

How to extract pages from PDF in VBScript and ByteScout PDF Extractor SDK

How to extract pages from PDF in VBScript

On this page you will learn from code samples for programming in VBScript. Writing of the code to extract pages from PDF in VBScript can be done by developers of any level using ByteScout PDF Extractor SDK. ByteScout PDF Extractor SDK: the SDK is designed to help developers with pdf tables and pdf data extraction from unstructured documents like pdf, tiff, scans, images, scanned and electronic forms. The library is powered by OCR, computer vision and AI to provide unique functionality like table detection, automatic table structure extraction, data restoration, data restructuring and reconstruction. Supports PDF, TIFF, PNG, JPG images as input and can output CSV, XML, JSON formatted data. Includes full set of utilities like pdf splitter, pdf merger, searchable pdf maker and other utilities. It can extract pages from PDF in VBScript.

VBScript code samples for VBScript developers help to speed up coding of your application when using ByteScout PDF Extractor SDK. In your VBScript project or application you may simply copy & paste the code and then run your app! Enjoy writing a code with ready-to-use sample codes in VBScript.

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 PDF Extractor SDK](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout PDF Extractor SDK](#)

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

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

Source Code Files:

SplitPDF.vbs

```
' Create Bytescout.PDFExtractor.DocumentSplitter object
Set splitter = CreateObject("Bytescout.PDFExtractor.DocumentSplitter")
splitter.RegistrationName = "demo"
splitter.RegistrationKey = "demo"

inputFile = "sample.pdf"

' Enable optimization of output documents
splitter.OptimizeSplittedDocuments = true

splitter.ExtractPage inputFile, "page3.pdf", 3 ' (!) Note: page number is 1-based.
WScript.Echo "Extracted page 3 to file page3.pdf"

' Split in two parts:
' =====

splitter.Split inputFile, "part1.pdf", "part2.pdf", 3 ' (!) Note: page number is 1-based
WScript.Echo "Splitted at page 3 to files part1.pdf and part2.pdf"

' Split by ranges:
' =====

' SplitCOM() returns array with a list of filenames
splittedParts = splitter.SplitCOM(inputFile, "1-3,4-6,7,8-") ' (!) Note: page numbers are 1-based
WScript.Echo "Splitted by ranges sucessfully"

' Display list of generated outputfiles
For Each file in splittedParts
    Wscript.Echo file
Next
```

VIDEO

https://www.youtube.com/watch?v=s28W3_KMraU

ON-PREMISE OFFLINE SDK

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