www.bytescout.com

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

Learn in simple ways: How to convert PDF to HTML from uploaded file for PDF to HTML API in PowerShell

If you want a quick and easy way to add a required functionality into your application then check this sample source code documentation. ByteScout Cloud API Server was designed to assist PDF to HTML 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!.

Want to learn quickly? These fast application programming interfaces of ByteScout Cloud API Server for PowerShell plus the instruction and the code below will help to learn how to convert PDF to HTML from uploaded file. For implementation of this functionality, please copy and paste the code below into your app using code editor. Then compile and run your app. You can use these PowerShell sample examples in one or many applications.

Our website provides free trial version of ByteScout Cloud API Server that gives source code samples to assist with your PowerShell project.

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:

ConvertPdfToHtmlFromUploadedFile.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 PDF file
$SourceFile = ".\sample.pdf"
# 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 HTML file name
$DestinationFile = ".\result.html"
# Set to $true to get simplified HTML without CSS. Default is the rich HTML keeping th
$PlainHtml = $false
# Set to $true if your document has the column layout like a newspaper.
$ColumnLayout = $false


# 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 HTML.

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

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

            if ($jsonResponse.error -eq $false) {
```

```
                # Get URL of generated HTML file
                $resultFileUrl = $jsonResponse.url;

                # Download HTML file
                Invoke-WebRequest  -OutFile $DestinationFile -Uri $resultFileUrl

                Write-Host "Generated HTML 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

```
@echo off

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

pause
```

VIDEO

https://www.youtube.com/watch?v=NEwNs2b9YN8