www.bytescout.com

How to PDF make searchable API in Python using ByteScout Cloud API Server

Learn to code in Python to PDF make searchable API with this step-by-step tutorial

The coding instructions are formulated to help you to try-out the features without the requirement to write your own code. Want to PDF make searchable API in your Python app? ByteScout Cloud API Server is designed for it. ByteScout Cloud API Server is the ready to deploy Web API Server that can be deployed in less than thirty minutes into your own in-house Windows server (no Internet connnection is required to process data!) 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 built-in ByteScout powered engine, no cloud services are used to process your data!.

The following code snippet for ByteScout Cloud API Server works best when you need to quickly PDF make searchable API in your Python application. Just copy and paste the code into your Python application's code and follow the instructions. If you want to use these Python sample examples in one or many applications then they can be used easily.

If you want to try other source code samples then the free trial version of ByteScout Cloud API Server is available for download from our website. Just try other source code samples for Python.

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:

MakeSearchablePdfFromUrlAsynchronously.py

```
""" Cloud API asynchronous "PDF To Text" job example.
    Allows to avoid timeout errors when processing huge or scanned PDF documents.
"""
import os
import requests # pip install requests
import time
import datetime

# Please NOTE: In this sample we're assuming Cloud Api Server is hosted at "https://localhost".
# If it's not then please replace this with with your hosting url.

# Base URL for PDF.co Web API requests
BASE_URL = "https://localhost"

# Direct URL of source PDF file.
SourceFileURL = "https://bytescout-com.s3.amazonaws.com/files/demo-files/cloud-api/pdf-make-searchable/sample.p
# 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 PDF file name
DestinationFile = ".\\result.pdf"
# OCR language. "eng", "fra", "deu", "spa"  supported currently. Let us know if you need more.
Language = "eng"
# (!) Make asynchronous job
Async = True


def main(args = None):
    makeSearchablePDF(SourceFileURL, DestinationFile)


def makeSearchablePDF(uploadedFileUrl, destinationFile):
    """Make Searchable PDF using PDF.co Web API"""

    # Prepare URL for 'Make Searchable PDF' API request
    url = "{}/pdf/makesearchable?async={}&name={}&password={}&pages={}&lang={}&url={}".format(
        BASE_URL,
        Async,
        os.path.basename(destinationFile),
        Password,
        Pages,
        Language,
        uploadedFileUrl
    )

    # Execute request and get response as JSON
    response = requests.get(url, headers={  "content-type": "application/octet-stream" })
    if (response.status_code == 200):
        json = response.json()

        if json["error"] == False:
            # Asynchronous job ID
            jobId = json["jobId"]
            #  URL of the result file
            resultFileUrl = json["url"]

            # Check the job status in a loop.
            # If you don't want to pause the main thread you can rework the code
            # to use a separate thread for the status checking and completion.
            while True:
                status = checkJobStatus(jobId) # Possible statuses: "working", "failed", "aborted", "success".

                # Display timestamp and status (for demo purposes)
                print(datetime.datetime.now().strftime("%H:%M.%S") + ": " + status)

                if status == "success":
                    # Download result file
                    r = requests.get(resultFileUrl, stream=True)
                    if (r.status_code == 200):
```

```
                    with open(destinationFile, 'wb') as file:
                        for chunk in r:
                            file.write(chunk)
                    print(f"Result file saved as \"{destinationFile}\" file.")
                else:
                    print(f"Request error: {response.status_code} {response.reason}")
                break
            elif status == "working":
                # Pause for a few seconds
                time.sleep(3)
            else:
                print(status)
                break
        else:
            # Show service reported error
            print(json["message"])
    else:
        print(f"Request error: {response.status_code} {response.reason}")


def checkJobStatus(jobId):
    """Checks server job status"""

    url = f"{BASE_URL}/job/check?jobid={jobId}"

    response = requests.get(url)
    if (response.status_code == 200):
        json = response.json()
        return json["status"]
    else:
        print(f"Request error: {response.status_code} {response.reason}")

    return None


if __name__ == '__main__':
    main()
```

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