

How to generate barcode in ASP in ASP.NET and ByteScout Barcode SDK

How to generate barcode in ASP in ASP.NET

This sample source code below will demonstrate you how to generate barcode in ASP in ASP.NET. ByteScout Barcode SDK: the robust library (Software Development Kit) that is designed for automatic generation of high-quality barcodes for printing, electronic documents and pdf. All popular barcode types are supported from Code 39 and Code 129 to QR Code, UPC, GS1, GS-128, Datamatrix, PDF417, Maxicode and many others. Provides support for full customization of fonts, colors, output and printing sizes. Special tools are included to verify output quality and printing quality. Can add generated barcode into new or existing documents, images and PDF. It can generate barcode in ASP in ASP.NET.

Fast application programming interfaces of ByteScout Barcode SDK for ASP.NET plus the instruction and the code below will help you quickly learn how to generate barcode in ASP. In your ASP.NET project or application you may simply copy & paste the code and then run your app! This basic programming language sample code for ASP.NET will do the whole work for you to generate barcode in ASP.

Free trial version of ByteScout Barcode SDK is available for download from our website. Get it to try other source code samples for ASP.NET.

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout Barcode SDK](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout Barcode SDK](#)

[Get Free API key for Web API](#)

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

Source Code Files:

```
using System;
using System.Web;
using System.Web.Mvc;
using System.Linq;
using Bytescout.BarCode;

namespace BarcodeGenerator.Helpers
{
    public static class GeneratorHelper
    {
        public static string GetBase64Barcode(string value, SymbologyType symbology, bool addChecksum)
        {
            using (var barcode = new Barcode())
            {
                barcode.Symbology = symbology;
                barcode.Value = value;

                barcode.AddChecksum = addChecksum;

                var imageBytes = barcode.GetImageBytesPNG();

                return Convert.ToBase64String(imageBytes);
            }
        }

        public static IHtmlString GetBarcode(this HtmlHelper helper,
            string value,
            SymbologyType symbology,
            bool addChecksum = false)
        {
            var base64ImageBytes = GetBase64Barcode(value, symbology, addChecksum);
            var str = string.Format("<img src=\"data:image/png;base64,{0}\">", base64ImageBytes);
            return MvcHtmlString.Create(str);
        }

        public static IHtmlString EmbeddedBarcodeImage(this HtmlHelper helper,
            string value,
            SymbologyType symbology,
            bool addChecksum = false)
        {
            using (var barcode = new Barcode())
            {
                barcode.Symbology = symbology;
                barcode.Value = value;

                barcode.AddChecksum = addChecksum;

                var imageBytes = barcode.GetImageBytesPNG();
                var str = string.Format("data:image/png;base64,{0}",
                    Convert.ToBase64String(imageBytes));
                return MvcHtmlString.Create(str);
            }
        }
    }
}
```

```
}  
}  
}
```

Global.asax.cs

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Web;  
using System.Web.Http;  
using System.Web.Mvc;  
using System.Web.Optimization;  
using System.Web.Routing;  
  
namespace MvcApplication1  
{  
    // Note: For instructions on enabling IIS6 or IIS7 classic mode,  
    // visit http://go.microsoft.com/?LinkId=9394801  
  
    public class MvcApplication : System.Web.HttpApplication  
    {  
        protected void Application_Start()  
        {  
            AreaRegistration.RegisterAllAreas();  
  
