

OCR analyser in C# and ByteScout PDF Extractor SDK

Tutorial: how to do OCR analyser in C#

Here you may find thousands pre-made source code pieces for easy implementation in your own programming C# projects. ByteScout PDF Extractor SDK was made to help with OCR analyser in C#. ByteScout PDF Extractor SDK is the Software Development Kit (SDK) that is designed to help developers with 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.

Fast application programming interfaces of ByteScout PDF Extractor SDK for C# plus the instruction and the C# code below will help you quickly learn OCR analyser. This C# sample code should be copied and pasted into your application's code editor. Then just compile and run it to see how it works. C# application implementation typically includes multiple stages of the software development so even if the functionality works please test it with your data and the production environment.

Visit our website provides for free trial version of ByteScout PDF Extractor SDK. Free trial includes lots of source code samples to help you with your C# project.

C# - Program.cs

```
using System;
using System.Drawing;
using System.Diagnostics;
using Bytescout.PDFExtractor;

// This example demonstrates the use of OCR Analyser - a tooling class for analysis
// of scanned documents
// in PDF or raster image formats to find best parameters for Optical Character
// Recognition (OCR) that
// provide highest recognition quality.

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

namespace OCRAnalyser
{
    class Program
    {
        static void Main(string[] args)
        {
            // Input document
            string inputDocument = @".\sample_ocr.pdf";
```

```

// Document page index
int pageIndex = 0;

// Area of the document page to perform the analysis (optional).
// RectangleF.Empty means the full page.
RectangleF rectangle = RectangleF.Empty; // new RectangleF(100, 50, 350,
250);

// Location of language data files
string ocrLanguageDataFolder = @"c:\Program Files\Bytescout PDF Extractor
SDK\ocrdata\";

// OCR language
string ocrLanguage = "eng"; // "eng" for english, "deu" for German, "fra"
for French, "spa" for Spanish etc - according to files in "ocrdata" folder
// Find more language files at https://github.com/bytescout/ocrdata

// Create OCRAnalyzer instance and activate it with your registration
information
using (OCRAnalyzer ocrAnalyzer = new OCRAnalyzer("demo", "demo"))
{
    // Display analysis progress
    ocrAnalyzer.ProgressChanged += (object sender, string message, double
progress, ref bool cancel) =>
    {
        Console.WriteLine(message);
    };

    // 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
    OCRAnalysisResults analysisResults =
ocrAnalyzer.AnalyzeByOCRConfidence(pageIndex);

    // Now extract the text using detected OCR parameters
    string outputDocument = @".\result.txt";

    // Create TextExtractor instance
    using (TextExtractor 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 result document in default associated application (for
demo purpose)
    ProcessStartInfo processStartInfo = new
ProcessStartInfo(outputDocument);
    processStartInfo.UseShellExecute = true;
    Process.Start(processStartInfo);
}
}
}
}
```

FOR MORE INFORMATION AND FREE TRIAL:

[Download Free Trial SDK \(on-premise version\)](#)

[Read more about ByteScout PDF Extractor SDK](#)

[Explore documentation](#)

[Visit www.ByteScout.com](#)

or

[Get Your Free API Key for www.PDF.co Web API](#)