Rock,Paper,Scissors:

import random

while True:

user\_action = input("Enter a choice (rock, paper, scissors): ")

possible\_actions = ["rock", "paper", "scissors"]

computer\_action = random.choice(possible\_actions)

print(f"\nYou chose {user\_action}, computer chose {computer\_action}.\n")

if user\_action == computer\_action:

print(f"Both players selected {user\_action}. It's a tie!")

elif user\_action == "rock":

if computer\_action == "scissors":

print("Rock smashes scissors! You win!")

else:

print("Paper covers rock! You lose.")

elif user\_action == "paper":

if computer\_action == "rock":

print("Paper covers rock! You win!")

else:

print("Scissors cuts paper! You lose.")

elif user\_action == "scissors":

if computer\_action == "paper":

print("Scissors cuts paper! You win!")

else:

print("Rock smashes scissors! You lose.")

Output:

Enter a choice (rock, paper, scissors): rock

You chose rock, computer chose paper.

Paper covers rock! You lose.

Enter a choice (rock, paper, scissors):

Random password generator:

import random

import array

MAX\_LEN = int(input("enter the length of the password:"))

DIGITS = ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']

LOCASE\_CHARACTERS = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h',

'i', 'j', 'k', 'm', 'n', 'o', 'p', 'q',

'r', 's', 't', 'u', 'v', 'w', 'x', 'y',

'z']

UPCASE\_CHARACTERS = ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H',

'I', 'J', 'K', 'M', 'N', 'O', 'P', 'Q',

'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y',

'Z']

SYMBOLS = ['@', '#', '$', '%', '=', ':', '?', '.', '/', '|', '~', '>',

'\*', '(', ')', '<']

COMBINED\_LIST = DIGITS + UPCASE\_CHARACTERS + LOCASE\_CHARACTERS + SYMBOLS

rand\_digit = random.choice(DIGITS)

rand\_upper = random.choice(UPCASE\_CHARACTERS)

rand\_lower = random.choice(LOCASE\_CHARACTERS)

rand\_symbol = random.choice(SYMBOLS)

temp\_pass = rand\_digit + rand\_upper + rand\_lower + rand\_symbol

for x in range(MAX\_LEN - 4):

temp\_pass = temp\_pass + random.choice(COMBINED\_LIST)

temp\_pass\_list = array.array('u', temp\_pass)

random.shuffle(temp\_pass\_list)

password = ""

for x in temp\_pass\_list:

password = password + x

print(password)

Output:

enter the length of the password:11

.>:m8Faq8Ob

Guess the number:

import random

num = random.randint(1, 10)

guess = None

while guess != num:

guess = input("guess a number between 1 and 10: ")

guess = int(guess)

if guess == num:

print("congratulations! you won!")

break

else:

print("nope, sorry. try again!")

Output:

guess a number between 1 and 10: 5

congratulations! you won!

Or

guess a number between 1 and 10: 3

nope, sorry. try again!

guess a number between 1 and 10:

Email slicer:

email = input("Enter Your Email: ").strip()

username = email[:email.index("@")]

domain\_name = email[email.index("@")+1:]

slicer = (f"Your user name is '{username}' and your domain is '{domain\_name}'")

print(slicer)

output:

Enter Your Email: sathyabama@gmail.com

Your user name is 'sathyabama' and your domain is 'gmail.com'

Check the validity of password:

l, u, p, d = 0, 0, 0, 0

s = input("Enter your password:")

if (len(s) >= 8):

for i in s:

if (i.islower()):

l+=1

if (i.isupper()):

u+=1

if (i.isdigit()):

d+=1

if(i=='@'or i=='$' or i=='\_'):

p+=1

if (l>=0 and u>=0 and p>=0 and d>=0 and l+p+u+d==len(s)):

print("Valid Password")

else:

print("Invalid Password")

Output:

Enter your password:qPnsfkng@2

Valid Password

Or

Enter your password:gsld

Invalid Password

Writing CSV files using python:

import csv

fields = ['Name', 'Branch', 'Year', 'CGPA']

rows = [ ['Nikhil', 'COE', '2', '9.0'],

['Sanchit', 'COE', '2', '9.1'],

['Aditya', 'IT', '2', '9.3'],

['Sagar', 'SE', '1', '9.5'],

['Prateek', 'MCE', '3', '7.8'],

['Sahil', 'EP', '2', '9.1']]

filename = "university\_records.csv"

with open(filename, 'w') as csvfile:

