www.bytescout.com

How to extract images by page from PDF in VB.NET and ByteScout PDF Extractor SDK

How to code in VB.NET to extract images by page from PDF with this step-by-step tutorial

Extract images by page from PDF is easy to implement in VB.NET if you use these source codes below. Want to extract images by page from PDF in your VB.NET app? ByteScout PDF Extractor SDK is designed for it. ByteScout PDF Extractor SDK is the SDK is designed to help developers with pdf tables and pdf data extraction from unstructured documents like pdf, tiff, scans, images, scanned and electronic forms. The library is powered by OCR, computer vision and AI to provide unique functionality like table detection, automatic table structure extraction, data restoration, data restructuring and reconstruction. Supports PDF, TIFF, PNG, JPG images as input and can output CSV, XML, JSON formatted data. Includes full set of utilities like pdf splitter, pdf merger, searchable pdf maker and other utilities.

You will save a lot of time on writing and testing code as you may just take the VB.NET code from ByteScout PDF Extractor SDK for extract images by page from PDF below and use it in your application. This VB.NET sample code is all you need for your app. Just copy and paste the code, add references (if needs to) and you are all set! Enjoy writing a code with ready-to-use sample codes in VB.NET.

Download free trial version of ByteScout PDF Extractor SDK from our website with this and other source code samples for VB.NET.

FOR MORE INFORMATION AND FREE TRIAL:

Download Free Trial SDK (on-premise version)

Read more about ByteScout PDF Extractor SDK

Explore API Documentation

Get Free Training for ByteScout PDF Extractor SDK

Get Free API key for Web API

visit www.ByteScout.com

Source Code Files:

Program.vb

```
Imports Bytescout.PDFExtractor
Imports System.Drawing.Imaging

Class Program
        Friend Shared Sub Main(args As String())

        ' Create Bytescout.PDFExtractor.ImageExtractor instance
        Dim extractor As New ImageExtractor()
                extractor.RegistrationName = "demo"
                extractor.RegistrationKey = "demo"

                ' Load sample PDF document
        extractor.LoadDocumentFromFile(".\sample1.pdf")

        ' Get count of pages
                Dim pageCount As Integer = extractor.GetPageCount()

        ' Extract images from each page
        For i As Integer = 0 To pageCount - 1

            Dim j As Integer = 0

            ' Initialize page images enumeration
            If extractor.GetFirstPageImage(i) Then
                Do
                    Dim outputFileName As String = "page" & i & "image" & j & ".png"

                    ' Save image to file
                    extractor.SaveCurrentImageToFile(outputFileName, ImageFormat.Png)

                    j = j + 1

                Loop While extractor.GetNextImage() ' Advance image enumeration
            End If

        Next

        ' Cleanup
                extractor.Dispose()

                ' Open result file in default associated application (for demo purpose
        System.Diagnostics.Process.Start("page0image0.png")

        End Sub
End Class
```

VIDEO

https://www.youtube.com/watch?v=s28W3_KMraU

ON-PREMISE OFFLINE SDK

60 Day Free Trial or Visit ByteScout PDF Extractor SDK Home Page
Explore ByteScout PDF Extractor SDK Documentation
Explore Samples
Sign Up for ByteScout PDF Extractor SDK 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