www.bytescout.com

How to read barcodes from image with barcode reader sdk in ASP.NET C# with ByteScout Premium Suite

Learning is essential in computer world and the tutorial below will demonstrate how to read barcodes from image with barcode reader sdk in ASP.NET C#

We made thousands of pre-made source code pieces for easy implementation in your own programming projects. ByteScout Premium Suite is the bundle that includes twelve SDK products from ByteScout including tools and components for PDF, barcodes, spreadsheets, screen video recording. It can be applied to read barcodes from image with barcode reader sdk using ASP.NET C#.

The SDK samples given below describe how to quickly make your application do read barcodes from image with barcode reader sdk in ASP.NET C# with the help of ByteScout Premium Suite. IF you want to implement the functionality, just copy and paste this code for ASP.NET C# below into your code editor with your app, compile and run your application. This basic programming language sample code for ASP.NET C# will do the whole work for you to read barcodes from image with barcode reader sdk.

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

Default.aspx

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

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

<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 an image<br />
        <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 />
        <asp:Image ID="Image1" runat="server" Height="74px" Width="71px" /></div>
    </form>
</body>
</html>
```

Default.aspx.cs

```
using System;
using System.Drawing;
using System.IO;
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 te
```

```
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 virtualFilePath = Path.Combine(savePath, FileUpload1

            string serverFilePath = Server.MapPath(virtualFilePath);
            if (!Directory.Exists(Path.GetDirectoryName(serverFilePath)))
                Directory.CreateDirectory(Path.GetDirectoryName(serverFilePath));

            FileUpload1.SaveAs(serverFilePath);

                            Image image = null;

                            try
                            {
                using (Stream fileStream = File.OpenRead(serverFilePath))
                                {
                                    image = Image.FromStream(fileStream);
                                }
                            }
                            catch (Exception)
                            {
                            }

                            if (image == null)
                            {
                                    UploadStatusLabel.Visible = true;
                                    UploadStatusLabel.Text = "Your file is not an
                                    Image1.Visible = false;
                                    ListBox1.Visible = false;
                            }
                            else
                            {
                                    UploadStatusLabel.Visible = false;
                Image1.ImageUrl = virtualFilePath;
                                    Image1.Visible = true;
                                Image1.Width = image.Width;
                                Image1.Height = image.Height;
                                    ListBox1.Items.Clear();
                                    ListBox1.Visible = true;

                FindBarcodes(serverFilePath);
```

```
                                        if (ListBox1.Items.Count == 0)
                                                ListBox1.Items.Add("No barcodes found"
                                }
                        }
                        else
                        {
                                // Notify the user that a file was not uploaded.
                                UploadStatusLabel.Text = "You did not specify a file to
                        }
                }

                private void FindBarcodes(string fileName)
                {
                        Reader reader = new Reader();

                        // Limit search to 1D barcodes only (exclude 2D barcodes to spe
                        // Change to reader.BarcodeTypesToFind.SetAll() to scan for all
                        // or select specific type, e.g. reader.BarcodeTypesToFind.PDF4
                        reader.BarcodeTypesToFind.SetAll1D();

                        // reader.MediumTrustLevelCompatible = true; // uncomment this

                        reader.ReadFromFile(fileName);

                        foreach (FoundBarcode barcode in reader.FoundBarcodes)
                        {
                                ListBox1.Items.Add(String.Format("{0} : {1}", barcode.T
                        }
                }
        }
}
```

Default.aspx.designer.cs

```
//------------------------------------------------------------------------------
// <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>
//------------------------------------------------------------------------------

namespace WebTestSharp {


    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;

        /// <summary>
        /// Image1 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.Image Image1;
    }
}
```

Global.asax.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.SessionState;

namespace Barcodes_From_Image
{
    public class Global : System.Web.HttpApplication
    {

        void Application_Start(object sender, EventArgs e)
        {
            // Code that runs on application startup

        }

        void Application_End(object sender, EventArgs e)
        {
            //  Code that runs on application shutdown

        }

        void Application_Error(object sender, EventArgs e)
        {
            // Code that runs when an unhandled error occurs

        }

        void Session_Start(object sender, EventArgs e)
        {
            // Code that runs when a new session is started

        }

        void Session_End(object sender, EventArgs e)
        {
            // Code that runs when a session ends.
            // Note: The Session_End event is raised only when the sessionstate mode
            // is set to InProc in the Web.config file. If session mode is set to State
            // or SQLServer, the event is not raised.

        }

    }
}
```

Site.Master.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace Barcodes_From_Image
{
    public partial class SiteMaster : System.Web.UI.MasterPage
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }
    }
}
```

Site.Master.designer.cs

