How to convert PDF to XLS from uploaded file for PDF to excel API in PowerShell using PDF.co Web API

Step By Step Tutorial: how to convert PDF to XLS from uploaded file for PDF to excel API in PowerShell

The sample source codes on this page will demonstrate you how to make PDF to excel API in PowerShell. PDF.co Web API helps with PDF to excel API in PowerShell. PDF.co Web API is the Web API with a set of tools for documents manipulation, data conversion, data extraction, splitting and merging of documents. Includes image recognition, built-in OCR, barcode generation and barcode decoders to decode bar codes from scans, pictures and pdf.

PowerShell code snippet like this for PDF.co Web API works best when you need to quickly implement PDF to excel API in your PowerShell application. For implimentation of this functionality, please copy and paste code below into your app using code editor. Then compile and run your app. Test PowerShell sample code examples whether they respond your needs and requirements for the project.

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

PowerShell - ConvertPdfToXlsFromUploadedFile.ps1

```
# The authentication key (API Key).
# Get your own by registering at https://app.pdf.co/documentation/api
$API_KEY = "***********************************"

# Source PDF file
$SourceFile = ".\sample.pdf"
# Comma-separated list of page indices (or ranges) to process. Leave empty for all
pages. Example: '0,2-5,7-'.
$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
$query = "https://api.pdf.co/v1/file/upload/get-presigned-url?
contenttype=application/octet-stream&name=" + `
    [System.IO.Path]::GetFileName($SourceFile)
$query = [System.Uri]::EscapeUriString($query)
```

```
try {
    # Execute request
    $jsonResponse = Invoke-RestMethod -Method Get -Headers @{ "x-api-key" = $API_KEY
} -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 @{ "x-api-key" = $API_KEY;
"content-type" = "application/octet-stream" } -InFile $SourceFile -Uri $uploadUrl

        if ($r.StatusCode -eq 200) {

            # 3. CONVERT UPLOADED PDF FILE TO XLS

            # Prepare URL for `PDF To XLS` API call
            $query = "https://api.pdf.co/v1/pdf/convert/to/xls?name={0}&password=
{1}&pages={2}&url={3}" -f `
                $(Split-Path $DestinationFile -Leaf), $Password, $Pages,
$uploadedFileUrl
            $query = [System.Uri]::EscapeUriString($query)

            # Execute request
            $jsonResponse = Invoke-RestMethod -Method Get -Headers @{ "x-api-key" =
$API_KEY } -Uri $query

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

                # Download XLS file
                Invoke-WebRequest -Headers @{ "x-api-key" = $API_KEY } -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
}
```

PowerShell - run.bat

```
@echo off

powershell -NoProfile -ExecutionPolicy Bypass -Command "& .\ConvertPdfToXlsFromUploadedFile.ps1"
echo Script finished with errorlevel=%errorlevel%

pause
```

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

or

Get Your Free API Key for www.PDF.co Web API