How to convert PDF to XLSX from URL for PDF to excel API in PowerShell using PDF.co Web API

Learn how to convert PDF to XLSX from URL to have PDF to excel API in PowerShell

The documentation is designed to help you to implement the features on your side. PDF to excel API in PowerShell can be implemented with PDF.co Web API. PDF.co Web API is the flexible Web API that includes full set of functions from e-signature requests to data extraction, OCR, images recognition, pdf splitting and pdf splitting. Can also generate barcodes and read barcodes from images, scans and pdf.

You will save a lot of time on writing and testing code as you may just take the code below and use it in your application. For implimentation of this functionality, please copy and paste code below into your app using code editor. Then compile and run your app. This basic programming language sample code for PowerShell will do the whole work for you in implementing PDF to excel API in your app.

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

PowerShell - ConvertPdfToXlsxFromUrl.ps1

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

# Direct URL of source PDF file.
$SourceFileUrl = "https://bytescout-com.s3.amazonaws.com/files/demo-files/cloud-
api/pdf-to-excel/sample.pdf"
# Comma-separated list of page indices (or ranges) to process. Leave empty for all
pages. Example: '0,2-5,7-'.
$Pages = ""
# PDF document password. Leave empty for unprotected documents.
$Password = ""
# Destination XLSX file name
$DestinationFile = ".\result.xlsx"


# Prepare URL for `PDF To XLSX` API call
$query = "https://api.pdf.co/v1/pdf/convert/to/xlsx?name={0}&password={1}&pages=
{2}&url={3}" -f `
    $(Split-Path $DestinationFile -Leaf), $Password, $Pages, $SourceFileUrl
$query = [System.Uri]::EscapeUriString($query)

try {
    # Execute request
    $jsonResponse = Invoke-RestMethod -Method Get -Headers @{ "x-api-key" = $API_KEY
```

```
} -Uri $query

    if ($jsonResponse.error -eq $false) {
        # Get URL of generated XLSX file
        $resultFileUrl = $jsonResponse.url;

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

        Write-Host "Generated XLSX 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 "&
.\ConvertPdfToXlsxFromUrl.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