

## How to add custom images to PDF document in VB.NET with ByteScout Barcode SDK

This tutorial will show how to add custom images to PDF document in VB.NET

Add custom images to PDF document is easy to implement in VB.NET if you use these source codes below. 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 add custom images to PDF document in VB.NET.

You will save a lot of time on writing and testing code as you may just take the VB.NET code from ByteScout Barcode SDK for add custom images to PDF document below and use it in your application. Just copy and paste the code into your VB.NET application's code and follow the instruction. Use of ByteScout Barcode SDK in VB.NET is also explained in the documentation included along with the product.

ByteScout Barcode SDK free trial version is available on our website. VB.NET and other programming languages are supported.

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:

## Program.vb

```
Imports System.Diagnostics
Imports System.Drawing
Imports Bytescout.BarCode

' This example demonstrates adding of barcode together with some custom images (e.g. target mark)
' Also shows the manual calculation of barcode size from inches to pixels and document resolution
' Task: place Code39 barcode of 2" x 3/8" size at 2" from top-right corner and two target marks

Namespace AddBarcodeToPdfDocument
    Class Program
        Friend Shared Sub Main(args As String())

            ' Create Barcode instance and set it up
            Dim barcode As New Barcode("demo", "demo")
            barcode.Symbology = SymbologyType.Code39
            barcode.Value = "00090112"
            barcode.DrawCaption = True
            barcode.CaptionFont = New Font("Courier", 12.0F, FontStyle.Bold, GraphicsUnit.Pixel)
            barcode.DrawQuietZones = False
            barcode.ResolutionX = 300 ' High resolution for better quality on document
            barcode.ResolutionY = 300

            ' Compute barcode image dimension from inches to pixels at 300 DPI:

            Dim barHeight As Integer = CInt(Math.Truncate(3.0F / 8 * 300)) ' = 3/8" * 300 DPI = height of barcode
            Dim captionHeight As Integer = CInt(Math.Truncate(barcode.CaptionFont.GetHeight(1))) ' = height of caption
            Dim captionGap As Integer = CInt(Math.Truncate(1.0F / 8 * 300)) ' = 1/8" gap between barcode and caption

            Dim barcodeImageWidth As Integer = 2 * 300 ' = 2" * 300 DPI = width of barcode image
            Dim barcodeImageHeight As Integer = barHeight + captionGap + captionHeight

            ' Get final barcode image:

            barcode.BarHeight = barHeight
            barcode.FitInto(barcodeImageWidth, barcodeImageHeight, UnitOfMeasure.Pixel)
            Dim barcodeImage As Image = barcode.GetImage()

            ' Arrays of images and points to apply to PDF document
            Dim images As Image() = New Image(2) {}
            Dim points As Point() = New Point(2) {}

            ' Compute coordinates of barcode image and target marks at 72 DPI
            ' (page size is 8.5" x 11", PDF document resolution is always 72 DPI):

            Dim x As Integer = CInt(Math.Truncate(8.5F * 72 - 2 * 72 - barcodeImageWidth)) ' = X coordinate to put the barcode image
            Dim y As Integer = 2 * 72 ' = 2" - Y coordinate to put the barcode image

            ' Put barcode image into array
            images(0) = barcodeImage
            points(0) = New Point(x, y)

            ' Load target mark image.
            ' TargetMark.png is 1/8" 300 DPI image (38x38 pixels)
```

```

Dim targetMarkImage As Image = Image.FromFile("TargetMark.png")

' Coordinates of top-right target mark
x = CInt(Math.Truncate(8.5F * 72 - 1 * 72 - 1.0F / 8 * 72)) ' = pageWidth
y = 1 * 72 ' = 1" - Y coordinate to put the barcode image

' Put first target mark image into array
images(1) = targetMarkImage
points(1) = New Point(x, y)

' Coordinates of bottom-left target mark
x = 1 * 72 ' = 1"
y = CInt(Math.Truncate(11.0F * 72 - 1 * 72 - 1.0F / 8 * 72)) ' = page height

' Put second target mark image into array
images(2) = targetMarkImage
points(2) = New Point(x, y)

' Draw images on all PDF document pages.
barcode.DrawImagesToPDF("wikipedia.pdf", -1, images, points, "result.pdf")

' Cleanup
barcodeImage.Dispose()
targetMarkImage.Dispose()
barcode.Dispose()

' Open the result document in default associated application
Process.Start("result.pdf")

End Sub

End Class

End Namespace

```

---

## 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](http://www.bytescout.com)