار آن دریافت شده و بر اساس شرطی مشخص، برای نمونه رنگ پس زمینه آن سلول تغییر داده می‌شود.
ب) همانطور که در قسمت تعریف ستون «ماه» ملاحظه می‌کنید، توسط متد template.ConditionalFormatFormula نیز، امکان فرمت شرطی اطلاعات فراهم شده است. در اینجا می‌توان به لیست اطلاعات سلول‌های ردیف جاری دسترسی یافت و سپس بر اساس آن تصمیم‌گیری کرد.

- جهت تعریف قالب‌های سفارشی سلول‌ها کافی است اینترفیس IColumnItemsTemplate را پیاده‌سازی کنیم؛ که نمونه‌ای از آن را در کدهای MyCustomCellTemplate فوق ملاحظه می‌کنید. در اینجا فرصت خواهید داشت هر شکل و طرح متنوعی را تهیه کرده و به صورت یک PdfPCell بازگشت دهید. برای نمونه در مثال فوق، یک جدول را در سلول تعریف شده قرار داده‌ایم. این جدول یک ستون دارد و هر سلولی که به آن اضافه خواهد شد، یک ردیف را تشکیل خواهد داد. در ردیف اول آن تصویر قرار گرفته و در ردیف دوم آن مقدار سلول جاری.