www.bytescout.com

How to PDF text search API in PowerShell using ByteScout Cloud API Server

How to write a robust code in PowerShell to PDF text search API with this step-by-step tutorial

We made thousands of pre-made source code pieces for easy implementation in your own programming projects. What is ByteScout Cloud API Server? It 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!. It can help you to PDF text search API in your PowerShell application.

The SDK samples given below describe how to quickly make your application do PDF text search API in PowerShell with the help of ByteScout Cloud API Server. Simply copy and paste in your PowerShell project or application you and then run your app! Check PowerShell sample code samples to see if they respond to your needs and requirements for the project.

If you want to try other source code samples then the free trial version of ByteScout Cloud API Server is available for download from our website. Just try other source code samples for PowerShell.

Source Code Files:

PDFTextSearchFromUploadedFileAsync.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 file name
$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 = ""

# Search string.
$SearchString = '\d{1,}\.\d\d' #Regular expression to find numbers like '100.00'

# Enable regular expressions (Regex)
$RegexSearch = 'True'

# (!) Make asynchronous job
$Async = $true

# 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. TEXT SEARCH FROM UPLOADED FILE

            # Prepare URL for PDF text search API call.
            $query = "https://localhost/pdf/find?password=$($Password)&pages=$($Pages)&
            $query = [System.Uri]::EscapeUriString($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 JSON file with search result that will availabl
                    $resultFileUrl = $jsonResponse.url

                    # Check the job status in a loop.
                    # If you don't want to pause the main thread you can rework the co
                    # to use a separate thread for the status checking and completion.
                    do {
                        $statusCheckUrl = "https://localhost/job/check?jobid=" + $jobI
                        $jsonStatus = Invoke-RestMethod -Method Get  -Uri $statusCheckU

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

                        if ($jsonStatus.status -eq "success") {
                            # Get JSON for search result
                            $jsonSearchResult = Invoke-RestMethod -Method Get  -Uri $re

                            # Display found result in console
                            foreach ($item in $jsonSearchResult)
                            {
                                Write-Host "Found text $($item.text) at coordinates $(
                            }
                            break
                        }
                        elseif ($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
            }

        }
        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 "& .\PDFTextSearchFromUploadedF
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