

How to convert PDF to excel from uploaded file asynchronously for PDF to excel API in Python and PDF.co Web API

Learn in simple ways: How to convert PDF to excel from uploaded file asynchronously for PDF to excel API in Python

On this page, you will find sample source codes which show you how to handle a complex task, such as, PDF to excel API in Python. PDF.co Web API helps with PDF to excel API in Python. PDF.co Web API is the Rest API that provides set of data extraction functions, tools for documents manipulation, splitting and merging of pdf files. Includes built-in OCR, images recognition, can generate and read barcodes from images, scans and pdf.

This simple and easy to understand sample source code in Python for PDF.co Web API contains different functions and options you should do calling the API to implement PDF to excel API. Open your Python project and simply copy & paste the code and then run your app! This basic programming language sample code for Python will do the whole work for you in implementing PDF to excel API in your app.

PDF.co Web API - free trial version is available on our website. Also, there are other code samples to help you with your Python application included into trial version.

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:

ConvertPdfToExcelFromUploadedFileAsynchronously.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

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

# Base URL for PDF.co Web API requests
BASE_URL = "https://api.pdf.co/v1"

# Source PDF file
SourceFile = ".\\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 Excel file name
DestinationFile = ".\\result.xlsx"
# (!) Make asynchronous job
Async = True

def main(args = None):
    uploadedFileUrl = uploadFile(SourceFile)
    if (uploadedFileUrl != None):
        convertPdfToExcel(uploadedFileUrl, DestinationFile)

def convertPdfToExcel(uploadedFileUrl, destinationFile):
    """Converts PDF To Excel using PDF.co Web API"""

    # Prepare URL for 'PDF To Xlsx' API request
    url = "{}/pdf/convert/to/xlsx?async={}&name={}&password={}&pages={}&url={}".format(
        BASE_URL,
        Async,
        os.path.basename(destinationFile),
        Password,
        Pages,
        uploadedFileUrl
    )

    # Execute request and get response as JSON
    response = requests.get(url, headers={ "x-api-key": API_KEY, "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, headers={"x-api-key": API_KEY })
    if (response.status_code == 200):
        json = response.json()
        return json["status"]
    else:
        print(f"Request error: {response.status_code} {response.reason}")

    return None

def uploadFile(fileName):
    """Uploads file to the cloud"""

    # 1. RETRIEVE PRESIGNED URL TO UPLOAD FILE.

    # Prepare URL for 'Get Presigned URL' API request
    url = "{}/file/upload/get-presigned-url?contenttype=application/octet-stream&name={}".format(
        BASE_URL, os.path.basename(fileName))

    # Execute request and get response as JSON
    response = requests.get(url, headers={"x-api-key": API_KEY })
    if (response.status_code == 200):
        json = response.json()

        if json["error"] == False:
            # URL to use for file upload
            uploadUrl = json["presignedUrl"]
            # URL for future reference
            uploadedFileUrl = json["url"]

            # 2. UPLOAD FILE TO CLOUD.
            with open(fileName, 'rb') as file:
                requests.put(uploadUrl, data=file, headers={"x-api-key": API_KEY, "content-type": "application/octet-stream"})

            return uploadedFileUrl
        else:
            # Show service reported error
            print(json["message"])
    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 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