How to set invisible text over image in pdf with pdf sdk in VB.NET using ByteScout Premium Suite

This code in VB.NET shows how to set invisible text over image in pdf with pdf sdk with this how to tutorial

On this page you will learn from code samples for programming in VB.NET.Writing of the code to set invisible text over image in pdf with pdf sdk in VB.NET can be executed by programmers of any level using ByteScout Premium Suite. What is ByteScout Premium Suite? It is the set that includes 12 SDK products from ByteScout including tools and components for PDF, barcodes, spreadsheets, screen video recording. It can help you to set invisible text over image in pdf with pdf sdk in your VB.NET application.

Want to quickly learn? This fast application programming interfaces of ByteScout Premium Suite for VB.NET plus the guidelines and the code below will help you quickly learn how to set invisible text over image in pdf with pdf sdk. IF you want to implement the functionality, just copy and paste this code for VB.NET below into your code editor with your app, compile and run your application. Further improvement of the code will make it more robust.

ByteScout provides the free trial version of ByteScout Premium Suite along with the documentation and source code samples.

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.vb

```vbnet
Imports Bytescout.PDF

''' <summary>
''' This example demonstrates how to create PDF document from scanned document image ar
''' </summary>
Class Program

    Shared Sub Main()

        ' Create new PDF document
        Dim pdfDocument = New Document()
        pdfDocument.RegistrationName = "demo"
        pdfDocument.RegistrationKey = "demo"


        ' Load image from file to System.Drawing.Image object (we need it to get the in
        Dim sysImage As System.Drawing.Image = System.Drawing.Image.FromFile(".\scanned
        ' Compute image size in PDF units (Points)
        Dim widthInPoints As Single = sysImage.Width / sysImage.HorizontalResolution *
        Dim heightInPoints As Single = sysImage.Height / sysImage.VerticalResolution *

        ' Create page of computed size
        Dim page = New Page(widthInPoints, heightInPoints)
        ' Add page to the document
        pdfDocument.Pages.Add(page)

        Dim canvas = page.Canvas

        ' Create Bytescout.PDF.Image object from loaded image
        Dim pdfImage = New Image(sysImage)
        ' Draw the image
        canvas.DrawImage(pdfImage, 0, 0, widthInPoints, heightInPoints)

        ' Dispose the System.Drawing.Image object to free resources
        sysImage.Dispose()

        ' Create brush
        Dim transparentBrush As SolidBrush = New SolidBrush(New ColorGray(0))
        ' ... and make it transparent
        transparentBrush.Opacity = 0

        ' Draw text with transparent brush
        Dim font16 As Font = New Font(StandardFonts.Helvetica, 16)
        canvas.DrawString("Your Company Name", font16, transparentBrush, 40, 40)

        ' Draw another text
        Dim font10 As Font = New Font(StandardFonts.Helvetica, 10)
        canvas.DrawString("Your Address", font10, transparentBrush, 40, 80)


        ' Save document to file
        pdfDocument.Save("result.pdf")
```

```vbnet
        ' Cleanup
            pdfDocument.Dispose()

        ' Open document in default PDF viewer app
        Process.Start("result.pdf")

    End Sub

End Class
```

VIDEO

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