

How to split PDF document in VBScript using ByteScout PDF Extractor SDK

This tutorial will show how to split PDF document in VBScript

This sample source code below will demonstrate you how to split PDF document in VBScript. ByteScout PDF Extractor SDK is the SDK that helps developers to extract data from unstructured documents, pdf, images, scanned and electronic forms. Includes AI functions like automatic table detection, automatic table extraction and restructuring, text recognition and text restoration from pdf and scanned documents. Includes PDF to CSV, PDF to XML, PDF to JSON, PDF to searchable PDF functions as well as methods for low level data extraction. It can be used to split PDF document using VBScript.

VBScript code samples for VBScript developers help to speed up coding of your application when using ByteScout PDF Extractor SDK. This VBScript sample code is all you need for your app. Just copy and paste the code, add references (if needs to) and you are all set! Implementing VBScript 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 PDF Extractor SDK is available for free. Source code samples are included to help you with your VBScript app.

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 = "..\..\sample2.pdf"

' Enable splitted parts optimization (remove unused resources)
splitter.OptimizeSplittedDocuments = true

' Extract a single page:
' =====

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 file names. Files are saved to the system temporary directory
Dim outFiles
outFiles = splitter.SplitCOM(inputFile, "1-3,4-") ' (!) Note: page numbers are 1-based

Dim fileNames
For each of in outFiles
    fileNames = fileNames & of & vbCRLF
Next

WScript.Echo "Splitted by ranges to files: " & vbCRLF & fileNames
```

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