```
import os
import pdfplumber
import pandas as pd
import gspread
from oauth2client.service_account import ServiceAccountCredentials

def extract_data_from_pdf(pdf_path):
    """
       Extracts data from a Pdf and converts in into a Dataframe.
       Adjust the logic here depending on the structure of your PDf.
    """
    with pdfplumber.open(pdf_path) as pdf:
        text = ""
        for page in pdf.pages:
            text += page.extract_text() + '\n'

    lines = text.splitlines()
    data = [line.split() for line in lines if line.strip()]
    df = pd.DataFrame(data)
    return df

def upload_pdfs_to_google_sheets(folder_path,sheet_id,creds_path):
    """
    Extracts data from all pdfs in a folder and uploads each to a
separate sheet in Google Sheet
    """
    # Autheticate with google sheets API

    scope = ["https://spreadsheets.google.com/feeds",
"https://www.googleapis.com/auth/drive"]
    creds =
ServiceAccountCredentials.from_json_keyfile_name(creds_path,scope)
    client = gspread.authorize(creds)

    # Open the Google Sheet
    sheet = client.open_by_key(sheet_id)

    # Iterate through all pdf files in the folder
    for file_name in os.listdir(folder_path):
        if file_name.endswith(".pdf"):
            pdf_path = os.path.join(folder_path,file_name)
            sheet_name = os.path.splitext(file_name)[0]

            # Extract data from the pdf
            df = extract_data_from_pdf(pdf_path)

            # create or open a worksheet the google Sheets
            try:
                worksheet =
sheet.add_worksheet(title=sheet_name,rows="1000",cols='20')
```

```
            except gpspread.exceptions.APIError:
                worksheet = sheet.worksheet(sheet_name)

             # Update the worksheet with the extracted data
            worksheet.clear()
            worksheet.update([df.columns.tolist()]+df.values.tolist())
            print(f"Uploaded {file_name} to sheet {sheet_name}")

if __name__ == "__main__":
    folder_path = r"C:\NISHANT\Skill Academy\Statistics\Probability
Distributions & Central Limit Theorem\Probability Distributions &
Central Limit Theorem"
    sheet_id = "1mUE0g_JYRWRhRE6GKtqRo_chW5d4_0UOM1Fg7yYrSzA"
    creds_path = r"C:\Users\Nishant shah\OneDrive\Desktop\Numpy\
subtle-reserve-441716-r6-6e49d5b4ced4.json"

    upload_pdfs_to_google_sheets(folder_path,sheet_id,creds_path)
```

```
----------------------------------------------------------------------
-----
APIError                                  Traceback (most recent call
last)
File c:\Users\Nishant shah\OneDrive\Desktop\Numpy\myenv\Lib\site-
packages\gspread\client.py:155, in Client.open_by_key(self, key)
    154 try:
--> 155     spreadsheet = Spreadsheet(self.http_client, {"id": key})
    156 except APIError as ex:

File c:\Users\Nishant shah\OneDrive\Desktop\Numpy\myenv\Lib\site-
packages\gspread\spreadsheet.py:29, in Spreadsheet.__init__(self,
http_client, properties)
     27 self._properties = properties
---> 29 metadata = self.fetch_sheet_metadata()
     30 self._properties.update(metadata["properties"])

File c:\Users\Nishant shah\OneDrive\Desktop\Numpy\myenv\Lib\site-
packages\gspread\spreadsheet.py:230, in
Spreadsheet.fetch_sheet_metadata(self, params)
    221 """Similar to :method
spreadsheets_get:`gspread.http_client.spreadsheets_get`,
    222 get the spreadsheet form the API but by default **does not get
the cells data**.
    223 It only retrieve the the metadata from the spreadsheet.
   (...)
    228 :rtype: dict
    229 """
--> 230 return self.client.fetch_sheet_metadata(self.id,
params=params)
```

```
File c:\Users\Nishant shah\OneDrive\Desktop\Numpy\myenv\Lib\site-
packages\gspread\http_client.py:305, in
HTTPClient.fetch_sheet_metadata(self, id, params)
    303 url = SPREADSHEET_URL % id
--> 305 r = self.request("get", url, params=params)
    307 return r.json()

File c:\Users\Nishant shah\OneDrive\Desktop\Numpy\myenv\Lib\site-
packages\gspread\http_client.py:128, in HTTPClient.request(self,
method, endpoint, params, data, json, files, headers)
    127 else:
--> 128     raise APIError(response)

APIError: APIError: [404]: Requested entity was not found.

The above exception was the direct cause of the following exception:

SpreadsheetNotFound                       Traceback (most recent call
last)
Cell In[17], line 54
     51 sheet_id = "1aRAQNiiRLnqK-Qvrb4SodgG3lkh7IhzbFPtOJv8VQxU"
     52 creds_path = r"C:\Users\Nishant shah\OneDrive\Desktop\Numpy\
subtle-reserve-441716-r6-6e49d5b4ced4.json"
---> 54 upload_pdfs_to_google_sheets(folder_path,sheet_id,creds_path)

Cell In[17], line 27, in upload_pdfs_to_google_sheets(folder_path,
sheet_id, creds_path)
     24 client = gspread.authorize(creds)
     26 # Open the Google Sheet
---> 27 sheet = client.open_by_key(sheet_id)
     29 # Iterate through all pdf files in the folder
     30 for file_name in os.listdir(folder_path):

File c:\Users\Nishant shah\OneDrive\Desktop\Numpy\myenv\Lib\site-
packages\gspread\client.py:158, in Client.open_by_key(self, key)
    156 except APIError as ex:
    157     if ex.response.status_code == HTTPStatus.NOT_FOUND:
--> 158         raise SpreadsheetNotFound(ex.response) from ex
    159     if ex.response.status_code == HTTPStatus.FORBIDDEN:
    160         raise PermissionError from ex

SpreadsheetNotFound: <Response [404]>
```