

How to convert PDF to HTML from uploaded file for PDF to HTML API in PowerShell with PDF.co Web API

See how to convert PDF to HTML from uploaded file to have PDF to HTML API in PowerShell

Today we will explain the steps and algorithm of how to convert PDF to HTML from uploaded file and how to make it work in your application. PDF.co Web API helps with PDF to HTML API in PowerShell. PDF.co Web API is the Rest API that provides set of data extraction functions, tools for documents manipulation, splitting and merging of pdf files. Includes built-in OCR, images recognition, can generate and read barcodes from images, scans and pdf.

The SDK samples like this one below explain how to quickly make your application do PDF to HTML API in PowerShell with the help of PDF.co Web API. Open your PowerShell project and simply copy & paste the code and then run your app! Tutorials are available along with installed PDF.co Web API if you'd like to dive deeper into the topic and the details of the API.

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

PowerShell - ConvertPdfToHtmlFromUploadedFile.ps1

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

# Source PDF file
$SourceFile = ".\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 HTML file name
$DestinationFile = ".\result.html"
# Set to $true to get simplified HTML without CSS. Default is the rich HTML keeping
the document design.
$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
$query = "https://api.pdf.co/v1/file/upload/get-presigned-url?
contenttype=application/octet-stream&name=" + `
[System.IO.Path]::GetFileName($SourceFile)
$query = [System.Uri]::EscapeUriString($query)

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

    if ($jsonResponse.error -eq $false) {
        # Get URL 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 @{ "x-api-key" = $API_KEY;
"content-type" = "application/octet-stream" } -InFile $SourceFile -Uri $uploadUrl

        if ($r.StatusCode -eq 200) {

            # 3. CONVERT UPLOADED PDF FILE TO HTML.

            # Prepare URL for `PDF To HTML` API call
            $query = "https://api.pdf.co/v1/pdf/convert/to/html?name={0}&password=
{1}&pages={2}&simple={3}&columns={4}&url={5}" -f `
                $(Split-Path $DestinationFile -Leaf), $Password, $Pages, $PlainHtml,
                $ColumnLayout, $uploadedFileUrl
            $query = [System.Uri]::EscapeUriString($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 HTML file
                $resultFileUrl = $jsonResponse.url;

                # Download HTML file
                Invoke-WebRequest -Headers @{ "x-api-key" = $API_KEY } -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
}
```

PowerShell - run.bat

```
@echo off

powershell -NoProfile -ExecutionPolicy Bypass -Command "&
.\ConvertPdfToHtmlFromUploadedFile.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](http://www.ByteScout.com)

or

[Get Your Free API Key for www.PDF.co Web API](#)