

How to read barcodes from PDF in ASP.NET C# with ByteScout BarCode Reader SDK

The tutorial shows how to read barcodes from PDF in ASP.NET C#

The sample source code below will teach you how to read barcodes from PDF in ASP.NET C#. Want to read barcodes from PDF in your ASP.NET C# app? ByteScout BarCode Reader SDK is designed for it. ByteScout BarCode Reader SDK is the SDK for barcode decoding. Can read all popular types from Code 128, GS1, UPC and Code 39 to QR Code, Datamatrix, PDF417. Images, pdf, TIF images and live web camera are supported as input. Designed to handle documents with noise and defects. Includes optional splitter and merger for pdf and tiff based on barcodes. Batch mode is optimized for high performance with multiple threads. Decoded values can be exported to XML, JSON, CSV or into custom data format.

This code snippet below for ByteScout BarCode Reader SDK works best when you need to quickly read barcodes from PDF in your ASP.NET C# application. In order to implement the functionality, you should copy and paste this code for ASP.NET C# below into your code editor with your app, compile and run your application. Enjoy writing a code with ready-to-use sample ASP.NET C# codes.

ByteScout free trial version is available for download from our website. It includes all these programming tutorials along with source code samples.

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout BarCode Reader SDK](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout BarCode Reader 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="WebBarcodeReaderTester.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>Web Barcode Reader Tester (C#)</title>
</head>
<body>
<form id="form1" runat="server">
<div>
Click browse button to upload a PDF document<br />
<asp:FileUpload ID="FileUpload1" runat="server" /><br />
<br />
<asp:Button id="UploadButton"
Text="Upload file"
OnClick="UploadButton_Click"
runat="server">
</asp:Button>
<br />
<br />
<asp:Label id="UploadStatusLabel" Text="" runat="server"></asp:Label>
<br />
<asp:ListBox ID="ListBox1" runat="server" Visible="False"></asp:ListBox><br />
<br />
</div>
</form>
</body>
</html>
```

Default.aspx.cs

```
using System;
using Bytescout.BarCodeReader;

namespace WebTestSharp
{
    /*
    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 is denied"
    SOLUTION:
```

If your IIS Application Pool has "Load User Profile" option enabled the IIS provide

If you are running Web Application under an impersonated account or IIS_IUSRS group

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 UploadButton_Click(object sender, EventArgs e)
    {
        String savePath = @"\uploads\";

        if (FileUpload1.HasFile)
        {
            String fileName = FileUpload1.FileName;
            savePath += fileName;
            FileUpload1.SaveAs(Server.MapPath(savePath));

            Reader barcodeReader = new Reader();

            // Limit search to 1D barcodes only (exclude 2D barcodes)
            // Change to barcodeReader.BarcodeTypesToFind.SetAll()
            // or select specific type, e.g. barcodeReader.BarcodeTypesToFind.SetAll1D();
            barcodeReader.BarcodeTypesToFind.SetAll1D();

            // reader.MediumTrustLevelCompatible = true; // uncomment
            UploadStatusLabel.Visible = false;

            ListBox1.Items.Clear();
            ListBox1.Visible = true;
            ListBox1.Items.Add("Reading barcode(s) from PDF file \");

            FoundBarcode[] barcodes = barcodeReader.ReadFrom(Server.MapPath(savePath));

            if (barcodes.Length == 0)
            {
                ListBox1.Items.Add("No barcodes found");
            }
            else
            {
                foreach (FoundBarcode barcode in barcodes)
                {
                    ListBox1.Items.Add(String.Format("Found barcode with type '{0}'\n", barcode.Type));
                }
            }
        }
        else
        {
            // Notify the user that a file was not uploaded.
            UploadStatusLabel.Text = "You did not specify a file to upload";
        }
    }
}
```

Default.aspx.designer.cs

```
//-----  
// <auto-generated>  
//   This code was generated by a tool.  
//   Runtime Version:2.0.50727.4927  
//  
//   Changes to this file may cause incorrect behavior and will be lost if  
//   the code is regenerated.  
// </auto-generated>  
//-----  
  
namespace WebTestSharp {  
  
    /// <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;  
  
        /// <summary>  
        /// FileUpload1 control.  
        /// </summary>  
        /// <remarks>  
        /// Auto-generated field.  
        /// To modify move field declaration from designer file to code-behind file.  
        /// </remarks>  
        protected global::System.Web.UI.WebControls.FileUpload FileUpload1;  
  
        /// <summary>  
        /// UploadButton control.  
        /// </summary>  
        /// <remarks>  
        /// Auto-generated field.  
        /// To modify move field declaration from designer file to code-behind file.  
        /// </remarks>  
        protected global::System.Web.UI.WebControls.Button UploadButton;  
    }  
}
```

```

    /// <summary>
    /// UploadStatusLabel control.
    /// </summary>
    /// <remarks>
    /// Auto-generated field.
    /// To modify move field declaration from designer file to code-behind file.
    /// </remarks>
    protected global::System.Web.UI.WebControls.Label UploadStatusLabel;

    /// <summary>
    /// ListBox1 control.
    /// </summary>
    /// <remarks>
    /// Auto-generated field.
    /// To modify move field declaration from designer file to code-behind file.
    /// </remarks>
    protected global::System.Web.UI.WebControls.ListBox ListBox1;
}
}

```

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="FileNotFound.htm" />
</customErrors>
-->
</system.web>
</configuration>
```

VIDEO

<https://www.youtube.com/watch?v=EARS PJFIJMU>

ON-PREMISE OFFLINE SDK

[60 Day Free Trial](#) or [Visit ByteScout BarCode Reader SDK Home Page](#)
[Explore ByteScout BarCode Reader SDK Documentation](#)
[Explore Samples](#)
[Sign Up for ByteScout BarCode Reader SDK Online Training](#)

ON-DEMAND REST WEB API

[Get Your API Key](#)
[Explore Web API Docs](#)
[Explore Web API Samples](#)

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

[visit www.PDF.co](http://www.PDF.co)

www.bytescout.com