

How to use drawing transformations for PDF in VB.NET and ByteScout PDF SDK

Write code in VB.NET to use drawing transformations for PDF with this step-by-step tutorial

Learn how to use drawing transformations for PDF in VB.NET with this source code sample. Want to use drawing transformations for PDF in your VB.NET app? ByteScout PDF SDK is designed for it. ByteScout PDF SDK is the component to help programmers in generating new pdf files, modifying and updating existing pdf documents or pdf forms. Provides support for auto-filling pdf forms, adding text with adjustable font, style, size, font family, new form fields, vector and raster drawings.

VB.NET code samples for VB.NET developers help to speed up coding of your application when using ByteScout PDF SDK. Follow the instructions from the scratch to work and copy the VB.NET code. Use of ByteScout PDF SDK in VB.NET is also explained in the documentation included along with the product.

Trial version of ByteScout PDF SDK can be downloaded for free from our website. It also includes source code samples for VB.NET and other programming languages.

VB.NET - Program.vb

```
Imports Bytescout.PDF
'''
''' This example demonstrates how to use coordinate space transformations.
'''

Class Program

    Shared Sub Main()

        ' Create new document
        Dim pdfDocument = New Document()
        pdfDocument.RegistrationName = "demo"
        pdfDocument.RegistrationKey = "demo"

        ' Add page
        Dim page = New Page(PaperFormat.A4)
        pdfDocument.Pages.Add(page)

        Dim canvas = page.Canvas

        ' Prepare pens
        Dim blackPen = New SolidPen(New ColorRGB(0, 0, 0), 2.0F)
        blackPen.Opacity = 50
```

```
Dim bluePen = New SolidPen(New ColorRGB(0, 0, 255), 2.0F)
bluePen.Opacity = 50
Dim greenPen = New SolidPen(New ColorRGB(0, 255, 0), 2.0F)
greenPen.Opacity = 50

' Move coordinate space zero point to (200, 200) and draw a rectangle of
50x50 size
canvas.TranslateTransform(200, 200)
canvas.DrawRectangle(blackPen, 0, 0, 50, 50)

' Now zoom the coordinate space in twice and draw the same rectangle in green
canvas.ScaleTransform(2, 2)
canvas.DrawRectangle(greenPen, 0, 0, 50, 50)

' Now rotate the coordinate space by 30 degrees and draw the same rectangle
in blue
canvas.RotateTransform(30)
canvas.DrawRectangle(bluePen, 0, 0, 50, 50)

' Save document to file
pdfDocument.Save("result.pdf")

' Cleanup
pdfDocument.Dispose()

' Open document in default PDF viewer app
Process.Start("result.pdf")

End Sub

End Class
```

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout PDF SDK](#)

[Explore documentation](#)

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

or

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