            WebApiConfig.Register(GlobalConfiguration.Configuration);  
            FilterConfig.RegisterGlobalFilters(GlobalFilters.Filters);  
            RouteConfig.RegisterRoutes(RouteTable.Routes);  
            BundleConfig.RegisterBundles(BundleTable.Bundles);  
        }  
    }  
}
```

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 attribute "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.com/fwlink/?LinkId=72351 -->
<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 attribute "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" encoding="utf-8"?>
<!--
  For more information on how to configure your ASP.NET application, please visit
  http://go.microsoft.com/fwlink/?LinkId=152368
-->
<configuration>
  <configSections>
    <!-- For more information on Entity Framework configuration, visit http://go.micro
    <section name="entityFramework" type="System.Data.Entity.Internal.ConfigFile.Entity
  </configSections>
  <connectionStrings>
    <add name="DefaultConnection" providerName="System.Data.SqlClient" connectionString
  </connectionStrings>
  <appSettings>
    <add key="webpages:Version" value="2.0.0.0" />
    <add key="webpages:Enabled" value="false" />
    <add key="PreserveLoginUrl" value="true" />
    <add key="ClientValidationEnabled" value="true" />
    <add key="UnobtrusiveJavaScriptEnabled" value="true" />
  </appSettings>
  <system.web>
    <compilation debug="true" targetFramework="4.0" />
    <authentication mode="Forms">
      <forms loginUrl="~/Account/Login" timeout="2880" />
    </authentication>
    <pages>
      <namespaces>
        <add namespace="System.Web.Helpers" />
        <add namespace="System.Web.Mvc" />
        <add namespace="System.Web.Mvc.Ajax" />
        <add namespace="System.Web.Mvc.Html" />
        <add namespace="System.Web.Optimization" />
        <add namespace="System.Web.Routing" />
        <add namespace="System.Web.WebPages" />
      </namespaces>
    </pages>
    <profile defaultProvider="DefaultProfileProvider">
      <providers>
```

```

    <add name="DefaultProfileProvider" type="System.Web.Providers.DefaultProfilePro
  </providers>
</profile>
<membership defaultProvider="DefaultMembershipProvider">
  <providers>
    <add name="DefaultMembershipProvider" type="System.Web.Providers.DefaultMembers
  </providers>
</membership>
<roleManager defaultProvider="DefaultRoleProvider">
  <providers>
    <add name="DefaultRoleProvider" type="System.Web.Providers.DefaultRoleProvider
  </providers>
</roleManager>
<!--
    If you are deploying to a cloud environment that has multiple web server in
    you should change session state mode from "InProc" to "Custom". In addition
    change the connection string named "DefaultConnection" to connect to an ins
    of SQL Server (including SQL Azure and SQL Compact) instead of to SQL Serv
  -->
<sessionState mode="InProc" customProvider="DefaultSessionProvider">
  <providers>
    <add name="DefaultSessionProvider" type="System.Web.Providers.DefaultSessionSt
  </providers>
</sessionState>
</system.web>
<system.webServer>
  <validation validateIntegratedModeConfiguration="false" />
  <modules runAllManagedModulesForAllRequests="true" />
  <handlers>
    <remove name="ExtensionlessUrlHandler-ISAPI-4.0_32bit" />
    <remove name="ExtensionlessUrlHandler-ISAPI-4.0_64bit" />
    <remove name="ExtensionlessUrlHandler-Integrated-4.0" />
    <add name="ExtensionlessUrlHandler-ISAPI-4.0_32bit" path="*." verb="GET,HEAD,POST
    <add name="ExtensionlessUrlHandler-ISAPI-4.0_64bit" path="*." verb="GET,HEAD,POST
    <add name="ExtensionlessUrlHandler-Integrated-4.0" path="*." verb="GET,HEAD,POST
  </handlers>
</system.webServer>
<runtime>
  <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
    <dependentAssembly>
      <assemblyIdentity name="System.Web.Helpers" publicKeyToken="31bf3856ad364e35" /
      <bindingRedirect oldVersion="1.0.0.0-2.0.0.0" newVersion="2.0.0.0" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="System.Web.Mvc" publicKeyToken="31bf3856ad364e35" />
      <bindingRedirect oldVersion="1.0.0.0-4.0.0.0" newVersion="4.0.0.0" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="System.Web.WebPages" publicKeyToken="31bf3856ad364e35"
      <bindingRedirect oldVersion="1.0.0.0-2.0.0.0" newVersion="2.0.0.0" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="WebGrease" publicKeyToken="31bf3856ad364e35" />
      <bindingRedirect oldVersion="0.0.0.0-1.3.0.0" newVersion="1.3.0.0" />
    </dependentAssembly>
  </assemblyBinding>
</runtime>
<entityFramework>
  <defaultConnectionFactory type="System.Data.Entity.Infrastructure.SqlConnectio
</entityFramework>

```

```
</configuration>
```

packages.config

```
<?xml version="1.0" encoding="utf-8"?>
<packages>
  <package id="EntityFramework" version="5.0.0" targetFramework="net40" />
  <package id="jQuery" version="1.8.2" targetFramework="net40" />
  <package id="jQuery.UI.Combined" version="1.8.24" targetFramework="net40" />
  <package id="jQuery.Validation" version="1.10.0" targetFramework="net40" />
  <package id="knockoutjs" version="2.2.0" targetFramework="net40" />
  <package id="Microsoft.AspNet.Mvc" version="4.0.30506.0" targetFramework="net40" />
  <package id="Microsoft.AspNet.Mvc.FixedDisplayModes" version="1.0.0" targetFramework="net40" />
  <package id="Microsoft.AspNet.Providers.Core" version="1.2" targetFramework="net40" />
  <package id="Microsoft.AspNet.Providers.LocalDB" version="1.1" targetFramework="net40" />
  <package id="Microsoft.AspNet.Razor" version="2.0.30506.0" targetFramework="net40" />
  <package id="Microsoft.AspNet.Web.Optimization" version="1.0.0" targetFramework="net40" />
  <package id="Microsoft.AspNet.WebApi" version="4.0.30506.0" targetFramework="net40" />
  <package id="Microsoft.AspNet.WebApi.Client" version="4.0.30506.0" targetFramework="net40" />
  <package id="Microsoft.AspNet.WebApi.Core" version="4.0.30506.0" targetFramework="net40" />
  <package id="Microsoft.AspNet.WebApi.WebHost" version="4.0.30506.0" targetFramework="net40" />
  <package id="Microsoft.AspNet.WebPages" version="2.0.30506.0" targetFramework="net40" />
  <package id="Microsoft.jQuery.Unobtrusive.Ajax" version="2.0.30506.0" targetFramework="net40" />
  <package id="Microsoft.jQuery.Unobtrusive.Validation" version="2.0.30506.0" targetFramework="net40" />
  <package id="Microsoft.Net.Http" version="2.0.20710.0" targetFramework="net40" />
  <package id="Microsoft.Web.Infrastructure" version="1.0.0.0" targetFramework="net40" />
  <package id="Modernizr" version="2.6.2" targetFramework="net40" />
  <package id="Newtonsoft.Json" version="4.5.11" targetFramework="net40" />
  <package id="WebGrease" version="1.3.0" targetFramework="net40" />
</packages>
```

VIDEO

<https://www.youtube.com/watch?v=REnj3A-oSPI>

ON-PREMISE OFFLINE SDK

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