

## ocr analyser in pdf with pdf extractor sdk in VB.NET and ByteScout Premium Suite

Simple tutorial on how to do ocr analyser in pdf with pdf extractor sdk in VB.NET

This page helps you to learn from code samples for programming in VB.NET. ByteScout Premium Suite helps with ocr analyser in pdf with pdf extractor sdk in VB.NET. ByteScout Premium Suite is the bundle that includes twelve SDK products from ByteScout including tools and components for PDF, barcodes, spreadsheets, screen video recording.

VB.NET code snippet like this for ByteScout Premium Suite works best when you need to quickly implement ocr analyser in pdf with pdf extractor sdk in your VB.NET application. If you want to know how it works, then this VB.NET sample code should be copied and pasted into your application's code editor. Then just compile and run it. Use of ByteScout Premium Suite in VB.NET is also described in the documentation included along with the product.

Trial version along with the source code samples for VB.NET can be downloaded from our website

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](http://www.Bytescout.com)

Source Code Files:

Program.vb

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

```
' This example demonstrates the use of OCR Analyser - a tooling class for analysis of s
' in PDF or raster image formats to find best parameters for Optical Character Recognit
' provide highest recognition quality.
```

```
' To make OCR work you should add the following references to your project:
' 'Bytescout.PDFExtractor.dll', 'Bytescout.PDFExtractor.OCRExtension.dll'.
```

Class Program

```
Friend Shared Sub Main(args As String())
```

```
' Input document
```

```
Dim inputDocument As String = ".\sample_ocr.pdf"
```

```
' Document page index
```

```
Dim pageIndex As Integer = 0
```

```
' Area of the document page to perform the analysis (optional).
```

```
' RectangleF.Empty means the full page.
```

```
Dim rectangle As RectangleF = RectangleF.Empty ' New RectangleF(100, 50, 350, 2
```

```
' Location of OCR language data files
```

```
Dim ocrLanguageDataFolder As String = "c:\Program Files\Bytescout PDF Extractor
```

```
' OCR language
```

```
Dim ocrLanguage As String = "eng" ' "eng" for english, "deu" for German, "fra"
```

```
' Find more language files at https://github.com/bytescout/ocrdata/tree/master/
```

```
' Create OCRAnalyzer instance and activate it with your registration informatio
Using ocrAnalyzer As New OCRAnalyzer("demo", "demo")
```

```
' Display analysis progress
```

```
AddHandler ocrAnalyzer.ProgressChanged, Sub(sender, message, progress, ByRe
                                           Console.WriteLine(message)
                                           End Sub
```

```
' Load document to OCRAnalyzer
```

```
ocrAnalyzer.LoadDocumentFromFile(inputDocument)
```

```
' Setup OCRAnalyzer
```

```
ocrAnalyzer.OCRLanguage = ocrLanguage
```

```
ocrAnalyzer.OCRLanguageDataFolder = ocrLanguageDataFolder
```

```
' Set page area for analysis (optional)
```

```
ocrAnalyzer.SetExtractionArea(rectangle)
```

```
' Perform analysis and get results
```

```
Dim analysisResults As OCRAnalysisResults = ocrAnalyzer.AnalyzeByOCRConfide
```

```
' Now extract page text using detected OCR parameters
```

```
Dim outputDocument As String = ".\result.txt"
```

```
' Create TextExtractor instance
Using textExtractor As TextExtractor = New TextExtractor("demo", "demo")

    ' Load document to TextExtractor
    textExtractor.LoadDocumentFromFile(inputDocument)

    ' Setup TextExtractor
    textExtractor.OCRMode = OCRMode.Auto
    textExtractor.OCRLanguageDataFolder = ocrLanguageDataFolder
    textExtractor.OCRLanguage = ocrLanguage

    ' Apply analysis results to TextExtractor instance
    ocrAnalyzer.ApplyResults(analysisResults, textExtractor)

    ' Set extraction area (optional)
    textExtractor.SetExtractionArea(rectangle)

    ' Save extracted text to file
    textExtractor.SaveTextToFile(outputDocument)

    ' Open output file in default associated application (for demonstration)
    System.Diagnostics.Process.Start(outputDocument)

End Using

End Using

End Sub

End Class
```

---

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](http://www.Bytescout.com)

[visit www.PDF.co](http://www.PDF.co)

[www.bytescout.com](http://www.bytescout.com)