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

Follow this simple tutorial to learn convert PDF to XML from uploaded file to have PDF to XML API in PowerShell

Every ByteScout tool includes sample PowerShell source codes that you can find here or in the folder with installed ByteScout product. ByteScout Cloud API Server was designed to assist PDF to XML API in PowerShell. ByteScout Cloud API Server is the ready to use Web API Server that can be deployed in less than 30 minutes into your own in-house server or into private cloud server. Can store data on in-house local server based storage or in Amazon AWS S3 bucket. Processing data solely on the server using buil-in ByteScout powered engine, no cloud services are used to process your data!.

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. For implementation of this functionality, please copy and paste the code below into your app using code editor. Then compile and run your app. Further improvement of the code will make it more robust.

ByteScout Cloud API Server - free trial version is available on our website. Also, there are other code samples to help you with your PowerShell application included into trial version.

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:

ConvertPdfToXmlFromUploadedFile.ps1

```powershell
# 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 page
$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
$query = "https://localhost/file/upload/get-presigned-url?contenttype=application/octe
    [System.IO.Path]::GetFileName($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 XML

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

                # Download XML file
                Invoke-WebRequest  -OutFile $DestinationFile -Uri $resultFileUrl
```

```powershell
            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
}
```

run.bat

```bat
@echo off

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

pause
```

VIDEO

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](#)

[www.bytescout.com](#)