

How to decode damaged QR code in C# using ByteScout BarCode Reader SDK

Tutorial on how to decode damaged QR code in C#

These source code samples are listed and grouped by their programming language and functions they use. Want to decode damaged QR code in your 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 decode damaged QR code in your C# application. In your C# project or application you may simply copy & paste the code and then run your app! Enjoy writing a code with ready-to-use sample C# codes.

ByteScout BarCode Reader SDK free trial version is available on our website. C# and other programming languages are supported.

DecodeDamagedQRCode.NETCore.csproj

```
<?xml version="1.0" encoding="utf-8"?>
<Project Sdk="Microsoft.NET.Sdk">
  <PropertyGroup>
    <OutputType>Exe</OutputType>
    <TargetFramework>netcoreapp2.0</TargetFramework>
    <EnableDefaultCompileItems>>false</EnableDefaultCompileItems>
    <GenerateAssemblyCompanyAttribute>>false</GenerateAssemblyCompanyAttribute>
    <GenerateAssemblyConfigurationAttribute>>false</GenerateAssemblyConfigurationAttribute>
    <GenerateAssemblyFileVersionAttribute>>false</GenerateAssemblyFileVersionAttribute>
    <GenerateAssemblyInformationalVersionAttribute>>false</GenerateAssemblyInformationalVersionAttribute>
    <GenerateAssemblyProductAttribute>>false</GenerateAssemblyProductAttribute>
    <GenerateAssemblyTitleAttribute>>false</GenerateAssemblyTitleAttribute>
    <GenerateAssemblyVersionAttribute>>false</GenerateAssemblyVersionAttribute>
    <GenerateAssemblyCopyrightAttribute>>false</GenerateAssemblyCopyrightAttribute>
    <GenerateAssemblyTrademarkAttribute>>false</GenerateAssemblyTrademarkAttribute>
    <GenerateAssemblyCultureAttribute>>false</GenerateAssemblyCultureAttribute>
    <GenerateAssemblyDescriptionAttribute>>false</GenerateAssemblyDescriptionAttribute>
  </PropertyGroup>
  <ItemGroup>
    <Compile Include="Program.cs" />
    <None Include="corrupted_barcode_qrcode.png">
```

```

    <CopyToOutputDirectory>Always</CopyToOutputDirectory>
  </None>
</ItemGroup>
<ItemGroup>
  <PackageReference Include="Microsoft.Windows.Compatibility" Version="2.0.0" />
</ItemGroup>
<ItemGroup>
  <Reference Include="Bytescout.BarCodeReader">
    <SpecificVersion>False</SpecificVersion>
    <HintPath>c:\Program Files\Bytescout BarCode Reader SDK\netcoreapp2.0\Bytescout.BarCodeReader.dll</HintPath>
  </Reference>
</ItemGroup>
</Project>

```

DecodeDamagedQRCode.csproj

```

<?xml version="1.0" encoding="utf-8"?>
<Project ToolsVersion="15.0" xmlns="http://schemas.microsoft.com/developer/msbuild/2003" >
  <Import Project="$(MSBuildExtensionsPath)\$(MSBuildToolsVersion)\Microsoft.Common.props" />
  <PropertyGroup>
    <Configuration Condition="'$(Configuration)' == ''">Debug</Configuration>
    <Platform Condition="'$(Platform)' == ''">AnyCPU</Platform>
    <ProjectGuid>{99735776-2956-463D-9795-EBCE16928C30}</ProjectGuid>
    <OutputType>Exe</OutputType>
    <RootNamespace>DecodeDamagedQRCode</RootNamespace>
    <AssemblyName>DecodeDamagedQRCode</AssemblyName>
    <TargetFrameworkVersion>v2.0</TargetFrameworkVersion>
    <FileAlignment>512</FileAlignment>
  </PropertyGroup>
  <PropertyGroup Condition="'$(Configuration)|$(Platform)' == 'Debug|AnyCPU' ">
    <PlatformTarget>AnyCPU</PlatformTarget>
    <DebugSymbols>>true</DebugSymbols>
    <DebugType>full</DebugType>
    <Optimize>>false</Optimize>
    <OutputPath>bin\Debug\</OutputPath>
    <DefineConstants>DEBUG;TRACE</DefineConstants>
    <ErrorReport>prompt</ErrorReport>
    <WarningLevel>4</WarningLevel>
  </PropertyGroup>
  <PropertyGroup Condition="'$(Configuration)|$(Platform)' == 'Release|AnyCPU' ">
    <PlatformTarget>AnyCPU</PlatformTarget>
    <DebugType>pdbonly</DebugType>
    <Optimize>>true</Optimize>
    <OutputPath>bin\Release\</OutputPath>
    <DefineConstants>TRACE</DefineConstants>
    <ErrorReport>prompt</ErrorReport>
    <WarningLevel>4</WarningLevel>
  </PropertyGroup>
  <ItemGroup>
    <Reference Include="Bytescout.BarCodeReader, Version=10.2.0.1799, Culture=neutral,
      <SpecificVersion>False</SpecificVersion>

