

How to generate PDF from large HTML file for HTML to PDF API in PowerShell using 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.

FOR MORE INFORMATION AND FREE TRIAL:

[Download Free Trial SDK \(on-premise version\)](#)

[Read more about PDF.co Web API](#)

[Explore API Documentation](#)

[Get Free Training for PDF.co Web API](#)

[Get Free API key for Web API](#)

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

Source Code Files:

GeneratePdfFromHtml.ps1

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

# HTML input
$htmlInput = [IO.File]::ReadAllText(".\sample.html")
# Destination PDF file name
$destinationFile = ".\result.pdf"

# Prepare URL for HTML to PDF API call
$query = "https://api.pdf.co/v1/pdf/convert/from/html"
$query = [System.Uri]::EscapeUriString($query)

# Prepare request body (will be auto-converted to JSON by Invoke-RestMethod)
# See documentation: https://apidocs.pdf.co
$body = @{
```

```

"html" = $HtmlInput
"name" = $(Split-Path $DestinationFile -Leaf)
"async" = $true # Make asynchronous job
} | ConvertTo-Json

try {
    # Execute request
    $response = Invoke-WebRequest -Method Post -Headers @{ "x-api-key" = $API_KEY; "Content-Type" = "application/json" } -Uri $url -Body $json

    if ($response.StatusCode -eq 200) {

        $jsonResponse = $response.Content | ConvertFrom-Json

        if ($jsonResponse.error -eq $false) {
            # Asynchronous job ID
            $jobId = $jsonResponse.jobId
            # URL of generated PDF file that will be available after the job completion
            $resultFileUrl = $jsonResponse.url

            # Check the job status in a loop.
            do {
                $statusCheckUrl = "https://api.pdf.co/v1/job/check?jobid=" + $jobId
                $jsonStatus = Invoke-RestMethod -Method Get -Headers @{ "x-api-key" = $API_KEY } -Uri $statusCheckUrl

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

                if ($jsonStatus.status -eq "success") {
                    # Download PDF file
                    Invoke-WebRequest -Headers @{ "x-api-key" = $API_KEY } -OutFile $DestinationFile -Uri $resultFileUrl
                    Write-Host "Generated PDF file saved as `"$($DestinationFile)`" file."
                    break
                }
                elseif ($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
        }
    }
    else {
        # Display request error status
        Write-Host $response.StatusCode + " " + $response.StatusDescription
    }
}
catch {
    # Display request error
    Write-Host $_.Exception
}

```

VIDEO

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

ON-PREMISE OFFLINE SDK

[60 Day Free Trial](#) or [Visit PDF.co Web API Home Page](#)
[Explore PDF.co Web API Documentation](#)
[Explore Samples](#)
[Sign Up for PDF.co Web API 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