How to convert PDF to XML from uploaded file for PDF to XML API in PowerShell and PDF.co Web API

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

Sample source codes below will show you how to cope with a difficult task, for example, PDF to XML API in PowerShell. PDF.co Web API helps with PDF to XML API in PowerShell. 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 PowerShell plus the instruction and the code below will help to learn how to convert PDF to XML from uploaded file. Sample code in PowerShell 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! Enjoy writing a code with ready-to-use sample PowerShell codes to implement PDF to XML API using PDF.co Web API.

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

PowerShell - ConvertPdfToXmlFromUploadedFile.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 XML file name
$DestinationFile = ".\result.xml"


# 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 XML

            # Prepare URL for `PDF To XML` API call
            $query = "https://api.pdf.co/v1/pdf/convert/to/xml?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 XML file
                $resultFileUrl = $jsonResponse.url;

                # Download XML file
                Invoke-WebRequest -Headers @{ "x-api-key" = $API_KEY } -OutFile
$DestinationFile -Uri $resultFileUrl

                Write-Host "Generated XML 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 "&
.\ConvertPdfToXmlFromUploadedFile.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