

transition effects in VB.NET using ByteScout Image To Video SDK

Write code in VB.NET to make transition effects with this How-To tutorial

Today we will explain the steps and algorithm of implementing transition effects and how to make it work in your application. Transition effects in VB.NET can be implemented with ByteScout Image To Video SDK. ByteScout Image To Video SDK is the SDK that is designed to generate video slideshow with 100+ 2d and 3D transition effects. Can generate WMV, AVI, WEBM video file with adjustable quality, framerate, output video size.

VB.NET, code samples for VB.NET, developers help to speed up the application development and writing a code when using ByteScout Image To Video SDK. VB.NET sample code is all you need: copy and paste the code to your VB.NET application's code editor, add a reference to ByteScout Image To Video SDK (if you haven't added yet) and you are ready to go! Use of ByteScout Image To Video SDK in VB.NET is also explained in the documentation included along with the product.

On our website you may get trial version of ByteScout Image To Video SDK for free. Source code samples are included to help you with your VB.NET application.

VB.NET - Module1.vb

```
Imports ByteScoutImageToVideo

Module Module1

    Sub Main()

        Try

            Console.WriteLine("Converting JPG slides to video, please wait...")

            ' Create ByteScoutImageToVideoLib.ImageToVideo object instance
            Dim converter As ImageToVideo = New ImageToVideo()

            ' Activate the component
            converter.RegistrationName = "demo"
            converter.RegistrationKey = "demo"

            ' Enable transition effects for the first and last slide
            converter.UseInEffectForFirstSlide = True
            converter.UseOutEffectForLastSlide = True

            ' Add images and set slide durations and transition effects
            Dim slide As Slide
            slide = converter.AddImageFromFileName("../..\\..\\..\\slide1.jpg")
            slide.Duration = 3000 ' 3000ms = 3s
```

```

slide.InEffect = TransitionEffectType.teFade
slide.OutEffect = TransitionEffectType.teFade

slide = converter.AddImageFromFileName("../..\..\..\slide2.jpg")
slide.Duration = 3000
slide.InEffect = TransitionEffectType.teWipeLeft
slide.OutEffect = TransitionEffectType.teWipeRight

slide = converter.AddImageFromFileName("../..\..\..\slide3.jpg")
slide.Duration = 3000
slide.InEffect = TransitionEffectType.teWipeLeft
slide.OutEffect = TransitionEffectType.teWipeRight

' Set output video size
converter.OutputWidth = 640
converter.OutputHeight = 480

' Set output video file name
converter.OutputVideoFileName = "result.wmv"

' Run the conversion
converter.RunAndWait()

' Release resources
System.Runtime.InteropServices.Marshal.ReleaseComObject(converter)

' Open the result video file in default media player
Process.Start("result.wmv")

Console.WriteLine("Done. Press any key to continue...")
Console.ReadKey()

Catch ex As Exception
    Console.WriteLine("Error: " & ex.Message)
    Console.ReadKey()
End Try

End Sub

End Module

```

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout Image To Video SDK](#)

[Explore documentation](#)

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

or

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