www.bytescout.com

PDF to XML API in PowerShell with ByteScout Cloud API Server

PDF to XML API in PowerShell

ByteScout tutorials describe the stuff for programmers who use PowerShell. PDF to XML API in PowerShell can be applied with ByteScout Cloud API Server. ByteScout Cloud API Server is the ready to deploy Web API Server that can be deployed in less than thirty minutes into your own in-house Windows server (no Internet connnection is required to process data!) 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 built-in ByteScout powered engine, no cloud services are used to process your data!.

Want to speed up the application development? Then this PowerShell, code samples for PowerShell, developers help to speed up the application development and writing a code when using ByteScout Cloud API Server. Follow the steps-by-step instructions from the scratch to work and copy and paste code for PowerShell into your editor. This basic programming language sample code for PowerShell will do the whole work for you in implementing PDF to XML API in your app.

ByteScout Cloud API Server free trial version is available for download from our website. Free trial also includes programming tutorials along with source code samples.

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:

ConvertPdfToXmlFromUrlAsynchronously.ps1

```
# Cloud API asynchronous "PDF To XML" job example.
# Allows to avoid timeout errors when processing huge or scanned PDF documents.

# 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.

# Direct URL of source PDF file.
$SourceFileUrl = "https://bytescout-com.s3.amazonaws.com/files/demo-files/cloud-api/pd
# 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 XML file name
$DestinationFile = ".\result.xml"
# (!) Make asynchronous job
$Async = $true

# Sample profile that sets advanced conversion options
# Advanced options are properties of XMLExtractor class from ByteScout XML Extractor S
# https://cdn.bytescout.com/help/BytescoutPDFExtractorSDK/html/6f2b5e9c-ba15-f9fe-192b-
$Profiles = '{ "profiles": [ { "profile1": { "TrimSpaces": "False", "PreserveFormattin


# Prepare URL for `PDF To XML` API call
$query = "https://localhost/pdf/convert/to/xml?name={0}&password={1}&pages={2}&url={3}&
    $(Split-Path $DestinationFile -Leaf), $Password, $Pages, $SourceFileUrl, $Async, $P
$query = [System.Uri]::EscapeUriString($query)

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

    if ($jsonResponse.error -eq $false) {
        # Asynchronous job ID
        $jobId = $jsonResponse.jobId
        # URL of generated XML file that will available after the job completion
        $resultFileUrl = $jsonResponse.url

        # Check the job status in a loop.
        do {
            $statusCheckUrl = "https://localhost/job/check?jobid=" + $jobId
            $jsonStatus = Invoke-RestMethod -Method Get  -Uri $statusCheckUrl

            # Display timestamp and status (for demo purposes)
            Write-Host "$(Get-date): $($jsonStatus.status)"

            if ($jsonStatus.status -eq "success") {
                # Download XML file
                Invoke-WebRequest  -OutFile $DestinationFile -Uri $resultFileUrl
                Write-Host "Generated XML file saved as `"$($DestinationFile)`" file."
                break
            }
            elseif ($jsonStatus.status -eq "working") {
                # Pause for a few seconds
                Start-Sleep -Seconds 3
```

```
            }
            else {
                Write-Host $jsonStatus.status
                break
            }
        }
        while ($true)
    }
    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 "& .\ConvertPdfToXmlFromUrlAsyn
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