How to optimize PDF from URL for PDF optimization API in PowerShell and ByteScout Cloud API Server

How to optimize PDF from URL for PDF optimization API in PowerShell: Step By Step Instructions

These source code samples are listed and grouped by their programming language and functions they use. ByteScout Cloud API Server was designed to assist PDF optimization API in PowerShell. ByteScout Cloud API Server is the ready to use Web API Server that can be deployed in less than 30 minutes into your own in-house server or into private cloud server. Can store data on in-house local server based storage or in Amazon AWS S3 bucket. Processing data solely on the server using buil-in ByteScout powered engine, no cloud services are used to process your data!.

The SDK samples displayed below below explain how to quickly make your application do PDF optimization API in PowerShell with the help of ByteScout Cloud API Server. This PowerShell sample code can be used by copying and pasting into your project. Once done,just compile your project and click Run. Check PowerShell sample code examples to see if they respond to your needs and requirements for the project.

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

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

Source Code Files:

OptimizePdfFromUrl.ps1

```powershell
# 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.

# Direct URL of source PDF file.
$SourceFileURL = "https://bytescout-com.s3.amazonaws.com/files/demo-files/cloud-api/pd
# PDF document password. Leave empty for unprotected documents.
$Password = ""
# Destination PDF file name
$DestinationFile = ".\result.pdf"

# Prepare URL for `Optimize PDF` API call
$query = "https://localhost/pdf/optimize?name={0}&password={1}&url={2}" -f `
    $(Split-Path $DestinationFile -Leaf), $Password, $SourceFileURL
$query = [System.Uri]::EscapeUriString($query)

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

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

        # Download PDF file
        Invoke-WebRequest  -OutFile $DestinationFile -Uri $resultFileUrl

        Write-Host "Generated PDF file saved as `"$($DestinationFile)`" file."
    }
    else {
        # Display service reported error
        Write-Host $jsonResponse.message
    }
}
catch {
    # Display request error
    Write-Host $_.Exception
}
```

run.bat

```bat
@echo off

powershell -NoProfile -ExecutionPolicy Bypass -Command "& .\OptimizePdfFromUrl.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

visit www.PDF.co