How to convert images to PDF from uploaded files for image to PDF API in PowerShell with PDF.co Web API

How to convert images to PDF from uploaded files for image to PDF API in PowerShell: How To Tutorial

We've created and updating regularly our sample code library so you may quickly learn image to PDF API and the step-by-step process in PowerShell. Image to PDF API in PowerShell can be implemented with PDF.co Web API. PDF.co Web API is the Web API with a set of tools for documents manipulation, data conversion, data extraction, splitting and merging of documents. Includes image recognition, built-in OCR, barcode generation and barcode decoders to decode bar codes from scans, pictures and pdf.

PowerShell code samples for PowerShell developers help to speed up the application's code writing when using PDF.co Web API. This PowerShell sample code should be copied and pasted into your project. After doing this just compile your project and click Run. Writing PowerShell application typically includes multiple stages of the software development so even if the functionality works please test it with your data and the production environment.

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

PowerShell - ConvertImagesToPdfFromUploadedFiles.ps1

```powershell
# The authentication key (API Key).
# Get your own by registering at https://app.pdf.co/documentation/api
$API_KEY = "***********************************"

# Source image files
$ImageFiles = ".\image1.png", ".\image2.jpg"
# Destination PDF file name
$DestinationFile = ".\result.pdf"

$uploadedFiles = @()

try {
    foreach ($imageFile in $ImageFiles ) {
        # 1a. RETRIEVE THE PRESIGNED URL TO UPLOAD THE FILE.

        # Prepare URL for `Get Presigned URL` API call
        $query = "https://api.pdf.co/v1/file/upload/get-presigned-url?
contenttype=application/octet-stream&name=" + `
            [IO.Path]::GetFileName($imageFile)
        $query = [System.Uri]::EscapeUriString($query)

        # Execute request
```

```powershell
        $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

            # 1b. UPLOAD THE FILE TO CLOUD.

            $r = Invoke-WebRequest -Method Put -Headers @{ "x-api-key" = $API_KEY;
"content-type" = "application/octet-stream" } -InFile $imageFile -Uri $uploadUrl

            if ($r.StatusCode -eq 200) {
                # Keep uploaded file URL
                $uploadedFiles += $uploadedFileUrl
            }
            else {
                # Display request error status
                Write-Host $r.StatusCode + " " + $r.StatusDescription
            }
        }
        else {
            # Display service reported error
            Write-Host $jsonResponse.message
        }
    }

    if ($uploadedFiles.length -gt 0) {
        # 2. CREATE PDF DOCUMENT FROM UPLOADED IMAGE FILES

        # Prepare URL for `DOC To PDF` API call
        $query = "https://api.pdf.co/v1/pdf/convert/from/image?name=$(Split-Path
$DestinationFile -Leaf)&url=$($uploadedFiles -join ",")"
        $query = [System.Uri]::EscapeUriString($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 PDF file
            $resultFileUrl = $jsonResponse.url;

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

            Write-Host "Generated PDF file saved as `"$($DestinationFile)`" file."
        }
        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 "&
.\ConvertImagesToPdfFromUploadedFiles.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](#)