

How to record screen from command line console in C# with ByteScout Screen Capturing SDK

This tutorial will show how to record screen from command line console in C#

Learn how to record screen from command line console in C# with this source code sample. ByteScout Screen Capturing SDK is the tool for developers who want to add screen capturing in their application. Can record screen into video and into single screenshots. Output formats are WMV, AVI, WebM for video and PNG for screenshots. You can adjust output video size, quality, resolution, framerate, video and audio codecs. Includes special privacy features for blacking out sensitive information on screen. Can also capture video from web camera, can add overlays with text or images. It can be used to record screen from command line console using C#.

The SDK samples like this one below explain how to quickly make your application do record screen from command line console in C# with the help of ByteScout Screen Capturing SDK. In order to implement the functionality, you should copy and paste this code for C# below into your code editor with your app, compile and run your application. Code testing will allow the function to be tested and work properly with your data.

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

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](#)

Source Code Files:

Program.cs

```
using System;
using System.Collections.Generic;
using System.Text;
using System.Globalization;
using System.Threading;
using System.Runtime.InteropServices;
using System.Drawing;
using 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
// 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 "Embed all referenced assemblies" in project properties

namespace CaptureScreenCSharp
{
    public class Win32Interop
    {
        [DllImport("crtdll.dll")]
        public static extern int _kbhit();
    }

    class Program
    {
        static void Main(string[] args)
        {
            Capturer capturer = new Capturer(); // create new screen capturer object

            if (args.Length < 1)
            {
                usage(capturer);
                return;
            }

            capturer.OutputFileName = args[0];
            capturer.CapturingType = CaptureAreaType.catRegion;
            setParams(args, capturer);

            // set border around captured area if we are not capturing entire screen
            if (
                capturer.CapturingType != CaptureAreaType.catScreen &&
                capturer.CapturingType != CaptureAreaType.catWebcamFullScreen
            )
            {
                // set border style
```

```

        capturer.CaptureAreaBorderType = CaptureAreaBorderType.cabtDashed;
        capturer.CaptureAreaBorderColor = (uint)ColorTranslator.ToOle(Color.Red);
    }

    // uncomment to enable recording of semitransparent or layered windows (Windows 7+)
    // capturer.CaptureTransparentControls = true;

    // WMV and WEBM output use WMVVideoBitrate property to control output video quality
    // so try to increase it by x2 or x3 times if you think the output video is too low quality
    // capturer.WMVVideoBitrate = capturer.WMVVideoBitrate * 2;

    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 ...");

        string s = 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));

        int i = 0;
        string spin = "|/-\\";
        while (Win32Interop._kbhit() == 0)
        {
            Console.Write(string.Format("\rEncoding {0}", spin[i++]));
            i %= 4;
            Thread.Sleep(50);
        }

        capturer.Stop();

        // Release resources
        System.Runtime.InteropServices.Marshal.ReleaseComObject(capturer);
        capturer = null;

        Console.Write("\nDone");
        Console.Read();
    }
    catch (Exception)
    {
        Console.WriteLine(capturer.LastError);
    }
}

static void usage(Capturer capturer)
{

```

```
Console.WriteLine("Usage : CaptureScreenCSharp.exe <outfilename> [left] [top] [width] [height] [v-codec] [a-codec] [audioline]");
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 use");
Console.WriteLine("If either codec is unspecified, it defaults to 'Microsoft DV' or 'Microsoft AAC'");
Console.WriteLine("If audioline is unspecified, it uses the microphone");
Console.WriteLine("To capture the currently playing output, select the stereo line");

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

for (int i = 0; i < capturer.VideoCodecsCount; i++)
{
    string name = capturer.GetVideoCodecName(i);
    Console.WriteLine(string.Format("{0}. {1}", i, name));
}

Console.WriteLine("\nInstalled Audio Codecs (Note : Not all of them may work)")
for (int i = 0; i < capturer.AudioCodecsCount; i++)
{
    string name = capturer.GetAudioCodecName(i);
    Console.WriteLine(string.Format("{0}. {1}", i, name));
}

Console.WriteLine("\nAudio input lines");
for (int i = 0; i < capturer.CurrentAudioDeviceLineCount; i++)
{
    string name = capturer.GetCurrentAudioDeviceLineName(i);
    Console.WriteLine(string.Format("{0}. {1}", i, name));
}

static void setParams(string[] args, Capturer capturer)
{
    if (args.Length > 1)
    {
        int left = int.Parse(args[1]);
        capturer.CaptureRectLeft = left;
    }

    if (args.Length > 2)
    {
        int top = int.Parse(args[2]);
        capturer.CaptureRectTop = top;
    }

    if (args.Length > 3)
    {
        int width = int.Parse(args[3]);
        capturer.CaptureRectWidth = width;
    }

    if (args.Length > 5)
    {
        int height = int.Parse(args[5]);
        capturer.CaptureRectHeight = height;
    }

    if (args.Length > 6)
    {

```

```
        float fps = float.Parse(args[6], CultureInfo.InvariantCulture);
        capturer.FPS = fps;
    }

    if (args.Length > 7)
    {
        int vCodec = int.Parse(args[7]);
        capturer.CurrentVideoCodec = vCodec;
    }

    if (args.Length > 8)
    {
        int aCodec = int.Parse(args[8]);
        capturer.CurrentAudioCodec = aCodec;
    }

    if (args.Length > 9)
    {
        int aLine = int.Parse(args[9]);
        capturer.CurrentAudioDeviceLine = aLine;
    }
}
}
```

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