

## How to decode page by page from pdf or tiff with barcode reader sdk in ASP.NET VB and ByteScout Data Extraction Suite

If you want to learn more then this tutorial will show how to decode page by page from pdf or tiff with barcode reader sdk in ASP.NET VB

This sample source code below will display you how to decode page by page from pdf or tiff with barcode reader sdk in ASP.NET VB. Want to decode page by page from pdf or tiff with barcode reader sdk in your ASP.NET VB app? ByteScout Data Extraction Suite is designed for it. ByteScout Data Extraction Suite is the bundle that includes three SDK tools for data extraction from PDF, scans, images and from spreadsheets: PDF Extractor SDK, Data Extraction SDK, Barcode Reader SDK.

Want to save time? You will save a lot of time on writing and testing code as you may just take the ASP.NET VB code from ByteScout Data Extraction Suite for decode page by page from pdf or tiff with barcode reader sdk below and use it in your application. IF you want to implement the functionality, just copy and paste this code for ASP.NET VB below into your code editor with your app, compile and run your application. Enjoy writing a code with ready-to-use sample ASP.NET VB codes.

Trial version of ByteScout Data Extraction Suite is available for free. Source code samples are included to help you with your ASP.NET VB app.

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout Data Extraction Suite](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout Data Extraction 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="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/TIFF document<br />
            <asp:FileUpload ID="FileUpload1" runat="server" /><br />
            Page No. to read barcode from:
            <asp:TextBox ID="txtPageNo" Text="1" runat="server"></asp:TextBox>
            <br />
            <asp:Button ID="UploadButton"
                Text="Upload file and read barcodes"
                OnClick="UploadButton_Click"
                runat="server"></asp:Button>
            <br />
            <asp:Label ID="UploadStatusLabel" Text="" runat="server"></asp:Label>
            <br />
            <asp:ListBox ID="ListBox1" runat="server" Visible="False"></asp:ListBox><br />
        </div>
    </form>
</body>
</html>
```

## Default.aspx.designer.vb

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

Option Strict On  
Option Explicit On

Partial Public Class \_Default

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

```
Protected WithEvents form1 As Global.System.Web.UI.HtmlControls.HtmlForm
```

```
'''<summary>  
'''FileUpload1 control.  
'''</summary>  
'''<remarks>  
'''Auto-generated field.  
'''To modify move field declaration from designer file to code-behind file.  
'''</remarks>
```

```
Protected WithEvents FileUpload1 As Global.System.Web.UI.WebControls.FileUpload
```

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

```
Protected WithEvents txtPageNo As Global.System.Web.UI.WebControls.TextBox
```

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

```
Protected WithEvents UploadButton As Global.System.Web.UI.WebControls.Button
```

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

```
Protected WithEvents UploadStatusLabel As Global.System.Web.UI.WebControls.Label
```

```
'''<summary>  
'''ListBox1 control.  
'''</summary>  
'''<remarks>  
'''Auto-generated field.  
'''To modify move field declaration from designer file to code-behind file.  
'''</remarks>
```

```
Protected WithEvents ListBox1 As Global.System.Web.UI.WebControls.ListBox
```

End Class

## Default.aspx.vb

```
Imports Bytescout.BarCodeReader

Public Partial Class _Default
    Inherits Web.UI.Page

    ' 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
    ' SOLUTION:

    ' If your IIS Application Pool has "Load User Profile" option enabled the IIS prov
    ' If you are running Web Application under an impersonated account or IIS_IUSRS gro

    ' 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 te
    ' - restart your web application and try again

    Protected Sub UploadButton_Click(ByVal sender As Object, ByVal e As EventArgs)
        Dim savePath As String = "\uploads\"

        If FileUpload1.HasFile Then
            Dim fileName As [String] = FileUpload1.FileName
            savePath += fileName
            FileUpload1.SaveAs(Server.MapPath(savePath))

            Dim barcodeReader As New Reader()

            ' Limit search to 1D barcodes only (exclude 2D barcodes to speed up the se
            ' Change to barcodeReader.BarcodeTypesToFind.SetAll() to scan
            ' or select specific type, e.g. barcodeReader.BarcodeTypesToFind
            barcodeReader.BarcodeTypesToFind.SetAll1D()

            'reader.MediumTrustLevelCompatible = true ' uncomment this line to enable M

            ' Page No to read
            Dim PageNoToRead As Integer = Convert.ToInt32(txtPageNo.Text)

            UploadStatusLabel.Visible = False
            ListBox1.Items.Clear()
            ListBox1.Visible = True

            ListBox1.Items.Add("Reading barcode(s) from file "" + fileName + ", Page 1")
        End If
    End Sub
End Class
```

```

        ' Reading barcode from document on page-by-page basic
        Dim barcodes As FoundBarcode() = barcodeReader.ReadFrom(Server.MapPath(save

    If barcodes.Length = 0 Then
        ListBox1.Items.Add("No barcodes found")
    Else
        For Each barcode As FoundBarcode In barcodes
            ListBox1.Items.Add(String.Format("Found barcode - Type: '{0}', Value: '{1}'"))
        Next
    End If

Else
    ' Notify the user that a file was not uploaded.
    UploadStatusLabel.Text = "You did not specify a file to upload."
End If
End Sub
End Class

```

## Web.config

```

<?xml version="1.0"?>
<configuration>
    <appSettings/>
    <connectionStrings/>
    <system.web>
        <httpRuntime maxRequestLength="1048576" />

        <!--
            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" targetFramework="4.0"/>
        <!--
            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" />
            </customErrors>
        -->
    </system.web>
</configuration>

```

```
        <error statusCode="404" redirect="FileNotFound.htm" />
    </customErrors>
    -->
        <pages controlRenderingCompatibilityVersion="3.5" clientIDMode="AutoID'
</configuration>
```

---

## VIDEO

<https://www.youtube.com/watch?v=NEwNs2b9YN8>

## ON-PREMISE OFFLINE SDK

[60 Day Free Trial](#) or [Visit ByteScout Data Extraction Suite Home Page](#)  
[Explore ByteScout Data Extraction Suite Documentation](#)  
[Explore Samples](#)  
[Sign Up for ByteScout Data Extraction Suite 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](http://www.bytescout.com)