```
//------------------------------------------------------------------------------
// <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>
//------------------------------------------------------------------------------

namespace Barcodes_From_Image
{


    public partial class SiteMaster
    {

        /// <summary>
        /// HeadContent 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.ContentPlaceHolder HeadContent;
```

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

        /// <summary>
        /// HeadLoginView 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.LoginView HeadLoginView;

        /// <summary>
        /// HeadHomeLink 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.HtmlAnchor HeadHomeLink;

        /// <summary>
        /// HeadAboutLink 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.HtmlAnchor HeadAboutLink;

        /// <summary>
        /// MainContent 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.ContentPlaceHolder MainContent;
    }
}
```

Web.Debug.config

```
```

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

<!-- For more information on using web.config transformation visit http://go.microsoft

<configuration xmlns:xdt="http://schemas.microsoft.com/XML-Document-Transform">
  <!--
    In the example below, the "SetAttributes" transform will change the value of
    "connectionString" to use "ReleaseSQLServer" only when the "Match" locator
    finds an atrribute "name" that has a value of "MyDB".

    <connectionStrings>
      <add name="MyDB"
        connectionString="Data Source=ReleaseSQLServer;Initial Catalog=MyReleaseDB;Inte
        xdt:Transform="SetAttributes" xdt:Locator="Match(name)"/>
    </connectionStrings>
  -->
  <system.web>
    <!--
      In the example below, the "Replace" transform will replace the entire
      <customErrors> section of your web.config file.
      Note that because there is only one customErrors section under the
      <system.web> node, there is no need to use the "xdt:Locator" attribute.

      <customErrors defaultRedirect="GenericError.htm"
        mode="RemoteOnly" xdt:Transform="Replace">
        <error statusCode="500" redirect="InternalError.htm"/>
      </customErrors>
    -->
  </system.web>
</configuration>
```

Web.Release.config

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

<!-- For more information on using web.config transformation visit http://go.microsoft

<configuration xmlns:xdt="http://schemas.microsoft.com/XML-Document-Transform">
  <!--
    In the example below, the "SetAttributes" transform will change the value of
    "connectionString" to use "ReleaseSQLServer" only when the "Match" locator
    finds an atrribute "name" that has a value of "MyDB".

    <connectionStrings>
      <add name="MyDB"
        connectionString="Data Source=ReleaseSQLServer;Initial Catalog=MyReleaseDB;Inte
        xdt:Transform="SetAttributes" xdt:Locator="Match(name)"/>
    </connectionStrings>
  -->
  <system.web>
    <compilation xdt:Transform="RemoveAttributes(debug)" />
```

```
    <!--
      In the example below, the "Replace" transform will replace the entire
      <customErrors> section of your web.config file.
      Note that because there is only one customErrors section under the
      <system.web> node, there is no need to use the "xdt:Locator" attribute.

      <customErrors defaultRedirect="GenericError.htm"
        mode="RemoteOnly" xdt:Transform="Replace">
        <error statusCode="500" redirect="InternalError.htm"/>
      </customErrors>
    -->
  </system.web>
</configuration>
```

Web.config

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

<!--
  For more information on how to configure your ASP.NET application, please visit
  http://go.microsoft.com/fwlink/?LinkId=169433
  -->

<configuration>
  <connectionStrings>
    <add name="ApplicationServices"
         connectionString="data source=.\SQLEXPRESS;Integrated Security=SSPI;AttachDBF
         providerName="System.Data.SqlClient" />
  </connectionStrings>

  <system.web>
    <compilation debug="true" targetFramework="4.0" />

    <authentication mode="Forms">
      <forms loginUrl="~/Account/Login.aspx" timeout="2880" />
    </authentication>

    <membership>
      <providers>
        <clear/>
        <add name="AspNetSqlMembershipProvider" type="System.Web.Security.SqlMembershi
             enablePasswordRetrieval="false" enablePasswordReset="true" requiresQuesti
             maxInvalidPasswordAttempts="5" minRequiredPasswordLength="6" minRequiredN
             applicationName="/" />
      </providers>
    </membership>

    <profile>
      <providers>
        <clear/>
        <add name="AspNetSqlProfileProvider" type="System.Web.Profile.SqlProfileProvid
```

```
      </providers>
    </profile>

    <roleManager enabled="false">
      <providers>
        <clear/>
        <add name="AspNetSqlRoleProvider" type="System.Web.Security.SqlRoleProvider" co
        <add name="AspNetWindowsTokenRoleProvider" type="System.Web.Security.WindowsTo
      </providers>
    </roleManager>

  </system.web>

  <system.webServer>
     <modules runAllManagedModulesForAllRequests="true"/>
  </system.webServer>
</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