

# How to convert PDF to CSV from URL for PDF to CSV API in PowerShell and PDF.co Web API

How to convert PDF to CSV from URL in PowerShell with easy ByteScout code samples to make PDF to CSV API. Step-by-step tutorial

Sample source codes below will show you how to cope with a difficult task, for example, PDF to CSV API in PowerShell. PDF.co Web API helps with PDF to CSV API in PowerShell. 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 snippet like this for PDF.co Web API works best when you need to quickly implement PDF to CSV API in your PowerShell application. For implementation of this functionality, please copy and paste code below into your app using code editor. Then compile and run your app. Code testing will allow the function to be tested and work properly with your data.

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

PowerShell - ConvertPdfToCsvFromUrl.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-csv/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 CSV file name
$DestinationFile = ".\result.csv"

# Prepare URL for `PDF To CSV` API call
$query = "https://api.pdf.co/v1/pdf/convert/to/csv?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 CSV file
    $resultFileUrl = $jsonResponse.url;

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

    Write-Host "Generated CSV 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 "&
.\ConvertPdfToCsvFromUrl.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](#)