

How to split PDF from uploaded file for PDF splitting API in VB.NET and PDF.co Web API

Tutorial: how to split PDF from uploaded file for PDF splitting API in VB.NET

Here you may find thousands pre-made source code pieces for easy implementation in your own programming projects. PDF.co Web API was made to help with PDF splitting API in VB.NET. PDF.co Web API is the flexible Web API that includes full set of functions from e-signature requests to data extraction, OCR, images recognition, pdf splitting and pdf splitting. Can also generate barcodes and read barcodes from images, scans and pdf.

Fast application programming interfaces of PDF.co Web API for VB.NET plus the instruction and the code below will help to learn how to split PDF from uploaded file. Sample code in VB.NET is all you need. Copy-paste it to your the code editor, then add a reference to PDF.co Web API and you are ready to try it! Use of PDF.co Web API in VB.NET is also explained in the documentation included along with the product.

ByteScout free trial version is available for FREE download from our website. Programming tutorials along with source code samples are included.

VB.NET - Module1.vb

```
Imports System.IO
Imports System.Net
Imports Newtonsoft.Json.Linq

Module Module1

    ' The authentication key (API Key).
    ' Get your own by registering at https://app.pdf.co/documentation/api
    Const API_KEY As String = "*****"

    ' Source PDF file to split
    const SourceFile as String = ".\sample.pdf"
    ' Comma-separated list of page numbers (or ranges) to process. Example: '1,3-5,7-'.
    const Pages as String = "1-2,3-"

    Sub Main()

        ' Create standard .NET web client instance
        Dim webClient As WebClient = New WebClient()

        ' Set API Key
        webClient.Headers.Add("x-api-key", API_KEY)
```

```

' 1. RETRIEVE THE PRESIGNED URL TO UPLOAD THE FILE.
' * If you already have a direct file URL, skip to the step 3.

' Prepare URL for `Get Presigned URL` API call
Dim query As string = Uri.EscapeUriString(String.Format(
    "https://api.pdf.co/v1/file/upload/get-presigned-url?
contenttype=application/octet-stream&name={0}",
    Path.GetFileName(SourceFile)))

Try
    ' Execute request
    Dim response As string = webClient.DownloadString(query)

    ' Parse JSON response
    Dim json As JObject = JObject.Parse(response)

    If json("error").ToObject(Of Boolean) = False Then

        ' Get URL to use for the file upload
        Dim uploadUrl As string =
json("presignedUrl").ToString()

        ' Get URL of uploaded file to use with later API
calls
        Dim uploadedFileUrl As string =
json("url").ToString()

        ' 2. UPLOAD THE FILE TO CLOUD.

        webClient.Headers.Add("content-type",
"application/octet-stream")
        webClient.UploadFile(uploadUrl, "PUT", SourceFile) '
You can use UploadData() instead if your file is byte array or Stream

        ' 3. SPLIT UPLOADED PDF

        ' Prepare URL for `Split PDF` API call
        query = Uri.EscapeUriString(String.Format(
            "https://api.pdf.co/v1/pdf/split?pages=
{0}&url={1}",
            Pages,
            uploadedFileUrl))

        ' Execute request
        response = webClient.DownloadString(query)

        ' Parse JSON response
        json = JObject.Parse(response)

        If json("error").ToObject(Of Boolean) = False Then

            ' Download generated PDF files
            Dim part As Integer = 1
            For Each token As JToken In json("urls")

                Dim resultFileUrl As string =
token.ToString()

                Dim localFileName As String =
String.Format(".\part{0}.pdf", part)

                webClient.DownloadFile(resultFileUrl,

```

```
localFileName)

                                Console.WriteLine("Downloaded ""
{0}"".", localFileName)
                                part = part + 1

                                Next

                                Else
                                Console.WriteLine(json("message").ToString())
                                End If

                                End If

                                Catch ex As WebException
                                Console.WriteLine(ex.ToString())
                                End Try

                                webClient.Dispose()

                                Console.WriteLine()
                                Console.WriteLine("Press any key...")
                                Console.ReadKey()

                                End Sub

End Module
```

VB.NET - packages.config

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about PDF.co Web API](#)

[Explore documentation](#)

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

or

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