

PDF to image in command line in VBScript and VB6 using ByteScout PDF Renderer SDK

How to use ByteScout PDF Renderer SDK for PDF to image in command line in VBScript and VB6

On this page you will learn from code samples for programming in VBScript and VB6. ByteScout PDF Renderer SDK was made to help with PDF to image in command line in VBScript and VB6. ByteScout PDF Renderer SDK is the SDK for rendering of PDF into thumbnails and images in high quality. Provides various functions like batch processing, PNG, TIFF output. Can be used from web and desktop applications.

You will save a lot of time on writing and testing code as you may just take the code below and use it in your application. This VBScript and VB6 sample code should be copied and pasted into your application's code editor. Then just compile and run it to see how it works. Enjoy writing a code with ready-to-use sample VBScript and VB6 codes to implement PDF to image in command line using ByteScout PDF Renderer SDK.

Trial version can be downloaded from our website. Source code samples for VBScript and VB6 and documentation are included.

VBScript and VB6 - PDF_to_Image.bat

```
@echo off
REM Run VBScript through the command line
REM The script convert first pages of all PDF documents in the outer folder
REM containing sample files (..\..)
REM and puts generated images to "output" sub-folder.
cscript.exe "PDF_to_Image.vbs" ..\.. output

pause
```

VBScript and VB6 - PDF_to_Image.vbs

```
if WScript.Arguments.Length < 2 Then
    WScript.Echo "Usage: cscript PDF_to_Image.vbs "
    WScript.Quit
```

```

End If

' Create Bytescout.PDFRenderer.RasterRenderer object
Set renderer = CreateObject("Bytescout.PDFRenderer.RasterRenderer")
renderer.RegistrationName = "demo"
renderer.RegistrationKey = "demo"

' Render PDF document at 96 DPI - default PC display resolution
' To get higher quality output, set 200, 300 or more
Dim renderingResolution
renderingResolution = 96

' Image format: 0 - BMP; 1 - JPEG; 2 - PNG; 3 - TIFF; 4 - GIF
Dim outputImageFormat
outputImageFormat = 2

Dim pageIndex
pageIndex = 0

' Create File System object
Set FSO = CreateObject("Scripting.FileSystemObject")

' Get folder object (current folder)
Set inputFolder = FSO.GetFolder(WScript.Arguments.Item(0))

' Get file list
Set files = inputFolder.Files

' Prepare output folder
outputFolder = WScript.Arguments.Item(1)
If FSO.FolderExists(outputFolder) Then
    FSO.DeleteFolder outputFolder, True
End If
FSO.CreateFolder outputFolder

' Run barcode search for PDF and TIFF files
For Each file in files

    ext = UCase(FSO.GetExtensionName(file))
    If ext = "PDF" Then

        WScript.Echo "Processing file " & FSO.GetFileName(file)

        renderer.LoadDocumentFromFile file

        fileName = FSO.BuildPath(WScript.Arguments.Item(1),
FSO.GetFileName(file) & ".page0.png")
        renderer.Save fileName, outputImageFormat, pageIndex,
renderingResolution

        WScript.Echo " first page converted to " & fileName

    End If
Next

WScript.Echo "Done."

Set renderer = Nothing

```

```
Set FS0 = Nothing
```

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout PDF Renderer SDK](#)

[Explore documentation](#)

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

or

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