```

```

    <HintPath>c:\Program Files\Bytescout BarCode Reader SDK\net2.00\Bytescout.BarCode
</Reference>
<Reference Include="System" />
<Reference Include="System.Drawing" />
</ItemGroup>
<ItemGroup>
    <Compile Include="Program.cs" />
</ItemGroup>
<ItemGroup>
    <Content Include="corrupted_barcode_qrcode.png">
        <CopyToOutputDirectory>Always</CopyToOutputDirectory>
    </Content>
</ItemGroup>
<Import Project="$(MSBuildToolsPath)\Microsoft.CSharp.targets" />
</Project>

```

DecodeDamagedQRCode.sln

```

Microsoft Visual Studio Solution File, Format Version 12.00
# Visual Studio 15
VisualStudioVersion = 15.0.27703.2026
MinimumVisualStudioVersion = 10.0.40219.1
Project("{FAE04EC0-301F-11D3-BF4B-00C04F79EFBC}") = "DecodeDamagedQRCode", "DecodeDamagedQRCode.sln"
EndProject
Global
    GlobalSection(SolutionConfigurationPlatforms) = preSolution
        Debug|Any CPU = Debug|Any CPU
        Release|Any CPU = Release|Any CPU
    EndGlobalSection
    GlobalSection(ProjectConfigurationPlatforms) = postSolution
        {99735776-2956-463D-9795-EBCE16928C30}.Debug|Any CPU.ActiveCfg = Debug|Any CPU
        {99735776-2956-463D-9795-EBCE16928C30}.Debug|Any CPU.Build.0 = Debug|Any CPU
        {99735776-2956-463D-9795-EBCE16928C30}.Release|Any CPU.ActiveCfg = Release|Any CPU
        {99735776-2956-463D-9795-EBCE16928C30}.Release|Any CPU.Build.0 = Release|Any CPU
    EndGlobalSection
    GlobalSection(SolutionProperties) = preSolution
        HideSolutionNode = FALSE
    EndGlobalSection
    GlobalSection(ExtensibilityGlobals) = postSolution
        SolutionGuid = {3ABE3EEF-B212-4E8B-9A74-67A52FD333AC}
    EndGlobalSection
EndGlobal

```

```
using System;
using Bytescout.BarCodeReader;

namespace DecodeDamagedQRCode
{
    class Program
    {
        static void Main(string[] args)
        {
            try
            {
                // Create and activate Bytescout.BarCodeReader.Reader instance
                using (Reader reader = new Reader("demo", "demo"))
                {
                    // Set barcode type to find
                    reader.BarcodeTypesToFind.QRCode = true;

                    // Read barcodes
                    FoundBarcode[] barcodes = reader.ReadFrom("corrupted_barcode_qrcode");

                    foreach (FoundBarcode code in barcodes)
                        Console.WriteLine("Found barcode with type '{0}' and value '{1}'");
                }
            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.Message);
            }

            Console.WriteLine();
            Console.WriteLine("Press any key to exit...");
            Console.ReadLine();
        }
    }
}
```

FOR MORE INFORMATION AND FREE TRIAL:

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

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

[Explore documentation](#)

[Visit www.ByteScout.com](#)

or

[Get Your Free API Key for www.PDF.co Web API](#)

www.bytescout.com