

printing barcodes with barcode sdk in Visual Basic 6 using ByteScout Premium Suite

How to apply ByteScout Premium Suite for printing barcodes with barcode sdk in Visual Basic 6

This page explains the steps and algorithm of implementing printing barcodes with barcode sdk and how to make it work in your application. ByteScout Premium Suite was made to help with printing barcodes with barcode sdk in Visual Basic 6. ByteScout Premium Suite is the bundle that includes twelve SDK products from ByteScout including tools and components for PDF, barcodes, spreadsheets, screen video recording.

Visual Basic 6 code snippet like this for ByteScout Premium Suite works best when you need to quickly implement printing barcodes with barcode sdk in your Visual Basic 6 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. Enjoy writing a code with ready-to-use sample Visual Basic 6 codes to implement printing barcodes with barcode sdk using ByteScout Premium Suite.

On our website you may get trial version of ByteScout Premium Suite for free. Source code samples are included to help you with your Visual Basic 6 application.

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](http://www.ByteScout.com)

Source Code Files:

```

VERSION 5.00
Begin VB.Form Form1
    AutoRedraw      = -1 'True
    Caption         = "Form1"
    ClientHeight    = 2175
    ClientLeft      = 45
    ClientTop       = 435
    ClientWidth     = 7770
    LinkTopic       = "Form1"
    ScaleHeight     = 2175
    ScaleWidth      = 7770
    StartupPosition = 3 'Windows Default
    Begin VB.CommandButton Command1
        Caption      = "Print Barcode to the selected printer with 300 DPI quality"
        Height       = 855
        Left         = 120
        TabIndex     = 1
        Top          = 840
        Width        = 6975
    End
    Begin VB.ComboBox cboPrinterList
        Height       = 315
        Left         = 120
        TabIndex     = 0
        Top          = 360
        Width        = 6975
    End
End
Attribute VB_Name = "Form1"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False

Private Sub Command1_Click()
    ' set printer to the selected printer
    Call SelectPrinter(cboPrinterList.Text)
    ' call function to print barcode to Printer object
    Call PrintBarCode
End Sub

' populate list of printers on form load
Private Sub Form_Load()

Dim dev As Printer, Index As Integer, CurrentPrinter As Integer
CurrentPrinter = -1
Index = 0
For Each dev In Printers
cboPrinterList.AddItem dev.DeviceName
If Printer.DeviceName = dev.DeviceName Then
CurrentPrinter = Index
End If
    Index = Index + 1
Next

```

```

If CurrentPrinter <> -1 Then
    cboPrinterList.ListIndex = CurrentPrinter
End If

End Sub

'=====
'Function to select printer
'=====

Function SelectPrinter(ByVal printer_name As String) As Boolean
    Dim i As Integer

    SelectPrinter = False
    For i = 0 To Printers.Count - 1
        ' if the specified printer is found, select it and return True
        If Printers(i).DeviceName = printer_name Then
            Set Printer = Printers(i)
            SelectPrinter = True
            Exit For
        End If
    Next i
End Function

' print barcode to the Printer object

Private Sub PrintBarCode()

' this method prints a barcode via temporary EMF metafile allowing to set target physical resolution

Const DPIQuality = 300 ' 300 DPI as printing resolution (DPI) quality
Const WidthInInches = 4 ' 4 inches as width for the barcode generated
Const HeightInInches = 1 ' 1 inches as height for the barcode generated

' create barcode object as ActiveX
Dim BC
Set BC = CreateObject("Bytescout.BarCode.Barcode")

Printer.PrintQuality = DPIQuality ' set the printer quality to 300 dpi

' IMPORTANT: call Printer.Print Space(1) to initialize the Printer
' this is how Microsoft advises to do in their KB article (search for Printer.Print Space(1))
Printer.Print Space(1) ' initialize hDC of Printer object

' set the same DPI resolution (300 DPI) to generate barcode image
BC.ResolutionX = DPIQuality
BC.ResolutionY = DPIQuality

' set barcode symbology type
BC.Symbology = 0 ' Code 39 = 0 (see the documentation for other barcode types)

' set barcode value
BC.Value = "1234567890"

' use .FitInto_3 method to set the size in inches (you can also set in mm, cm, pixels)
BC.FitInto_3 WidthInInches, HeightInInches, 2 ' fit into 3.5x0.875 inches (2 = inch)

' save barcode into temporary EMF image

```

```

' create FileSystemObject to work with files
Dim FSO As Object
Set FSO = CreateObject("Scripting.filesystemobject")

' get temporary filename to save a temporary .EMF file
Dim tempEMFFileName
tempEMFFileName = FSO.GetTempName() & ".EMF"

' save EMF metafile into the temporary file
BC.SaveImage tempEMFFileName

' now we paint EMF file to Printer object
' default units for Printer in VB6 is Twips, which is 1440 per Inch
' so we calculate 3.5 inches as 3.5 * TwipsPerInch
Const TwipsPerInch = 1440
' now printing the picture at 0,0 with 3.5 inches X 0.875 inches
Printer.PaintPicture LoadPicture(tempEMFFileName), 0, 200, 3.5 * TwipsPerInch, 0.875 *

' now remove temporary EMF file
FSO.deletefile tempEMFFileName, True
Set FSO = Nothing

' finally send command to print the page
Printer.EndDoc

Set BC = Nothing

End Sub

```

Project1.vbp

```

Type=Exe
Reference=*\\G{00020430-0000-0000-C000-000000000046}#2.0#0#C:\WINDOWS\system32\stdole2.t
Form=Form1.frm
Startup="Form1"
ExeName32="Project1.exe"
Command32=""
Name="Project1"
HelpContextID="0"
CompatibleMode="0"
MajorVer=1
MinorVer=0
RevisionVer=0
AutoIncrementVer=0
ServerSupportFiles=0
VersionCompanyName="000"
CompilationType=0
OptimizationType=0

```

```
FavorPentiumPro(tm)=0
CodeViewDebugInfo=0
NoAliasing=0
BoundsCheck=0
OverflowCheck=0
FPointCheck=0
FDIVCheck=0
UnroundedFP=0
StartMode=0
Unattended=0
Retained=0
ThreadPerObject=0
MaxNumberOfThreads=1
```

```
[MS Transaction Server]
AutoRefresh=1
```

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](http://www.ByteScout.com)

[visit www.PDF.co](http://www.PDF.co)

