

## How to record screen video from command line console in VB.NET with ByteScout Screen Capturing SDK

How to code in VB.NET to record screen video from command line console with this step-by-step tutorial

These source code samples are listed and grouped by their programming language and functions they use. ByteScout Screen Capturing SDK is the screen video recording SDK helps in quick implementation of screen video recording. WMV, AVI, WebM output options are available with adjustable quality, video size, framerate and video and audio codec. Includes special features like live multiple blacking out of selected areas, recording from web cam as main source and as overlay, optional watermarks for output video and you can use it to record screen video from command line console with VB.NET.

VB.NET code samples for VB.NET developers help to speed up coding of your application when using ByteScout Screen Capturing SDK. Follow the instructions from the scratch to work and copy the VB.NET code. Implementing VB.NET application typically includes multiple stages of the software development so even if the functionality works please test it with your data and the production environment.

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

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout Screen Capturing SDK](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout Screen Capturing SDK](#)

[Get Free API key for Web API](#)

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

Source Code Files:

## Module1.vb

```
Imports System
Imports System.Collections.Generic
Imports System.Text
Imports System.Globalization
Imports System.Threading
Imports System.Runtime.InteropServices

Imports BytescoutScreenCapturingLib

' NOTE: if you are getting error like "invalid image" related to loading the SDK's dll
' try to do the following:
' 1) remove the reference to the SDK by View - Solution Explorer
' then click on References, select Bytescout... reference name and right-click it and s
' 2) To re-add click on the menu: Project - Add Reference
' 3) In "Add Reference" dialog switch to "COM" tab and find Bytescout...
' 4) Select it and click "Add"
' 5) Recompile the application
' Note: if you need to run on both x64 and x86 then please make sure you have set "Embe

Module Module1
    Public Class Win32Interop
        <DllImport("crtddll.dll")> _
        Public Shared Function _kbhit() As Integer
        End Function
    End Class

    Sub Main(ByVal args As String())
        Dim capturer As New Capturer()

        If args.Length < 1 Then
            usage(capturer)
            Exit Sub
        End If

        capturer.OutputFileName = args(0)
        capturer.CapturingType = CaptureAreaType.catRegion

        ' set border style
        capturer.CaptureAreaBorderType = CaptureAreaBorderType.cabtDashed

        ' uncomment to enable recording of semitransparent or layered windows (Warning
        ' capturer.CaptureTransparentControls = true

        ' // WMV and WEBM output use WMVVideoBitrate property to control output video b
        ' // so try to increase it by x2 or x3 times if you think the output video are
        ' capturer.WMVVideoBitrate = capturer.WMVVideoBitrate * 2

        setParams(args, capturer)

        Try
            capturer.Run()
```

```

' IMPORTANT: if you want to check for some code if need to stop the recording
' using Thread.Sleep(1) inside the checking loop, so you have the loop like
' Do
' Thread.Sleep(1)
' While StopButtonNotClicked

    Console.WriteLine("Starting capture - Hit a key to stop ...")

    Dim s As String = capturer.CurrentVideoCodecName
    Console.WriteLine(String.Format("Using video compressor - {0}", s))

    s = capturer.CurrentAudioCodecName
    Console.WriteLine(String.Format("Using audio compressor - {0}", s))

    s = capturer.CurrentAudioDeviceLineName
    Console.WriteLine(String.Format("Using audio input line - {0}", s))

    Dim i As Integer = 0
    Dim spin As String = "|/-\"
    While Win32Interop._kbhit() = 0
        Console.Write(String.Format(vbCr & "Encoding {0}", spin(i)))
        i = i + 1
        i = i Mod 4
        Thread.Sleep(50)
    End While

    capturer.Stop()

' Release(Resources)
System.Runtime.InteropServices.Marshal.ReleaseComObject(capturer)
capturer = Nothing

    Console.Write(vbLf & "Done")
    Console.Read()
Catch generatedExceptionName As Exception
    Console.WriteLine(capturer.LastError)
End Try
End Sub

Private Sub usage(ByVal capturer As Capturer)
    Console.WriteLine("Usage : CaptureScreenCSharp.exe <outfilename> [left] [top] [
    Console.WriteLine("[left, top, width, height] is the rectangle to be captured")
    Console.WriteLine("[v-codec] is the index of the video codec in the list to use
    Console.WriteLine("[a-codec] is the index of the audio codec in the list to use
    Console.WriteLine("[audioline] is the index of the audio line in the list to c
    Console.WriteLine("If either codec is unspecified, it defaults to 'Microsoft V
    Console.WriteLine("If audioline is unspecified, it uses the microphone")
    Console.WriteLine("To capture the currently playing output, select the stereo,

    Console.WriteLine("Installed Video Codecs (Note : Not all of them may work)")

    For i As Integer = 0 To capturer.VideoCodecsCount - 1
        Dim name As String = capturer.GetVideoCodecName(i)
        Console.WriteLine(String.Format("    {0}. {1}", i, name))
    Next

    Console.WriteLine(vbLf & "Installed Audio Codecs (Note : Not all of them may wo
    For i As Integer = 0 To capturer.AudioCodecsCount - 1
        Dim name As String = capturer.GetAudioCodecName(i)

```

```

        Console.WriteLine(String.Format("    {0}. {1}", i, name))
    Next

    Console.WriteLine(vbLf & "Audio input lines")
    For i As Integer = 0 To capturer.CurrentAudioDeviceLineCount - 1
        Dim name As String = capturer.GetCurrentAudioDeviceLineName(i)
        Console.WriteLine(String.Format("    {0}. {1}", i, name))
    Next
End Sub

Private Sub setParams(ByVal args As String(), ByVal capturer As Capturer)
    If args.Length > 1 Then
        Dim left As Integer = Integer.Parse(args(1))
        capturer.CaptureRectLeft = left
    End If

    If args.Length > 2 Then
        Dim top As Integer = Integer.Parse(args(2))
        capturer.CaptureRectTop = top
    End If

    If args.Length > 3 Then
        Dim width As Integer = Integer.Parse(args(3))
        capturer.CaptureRectWidth = width
    End If

    If args.Length > 5 Then
        Dim height As Integer = Integer.Parse(args(5))
        capturer.CaptureRectHeight = height
    End If

    If args.Length > 6 Then
        Dim fps As Single = Single.Parse(args(6), CultureInfo.InvariantCulture)
        capturer.FPS = fps
    End If

    If args.Length > 7 Then
        Dim vCodec As Integer = Integer.Parse(args(7))
        capturer.CurrentVideoCodec = vCodec
    End If

    If args.Length > 8 Then
        Dim aCodec As Integer = Integer.Parse(args(8))
        capturer.CurrentAudioCodec = aCodec
    End If

    If args.Length > 9 Then
        Dim aLine As Integer = Integer.Parse(args(9))
        capturer.CurrentAudioDeviceLine = aLine
    End If
End Sub

End Module

```

---

## VIDEO

<https://www.youtube.com/watch?v=fujkvtWUVCw>

## ON-PREMISE OFFLINE SDK

[60 Day Free Trial](#) or [Visit ByteScout Screen Capturing SDK Home Page](#)  
[Explore ByteScout Screen Capturing SDK Documentation](#)  
[Explore Samples](#)  
[Sign Up for ByteScout Screen Capturing 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)