

## How to extract text from PDF with PDF extractor SDK in ASP.NET C# using ByteScout Premium Suite

How to write a robust code in ASP.NET C# to extract text from PDF with PDF extractor SDK with this step-by-step tutorial

The sample source code below will teach you how to extract text from PDF with PDF extractor SDK in ASP.NET C#. ByteScout Premium Suite is the set that includes 12 SDK products from ByteScout including tools and components for PDF, barcodes, spreadsheets, screen video recording. It can extract text from PDF with PDF extractor SDK in ASP.NET C#.

Want to save time? You will save a lot of time on writing and testing code as you may just take the ASP.NET C# code from ByteScout Premium Suite for extract text from PDF with PDF extractor SDK below and use it in your application. This ASP.NET C# sample code is all you need for your app. Just copy and paste the code, add references (if needs to) and you are all set! Use of ByteScout Premium Suite in ASP.NET C# is also described in the documentation included along with the product.

The trial version of ByteScout Premium Suite can be downloaded for free from our website. It also includes source code samples for ASP.NET C# and other programming languages.

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:

## Default.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Default.aspx.cs" Inherits="ExtractText._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;

namespace ExtractText
{
    public partial class _Default : System.Web.UI.Page
    {
        /*
        IF YOU SEE TEMPORARY FOLDER ACCESS ERRORS:

        Temporary folder access is required for web application when you use Bytescout PDF Extractor component.
        If you are getting errors related to the access to temporary folder like "Temporary folder access is denied" or "Access to temporary folder is denied", it means that your web application does not have sufficient permissions to access temporary folder.

        SOLUTION:

        If your IIS Application Pool has "Load User Profile" option enabled then you will need to grant "Load User Profile" permission to your application pool identity.

        If you are running Web Application under an impersonated account or IIS application pool, then you will need to grant "File System Read" and "File System Write" permission to the user account under which your application is running.

        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 in temporary folder
        - restart your web application and try again

        */

        protected void Page_Load(object sender, EventArgs e)
```

```

{
    String inputFile = Server.MapPath(@".\bin\sample1.pdf");

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

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

    Response.Clear();
    Response.ContentType = "text/html";

    Response.Write("<pre>");

    // Write extracted text to output stream
    extractor.SaveTextToStream(Response.OutputStream);

    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;
}
}
```

## ExtractText.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}") = "ExtractText", "ExtractText.csproj"
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=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](#)

[visit www.PDF.co](#)

[www.bytescout.com](#)