

How to convert PDF to JPEG from URL asynchronously for PDF to image API in PowerShell and ByteScout Cloud API Server

Step By Step Instructions on how to convert PDF to JPEG from URL asynchronously for PDF to image API in PowerShell

We regularly create and update our sample code library so you may quickly learn PDF to image API and the step-by-step process in PowerShell. PDF to image API in PowerShell can be applied with ByteScout Cloud API Server. ByteScout Cloud API Server is API server that is ready to use and can be installed and deployed in less than 30 minutes on your own Windows server or server in a cloud. It can save data and files on your local server-based file storage or in Amazon AWS S3 storage. Data is processed solely on the API server and is powered by ByteScout engine, no cloud services or Internet connection is required for data processing..

PowerShell code snippet like this for ByteScout Cloud API Server works best when you need to quickly implement PDF to image API in your PowerShell application. This sample code in PowerShell is all you need. Just copy-paste it to the code editor, then add a reference to ByteScout Cloud API Server and you are ready to try it! Further improvement of the code will make it more robust.

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](http://www.Bytescout.com)

Source Code Files:

ConvertPdfToJpegFromUrlAsynchronously.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.

# Cloud API asynchronous "PDF To JPEG" job example.
# Allows to avoid timeout errors when processing huge or scanned PDF documents.

# Direct URL of source PDF file.
$SourceFileUrl = "https://bytescout-com.s3.amazonaws.com/files/demo-files/cloud-api/pdf
# Comma-separated list of page indices (or ranges) to process. Leave empty for all page
$Pages = ""
# PDF document password. Leave empty for unprotected documents.
$Password = ""
# (!) Make asynchronous job
$Async = $true

# Prepare URL for `PDF To JPEG` API call
$query = "https://localhost/pdf/convert/to/jpg?password={0}&pages={1}&url={2}&async={3}
    $Password, $Pages, $SourceFileUrl, $Async
$query = [System.Uri]::EscapeUriString($query)

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

    if ($jsonResponse.error -eq $false) {
        # Asynchronous job ID
        $jobId = $jsonResponse.jobId
        # URL of generated JSON file available after the job completion; it will conta
        $resultJsonFileUrl = $jsonResponse.url

        # Check the job status in a loop.
        do {
            $statusCheckUrl = "https://localhost/job/check?jobid=" + $jobId
            $jsonStatus = Invoke-RestMethod -Method Get -Uri $statusCheckUrl

            # Display timestamp and status (for demo purposes)
            Write-Host "$(Get-date): $($jsonStatus.status)"

            if ($jsonStatus.status -eq "success") {
                # Download JSON file with URLs of result PDF files
                $jsonJpegUrls = Invoke-RestMethod -Method Get -Uri $resultJsonFileUrl

                # Download generated JPEG files
                $part = 1;
                foreach ($url in $jsonJpegUrls) {
                    $localFileName = ".\page${$part}.jpg"

                    Invoke-WebRequest -OutFile $localFileName -Uri $url

                    Write-Host "Downloaded `"$($localFileName)`""
                    $part++
                }
            }
        } while ($jsonStatus.status -ne "success")
    }
}
```

```
        }
        break
    }
    elseif ($jsonStatus.status -eq "working") {
        # Pause for a few seconds
        Start-Sleep -Seconds 3
    }
    else {
        Write-Host $jsonStatus.status
        break
    }
}
while ($true)
}
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 "& .\ConvertPdfToJpegFromUrlAsync.ps1"
echo Script finished with errorlevel=%errorlevel%
pause
```

VIDEO

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

ON-PREMISE OFFLINE SDK

[60 Day Free Trial](#) or [Visit ByteScout Cloud API Server Home Page](#)
[Explore ByteScout Cloud API Server Documentation](#)
[Explore Samples](#)
[Sign Up for ByteScout Cloud API Server Online Training](#)

ON-DEMAND REST WEB API

[Get Your API Key](#)
[Explore Web API Docs](#)
[Explore Web API Samples](#)

[visit www.ByteScout.com](http://www.ByteScout.com)

[visit www.PDF.co](http://www.PDF.co)

www.bytescout.com