How to convert scanned PDF to XML with PDF extractor SDK in C# and ByteScout Premium Suite

If you want to learn more then this tutorial will show how to convert scanned PDF to XML with PDF extractor SDK in C#

On this page you will learn from code samples for programming in C#.Writing of the code to convert scanned PDF to XML with PDF extractor SDK in C# can be executed by programmers of any level using ByteScout Premium Suite. ByteScout Premium Suite is the set that includes 12 SDK products from ByteScout including tools and components for PDF, barcodes, spreadsheets, screen video recording and you can use it to convert scanned PDF to XML with PDF extractor SDK with C#.

Want to save time? You will save a lot of time on writing and testing code as you may just take the C# code from ByteScout Premium Suite for convert scanned PDF to XML with PDF extractor SDK below and use it in your application. Follow the instructions from scratch to work and copy the C# code. Check C# sample code samples to see if they respond to your needs and requirements for the project.

You can download free trial version of ByteScout Premium Suite from our website to see and try many others source code samples for C#.

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.cs

```
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 ScannedPdfToXML
{
    class Program
    {
        static void Main(string[] args)
        {
            // Create Bytescout.PDFExtractor.XMLExtractor instance
            XMLExtractor extractor = new XMLExtractor();
            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

            // 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();

            // 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.SaveXMLToFile("output.xml");

            // Cleanup
            extractor.Dispose();

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

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