www.bytescout.com

How to convert PDF to XLS from uploaded file for PDF to excel API in PowerShell and ByteScout Cloud API Server

How to convert PDF to XLS from uploaded file in PowerShell with easy ByteScout code samples to make PDF to excel API. Step-by-step tutorial

The easy to understand coding guides help you check the features without any need to write your own code. ByteScout Cloud API Server was designed to assist PDF to excel API in PowerShell. ByteScout Cloud API Server is API server that is ready to use and can be installed and deployed in less than 30 minutes on your own Windows server or server in a cloud. It can save data and files on your local server-based file storage or in Amazon AWS S3 storage. Data is processed solely on the API server and is powered by ByteScout engine, no cloud services or Internet connection is required for data processing..

If you want to speed up the application's code writing then PowerShell code samples for PowerShell developers help to implement using ByteScout Cloud API Server. This sample code in PowerShell is all you need. Just copy-paste it to the code editor, then add a reference to ByteScout Cloud API Server and you are ready to try it! Use of ByteScout Cloud API Server in PowerShell is also described in the documentation given along with the product.

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

FOR MORE INFORMATION AND FREE TRIAL:

Download Free Trial SDK (on-premise version)

Read more about ByteScout Cloud API Server

Explore API Documentation

Get Free Training for ByteScout Cloud API Server

Get Free API key for Web API

visit www.ByteScout.com

Source Code Files:

ConvertPdfToXlsFromUploadedFile.ps1

```
# Please NOTE: In this sample we're assuming Cloud Api Server is hosted at "https://lo
# If it's not then please replace this with with your hosting url.

# Source PDF file
$SourceFile = ".\sample.pdf"
# Comma-separated list of page indices (or ranges) to process. Leave empty for all pag
$Pages = ""
# PDF document password. Leave empty for unprotected documents.
$Password = ""
# Destination XLS file name
$DestinationFile = ".\result.xls"


# 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
$query = "https://localhost/file/upload/get-presigned-url?contenttype=application/octe
    [System.IO.Path]::GetFileName($SourceFile)
$query = [System.Uri]::EscapeUriString($query)

try {
    # Execute request
    $jsonResponse = Invoke-RestMethod -Method Get  -Uri $query

    if ($jsonResponse.error -eq $false) {
        # Get URL to use for the file upload
        $uploadUrl = $jsonResponse.presignedUrl
        # Get URL of uploaded file to use with later API calls
        $uploadedFileUrl = $jsonResponse.url

        # 2. UPLOAD THE FILE TO CLOUD.

        $r = Invoke-WebRequest -Method Put -Headers @{ "content-type" = "application/o

        if ($r.StatusCode -eq 200) {

            # 3. CONVERT UPLOADED PDF FILE TO XLS

            # Prepare URL for `PDF To XLS` API call
            $query = "https://localhost/pdf/convert/to/xls?name={0}&password={1}&pages=
                $(Split-Path $DestinationFile -Leaf), $Password, $Pages, $uploadedFileU
            $query = [System.Uri]::EscapeUriString($query)

            # Execute request
            $jsonResponse = Invoke-RestMethod -Method Get  -Uri $query

            if ($jsonResponse.error -eq $false) {
                # Get URL of generated XLS file
                $resultFileUrl = $jsonResponse.url;

                # Download XLS file
```

```
                Invoke-WebRequest  -OutFile $DestinationFile -Uri $resultFileUrl

                Write-Host "Generated XLS file saved as `"$($DestinationFile)`" file."
            }
            else {
                # Display service reported error
                Write-Host $jsonResponse.message
            }
        }
        else {
            # Display request error status
            Write-Host $r.StatusCode + " " + $r.StatusDescription
        }
    }
    else {
        # Display service reported error
        Write-Host $jsonResponse.message
    }
}
catch {
    # Display request error
    Write-Host $_.Exception
}
```

run.bat

```
@echo off

powershell -NoProfile -ExecutionPolicy Bypass -Command "& .\ConvertPdfToXlsFromUploade
echo Script finished with errorlevel=%errorlevel%

pause
```

VIDEO

https://www.youtube.com/watch?v=NEwNs2b9YN8

ON-PREMISE OFFLINE SDK

60 Day Free Trial or Visit ByteScout Cloud API Server Home Page
Explore ByteScout Cloud API Server Documentation
Explore Samples
Sign Up for ByteScout Cloud API Server Online Training

ON-DEMAND REST WEB API

Get Your API Key
Explore Web API Docs
Explore Web API Samples

visit www.ByteScout.com

visit www.PDF.co