

How to capture given window and stop video on window closed in VBScript and VB6 using ByteScout Screen Capturing SDK

This tutorial will show how to capture given window and stop video on window closed in VBScript and VB6

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. It can capture given window and stop video on window closed in VBScript and VB6.

You will save a lot of time on writing and testing code as you may just take the VBScript and VB6 code from ByteScout Screen Capturing SDK for capture given window and stop video on window closed below and use it in your application. Just copy and paste the code into your VBScript and VB6 application's code and follow the instruction. This basic programming language sample code for VBScript and VB6 will do the whole work for you to capture given window and stop video on window closed.

ByteScout free trial version is available for download from our website. It includes all these programming tutorials along with source code samples.

VBScript and VB6 - CaptureGivenWindowAndStopIfWindowClosed.vbs

```
' create video capturer activex object
Set capturer = CreateObject("BytescoutScreenCapturing.Capturer")

' set output video file name (WMV), you can also set to AVI
capturer.OutputFileName = "WindowCaptured.wmv"

' set capturing type to the caVisibleWindow=4 (captures from the window) or
caWindow=1 (may flicker but captures from non-visible windows as well)
capturer.CapturingType = 4

MsgBox "This script will record video from Internet Explorer for 60 seconds OR will
stop if Window was closed." & vbCRLF & vbCRLF & "Please run Internet Explorer and
click OK"

capturer.WindowToCapture = "Internet Explorer"

' set width of the output video
capturer.OutputWidth = 640
' set height of the output video
capturer.OutputHeight = 480
```

```
' set border type to show around captured area
capturer.CaptureAreaBorderType = 2 ' CaptureAreaBorderType.cabtDashed;

' uncomment to enable recording of semitransparent or layered windows (Warning: may
cause mouse cursor flickering)
' capturer.CaptureTransparentControls = true

' uncomment to set another output video compression method, Bytescout Lossless, DivX
or MJPEG (if installed)
' you can use lossless video format, Bytescout Lossless Video Codec
' capturer.CurrentVideoCodecName = "Bytescout Lossless"

' run video capturing
capturer.Run()

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

' count number of seconds actually recorded
SecondsActuallyCaptured = 0

' wait for 60 seconds, checking if window is still available every 1 second
for i=0 to 60
  WScript.Sleep(1000) ' wait for 1 second
  SecondsActuallyCaptured = SecondsActuallyCaptured + 1
  if Not capturer.WindowToCaptureIsAvailable Then ' check if window to capture is
still available
    Exit For ' break if the window to capture from is not available anymore
  End If
Next

' stop capturing
capturer.Stop()

MsgBox "Recorded " & CStr(SecondsActuallyCaptured) & " second(s)"

' destroy Capturer object so the video will be saved into the disk
Set capturer = Nothing
```

FOR MORE INFORMATION AND FREE TRIAL:

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

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

[Explore documentation](#)

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

or

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