OCR with fast dataset with PDF extractor SDK in C# with ByteScout PDF Suite

Simple tutorial on how to do OCR with fast dataset with PDF extractor SDK in C#

Writing of the code to OCR with fast dataset with PDF extractor SDK in C# can be done by developers of any level using ByteScout PDF Suite. OCR with fast dataset with PDF extractor SDK in C# can be applied with ByteScout PDF Suite. ByteScout PDF Suite is the bundle that provides six different SDK libraries to work with PDF from generating rich PDF reports to extracting data from PDF documents and converting them to HTML. This bundle includes PDF (Generator) SDK, PDF Renderer SDK, PDF Extractor SDK, PDF to HTML SDK, PDF Viewer SDK and PDF Generator SDK for Javascript.

C# code snippet like this for ByteScout PDF Suite works best when you need to quickly implement OCR with fast dataset with PDF extractor SDK in your C# application. Just copy and paste this C# sample code to your C# application's code editor, add a reference to ByteScout PDF Suite (if you haven't added yet) and you are ready to go! Updated and detailed documentation and tutorials are available along with installed ByteScout PDF Suite if you'd like to learn more about the topic and the details of the API.

Our website gives free trial version of ByteScout PDF Suite. It includes all these source code samples with the purpose to assist you with your C# application implementation.

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout PDF Suite](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout PDF Suite](#)

[Get Free API key for Web API](#)

[visit www.ByteScout.com](http://www.ByteScout.com)

Source Code Files:

Program.cs

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

// This example demonstrates the use of Optical Character Recognition (OCR) to extract
// from scanned PDF documents and raster images.

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

namespace OCRExample
{
    class Program
    {
        static void Main(string[] args)
        {
            // Create Bytescout.PDFExtractor.TextExtractor instance
            TextExtractor extractor = new TextExtractor();
            extractor.RegistrationName = "demo";
            extractor.RegistrationKey = "demo";

            // Load sample PDF document
            extractor.LoadDocumentFromFile("sample_ocr.pdf");

            // Enable Optical Character Recognition (OCR)
            // in .Auto mode (SDK automatically checks if needs to use OCR or not)
            extractor.OCRMode = OCRMode.Auto;

            // Set the location of OCR language data files
            extractor.OCRLanguageDataFolder = @"c:\Program Files\Bytescout PDF Extracto

            // Set OCR language
            extractor.OCRLanguage = "eng"; // "eng" for english, "deu" for German, "fr
            // Find more language files at https://github.com/bytescout/ocrdata/tree/m

            // Set PDF document rendering resolution
            extractor.OCRResolution = 300;


            // You can also apply various preprocessing filters
            // to improve the recognition on low-quality scans.

            // Automatically deskew skewed scans
            //extractor.OCRImagePreprocessingFilters.AddDeskew();

            // Remove vertical or horizontal lines (sometimes helps to avoid OCR engine
            //extractor.OCRImagePreprocessingFilters.AddVerticalLinesRemover();
            //extractor.OCRImagePreprocessingFilters.AddHorizontalLinesRemover();

            // Repair broken letters
            //extractor.OCRImagePreprocessingFilters.AddDilate();

            // Remove noise
            //extractor.OCRImagePreprocessingFilters.AddMedian();
```

```csharp
            // Apply Gamma Correction
            //extractor.OCRImagePreprocessingFilters.AddGammaCorrection();

            // Add Contrast
            //extractor.OCRImagePreprocessingFilters.AddContrast(20);


            // (!) You can use new OCRAnalyser class to find an optimal set of image pr
            // filters for your specific document.
            // See "OCR Analyser" example.


            // Save extracted text to file
            extractor.SaveTextToFile("output.txt");

            // Cleanup
            extractor.Dispose();

            // Open result document in default associated application (for demo purpose
            ProcessStartInfo processStartInfo = new ProcessStartInfo("output.txt");
            processStartInfo.UseShellExecute = true;
            Process.Start(processStartInfo);
        }
    }
}
```

VIDEO

ON-PREMISE OFFLINE SDK

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