

## composed barcode in VB.NET and ByteScout Barcode SDK

### Simple tutorial on how to do composed barcode in VB.NET

Writing of the code to composed barcode in VB.NET can be done by developers of any level using ByteScout Barcode SDK. ByteScout Barcode SDK was created to assist composed barcode in VB.NET. ByteScout Barcode SDK is 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.

Save time on writing and testing code by using the code below and use it in your application. If you want to implement this functionality, you should copy and paste code below into your app using code editor. Then compile and run your application. These VB.NET sample examples can be used in one or many applications.

Trial version can be downloaded from our website for free. It contains this and other source code samples for VB.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:

```
Imports System.Drawing
Imports Bytescout.BarCode

Module Module1

    Sub Main()

        Using barcode As New Barcode("demo", "demo")

            barcode.Symbology = SymbologyType.Code128

            barcode.Margins = New Margins(0, 0, 0, 0)
            barcode.DrawQuietZones = False

            barcode.Value = "123123"
            Dim image1 As Image = barcode.GetImage()

            barcode.Value = "123"
            Dim image2 As Image = barcode.GetImage()

            Using composer As New ImageComposer(10, 5, CompositionMode.ArrangeHorizontally)

                composer.AddImage(image1)
                composer.AddImage(image2, 0, 0, 90)

                composer.SaveComposedImage("result.png")

                ' Show image in default image viewer
                Process.Start("result.png")

            End Using
        End Using
    End Sub

End Module
```

---

## 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)