

OCR (optical character recognition) in PDF in ASP.NET C# using ByteScout PDF Extractor SDK

Write code in ASP.NET C# to make OCR (optical character recognition) in PDF with this How-To tutorial

Every ByteScout tool contains example ASP.NET C# source codes that you can find here or in the folder with installed ByteScout product. OCR (optical character recognition) in PDF in ASP.NET C# can be implemented with ByteScout PDF Extractor SDK. 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 ASP.NET C# plus the instruction and the ASP.NET C# code below will help you quickly learn OCR (optical character recognition) in PDF. ASP.NET C# sample code is all you need: copy and paste the code to your ASP.NET C# application's code editor, add a reference to ByteScout PDF Extractor SDK (if you haven't added yet) and you are ready to go! Enjoy writing a code with ready-to-use sample ASP.NET C# codes to add OCR (optical character recognition) in PDF functions using ByteScout PDF Extractor SDK in ASP.NET C#.

Our website provides free trial version of ByteScout PDF Extractor SDK. It comes along with all these source code samples with the goal to help you with your ASP.NET C# application implementation.

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 \[www.ByteScout.com\]\(http://www.ByteScout.com\)](#)

Source Code Files:

Default.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Default.aspx.cs" Inherits="Opcalculator.Default" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" >
<head runat="server">
    <title>Untitled Page</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>

        </div>
    </form>
</body>
</html>
```

Default.aspx.cs

```
using System;
using Bytescout.PDFExtractor;

// Before running the example, copy missing *.traineddata files from "Redistributable"
// or download from https://github.com/bytescout/ocrdata
// Make sure "Copy to Output Directory" property of each added language file is set to true

namespace OpticalCharacterRecognition
{
    /*
    IF YOU SEE TEMPORARY FOLDER ACCESS ERRORS:

    Temporary folder access is required for web application when you use ByteScout SDK
    If you are getting errors related to the access to temporary folder like "Access to the path ...
    SOLUTION:
    If your IIS Application Pool has "Load User Profile" option enabled the IIS provider ...
    If you are running Web Application under an impersonated account or IIS_IUSRS group ...
    */

    public class Default : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (!IsPostBack)
            {
                string strText = ExtractText("C:/Temp/test.pdf");
                Response.Write(strText);
            }
        }

        private string ExtractText(string strPath)
        {
            string strText = "";
            PDFExtractor extractor = new PDFExtractor();
            extractor.RegistrationName = "Opcalculator";
            extractor.RegistrationKey = "RZD9-4L4X-9K3F-9K3F-9K3F";
            extractor.LoadDocumentFromFile(strPath);
            strText = extractor.GetText();
            extractor.Dispose();
            return strText;
        }
    }
}
```

In this case

- check the User or User Group your web application is running under
- then add permissions for this User or User Group to read and write into that temp
- restart your web application and try again

*/

```
public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        String inputFile = Server.MapPath(@".\bin\sample_ocr.pdf");

        // Location of language files
        String ocrLanguageDataFolder = Server.MapPath(@".\bin\ocrdata");

        // Create Bytescout.PDFExtractor.TextExtractor instance
        using (TextExtractor extractor = new TextExtractor())
        {
            extractor.RegistrationName = "demo";
            extractor.RegistrationKey = "demo";

            // Enable Optical Character Recognition (OCR)
            // in .Auto mode SDK automatically checks if needs to use OCR
            extractor.OCRMode = OCRMode.Auto;
            // Set the location of OCR language data files
            extractor.OCRLanguageDataFolder = ocrLanguageDataFolder;
            // Set OCR language
            extractor.OCRLanguage = "eng"; // "eng" for english, "deu" for
            // 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
            //extractor.OCRImagePreprocessingFilters.AddVerticalLinesRemove();
            //extractor.OCRImagePreprocessingFilters.AddHorizontalLinesRemove();

            // 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
            // filters for your specific document.
            // See "OCR Analyser" example.
        }
    }
}
```

```
// Load PDF document
extractor.LoadDocumentFromFile(inputFile);

// Write extracted text to output stream
Response.Clear();
Response.ContentType = "text/html";

Response.Write("<pre>");
// Write extracted text to output stream
Response.Write(extractor.GetText());
Response.Write("</pre>");

Response.End();
}

}

}
```

Default.aspx.designer.cs

```
//------------------------------------------------------------------------------
// <auto-generated>
//   This code was generated by a tool.
//   Runtime Version:2.0.50727.4952
//
//   Changes to this file may cause incorrect behavior and will be lost if
//   the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------

namespace ExtractAllText {

    /// <summary>
    /// _Default class.
    /// </summary>
    /// <remarks>
    /// Auto-generated class.
    /// </remarks>
    public partial class _Default {

        /// <summary>
        /// form1 control.
        /// </summary>
        /// <remarks>
        /// Auto-generated field.
        /// To modify move field declaration from designer file to code-behind file.
        /// </remarks>
    }
}
```

```
        protected global::System.Web.UI.HtmlControls.HtmlForm form1;
    }
}
```

OpticalCharacterRecognition.sln

```
Microsoft Visual Studio Solution File, Format Version 12.00
# Visual Studio 2013
VisualStudioVersion = 12.0.40629.0
MinimumVisualStudioVersion = 10.0.40219.1
Project("{FAE04EC0-301F-11D3-BF4B-00C04F79EFBC}") = "OpticalCharacterRecognition", "OpticalCharacterRecognition"
EndProject
Global
    GlobalSection(SolutionConfigurationPlatforms) = preSolution
        Debug|Any CPU = Debug|Any CPU
        Release|Any CPU = Release|Any CPU
    EndGlobalSection
    GlobalSection(ProjectConfigurationPlatforms) = postSolution
        {0C256397-34FA-4067-98A7-01D3D2BE0F7E}.Debug|Any CPU.ActiveCfg = Debug|Any CPU
        {0C256397-34FA-4067-98A7-01D3D2BE0F7E}.Debug|Any CPU.Build.0 = Debug|Any CPU
        {0C256397-34FA-4067-98A7-01D3D2BE0F7E}.Release|Any CPU.ActiveCfg = Release|Any CPU
        {0C256397-34FA-4067-98A7-01D3D2BE0F7E}.Release|Any CPU.Build.0 = Release|Any CPU
    EndGlobalSection
    GlobalSection(SolutionProperties) = preSolution
        HideSolutionNode = FALSE
    EndGlobalSection
    GlobalSection(ExtensibilityGlobals) = postSolution
        SolutionGuid = {0C72D936-8833-4995-B5F5-FFCC9B15733B}
    EndGlobalSection
EndGlobal
```

Web.config

```
<?xml version="1.0"?>

<configuration>

    <appSettings/>
    <connectionStrings/>
```

```
<system.web>
  <!--
    Set compilation debug="true" to insert debugging
    symbols into the compiled page. Because this
    affects performance, set this value to true only
    during development.
  -->
  <compilation debug="true" />
  <!--
    The <authentication> section enables configuration
    of the security authentication mode used by
    ASP.NET to identify an incoming user.
  -->
  <authentication mode="Windows" />
  <!--
    The <customErrors> section enables configuration
    of what to do if/when an unhandled error occurs
    during the execution of a request. Specifically,
    it enables developers to configure html error pages
    to be displayed in place of a error stack trace.

    <customErrors mode="RemoteOnly" defaultRedirect="GenericErrorPage.htm">
      <error statusCode="403" redirect="NoAccess.htm" />
      <error statusCode="404" redirect="NotFound.htm" />
    </customErrors>
  -->
</system.web>
</configuration>
```

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](#)

[www.bytescout.com](#)