Below is a minor example of parsing through PDFs. This concept, paired with the above examples, was used to create my automated reporting script.

This script uses the pdfminer.six library to convert a pdf into text. The text is then parsed using the re library, and split() function.

import re

import pdfminer

from pdfminer.high\_level import extract\_text

# Function to convert PDF to text

def pdf\_to\_text(path):

    text = extract\_text(path)

    # Write the extracted text to a file

    with open('doc2.txt', 'w') as f:

        f.write(text)

    with open('doc2.txt','r') as f:

        doc = f.read()

    # Parse out the data

    report = doc.split("Readings:")

    cleandata = report[1].replace('\n\n', ',')

    cleandata = cleandata.split("Area")[:1]

    cleandata = cleandata[0]

    # Extract numbers

    numbers\_only = re.findall(r'\d+', cleandata)

    # Conversion

    numbers\_integers = [int(num) for num in numbers\_only]

    numbers\_string = ','.join(numbers\_only)

    # Avg airflow

    avg = sum(numbers\_integers) / len(numbers\_integers)

    print(avg)

    print(' ')

    print(numbers\_integers)

    print(' ')

    # Parse out the report date

    cleandata = report[1].replace('\n', '')

    cleandata = cleandata.split("Report Date:")[1]

    cleandata = cleandata.split("Retest Date:")[:1]

    repdate = cleandata[0]

    print(repdate)

    # Parse out the next date

    cleandata = report[1].replace('\n', '')

    cleandata = cleandata.split("Retest Date:")[1]

    cleandata = cleandata.split("eData:")[:1]

    retdate = cleandata[0]

    print(retdate)

    # Get sap

    report = doc.split("SN:")

    cleandata = report[1].replace('\n','')

    cleandata = cleandata.split('Class')

    sap = '30000000' + cleandata[0]

    sap = sap.replace(' ','')

    print(sap)

    # Get location

    report = doc.split("Chesterfield, MO 63017")[1]

    cleandata = report[3:10].replace('\n','')

    loc = cleandata

    print(loc)

pdf\_to\_text('/content/CV 2024-3 300000000261228.pdf')

Import libraries (pip install required)

Writes pdf to txt file then references it.

The split() function is used to split a string into a list of substrings.

String Parsing: split() scans through the string from left to right, looking for the delimiter. Then cuts from there.

Script Output. Data was pulled from PDF file:

