www.bytescout.com

slice PDF by horizontal lines inside in C# using ByteScout Premium Suite

How to apply ByteScout Premium Suite for slice PDF by horizontal lines inside in C#

Writing of the code to slice PDF by horizontal lines inside in C# can be done by developers of any level using ByteScout Premium Suite. ByteScout Premium Suite helps with slice PDF by horizontal lines inside in C#. ByteScout Premium Suite is the bundle that includes twelve SDK products from ByteScout including tools and components for PDF, barcodes, spreadsheets, screen video recording.

If you want to quickly learn then these fast application programming interfaces of ByteScout Premium Suite for C# plus the guideline and the C# code below will help you quickly learn slice PDF by horizontal lines inside. Just copy and paste this C# sample code to your C# application's code editor, add a reference to ByteScout Premium Suite (if you haven't added yet) and you are ready to go! Use of ByteScout Premium Suite in C# is also described in the documentation included along with the product.

Trial version can be downloaded from our website for free. It contains this and other source code samples for C#.

FOR MORE INFORMATION AND FREE TRIAL:

Download Free Trial SDK (on-premise version)

Read more about ByteScout Premium Suite

Explore API Documentation

Get Free Training for ByteScout Premium Suite

Get Free API key for Web API

visit www.ByteScout.com

Source Code Files:

Program.cs

```
using System;
using System.Drawing;
using System.IO;
using Bytescout.PDF;
using Bytescout.PDFExtractor;
using Bytescout.PDFRenderer;
using Image = System.Drawing.Image;
using Path = System.IO.Path;

namespace SlicePdfByLinesExample
{
    /// <summary>
    /// The example demonstrates slicing of PDF document pages by horizontal lines.
    /// The example uses three products:
    /// - PDF Extractor SDK (https://bytescout.com/products/developer/pdfextractorsdk/f
    ///     to detect lines;
    /// - PDF Renderer SDK (https://bytescout.com/products/developer/pdfrenderersdk/ind
    ///     to render PDF documents to raster images;
    /// - PDF SDK (https://bytescout.com/products/developer/pdfsdk/index.html) -
    ///     to create new PDF documentsd from sliced images.
    /// </summary>
    class Program
    {
        static string InputFile = @".\sample.pdf";
        static string OutputFolder = @".\output";

        static void Main(string[] args)
        {
            if (!Directory.Exists(OutputFolder))
                Directory.CreateDirectory(OutputFolder);

            // Create LineDetector instance and load document
            LineDetector lineDetector = new LineDetector("demo", "demo");
            lineDetector.LoadDocumentFromFile(InputFile);

            // Create RasterRenderer instance and load document
            RasterRenderer rasterRenderer = new RasterRenderer("demo", "demo");
            rasterRenderer.LoadDocumentFromFile(InputFile);

            try
            {
                int pageCount = lineDetector.GetPageCount();

                for (int pageIndex = 0; pageIndex < pageCount; pageIndex++)
                {
                    Console.WriteLine("Processing page #{0}", pageIndex);

                    // Find horizontal lines on the page
                    FoundLinesCollection horzLines = lineDetector.FindLines(pageIndex,
                    // Slice page by separating lines and create new PDF documents
                    Slice(pageIndex, horzLines, rasterRenderer);
                }
            }
            catch (Exception exception)
            {
```

```
                Console.WriteLine(exception);
            }
            finally
            {
                // Cleanup
                rasterRenderer.Dispose();
                lineDetector.Dispose();
            }


            Console.WriteLine();
            Console.WriteLine("Press any key...");
            Console.ReadKey();
        }

        static void Slice(int pageIndex, FoundLinesCollection lines, RasterRenderer ras
        {
            const float pdfRenderingResolution = 300;
            float lastLineY = float.NaN;
            int sliceNum = 0;

            RectangleF pageRect = rasterRenderer.GetPageRectangle(pageIndex);

            foreach (FoundLine line in lines)
            {
                if (float.IsNaN(lastLineY))
                {
                    lastLineY = line.From.Y;
                }
                else
                {
                    // Compute slice rectangle
                    RectangleF sliceRect = new RectangleF(0, lastLineY, pageRect.Width
                    // Set extraction rectangle for RasterRenderer
                    rasterRenderer.SetExtractionArea(sliceRect);
                    // Render the page region to Image object
                    Image slice = rasterRenderer.GetImage(pageIndex, pdfRenderingResolu

                    // Create PDF document
                    Document document = new Document
                    {
                        RegistrationName = "demo",
                        RegistrationKey = "demo"
                    };

                    // Create page of A4 size
                    Page page = new Page(PaperFormat.A4);
                    document.Pages.Add(page);

                    // Create PDF Image object from System.Drawing.Image
                    Bytescout.PDF.Image pdfImage = new Bytescout.PDF.Image(slice);
                    // Draw image on the page keeping the aspect ratio
                    RectangleF r = new RectangleF(0, 20, page.Width, page.Width / slice
                    page.Canvas.DrawImage(pdfImage, r.Left, r.Top, r.Width, r.Height);

                    // Save PDF document
                    string fileName = string.Format(@"{0}\{1}-page{2}-slice{3}.pdf", Ou
                    document.Save(fileName);

                    Console.WriteLine("Saved slice '{0}'", fileName);
```

```
                    // Cleanup
                    document.Dispose();
                    slice.Dispose();

                    lastLineY = line.From.Y;
                }
            }
        }
    }
}
```

SlicePdfByLinesExample.sln

```
Microsoft Visual Studio Solution File, Format Version 12.00
# Visual Studio 15
VisualStudioVersion = 15.0.27428.2037
MinimumVisualStudioVersion = 10.0.40219.1
Project("{FAE04EC0-301F-11D3-BF4B-00C04F79EFBC}") = "SlicePdfByLinesExample", "SlicePd
EndProject
Global
        GlobalSection(SolutionConfigurationPlatforms) = preSolution
                Debug|Any CPU = Debug|Any CPU
                Release|Any CPU = Release|Any CPU
        EndGlobalSection
        GlobalSection(ProjectConfigurationPlatforms) = postSolution
                {11A7C946-41EC-4FD5-8414-EAAA0FB9FC48}.Debug|Any CPU.ActiveCfg = Debug
                {11A7C946-41EC-4FD5-8414-EAAA0FB9FC48}.Debug|Any CPU.Build.0 = Debug|A
                {11A7C946-41EC-4FD5-8414-EAAA0FB9FC48}.Release|Any CPU.ActiveCfg = Rel
                {11A7C946-41EC-4FD5-8414-EAAA0FB9FC48}.Release|Any CPU.Build.0 = Releas
        EndGlobalSection
        GlobalSection(SolutionProperties) = preSolution
                HideSolutionNode = FALSE
        EndGlobalSection
        GlobalSection(ExtensibilityGlobals) = postSolution
                SolutionGuid = {59CACFB0-9FC2-47A5-8E3B-C6497F97B068}
        EndGlobalSection
EndGlobal
```

VIDEO

https://www.youtube.com/watch?v=NEwNs2b9YN8

ON-PREMISE OFFLINE SDK

60 Day Free Trial or Visit ByteScout Premium Suite Home Page
Explore ByteScout Premium Suite Documentation
Explore Samples
Sign Up for ByteScout Premium Suite Online Training

ON-DEMAND REST WEB API

Get Your API Key
Explore Web API Docs
Explore Web API Samples

visit www.ByteScout.com

visit www.PDF.co