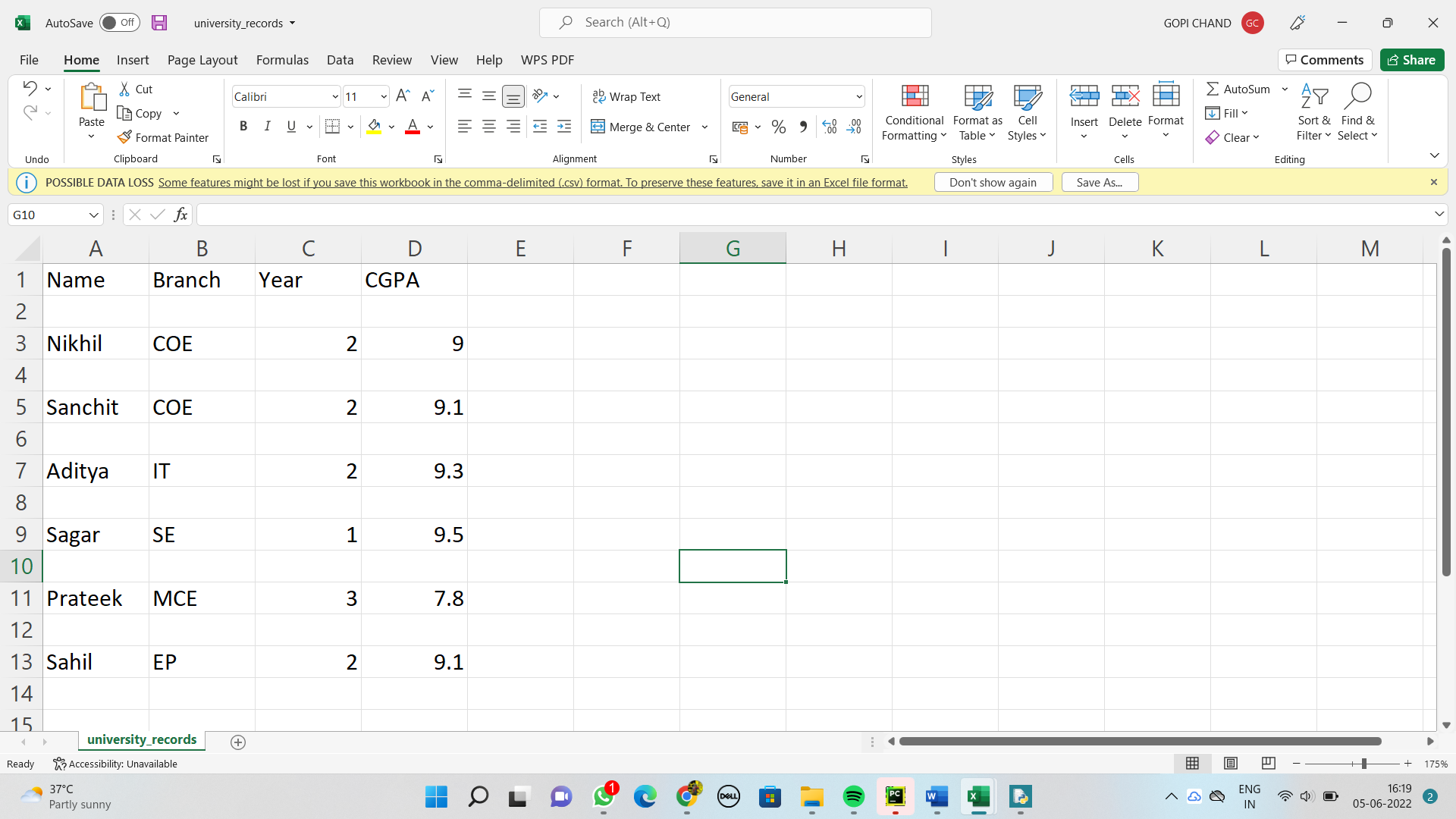
csvwriter = csv.writer(csvfile)

csvwriter.writerow(fields)

csvwriter.writerows(rows)

Output:





Simple calculator:

def add(x, y):

return x + y

def subtract(x, y):

return x - y

def multiply(x, y):

return x \* y

def divide(x, y):

return x / y

print("Select operation.")

print("1.Add")

print("2.Subtract")

print("3.Multiply")

print("4.Divide")

while True:

choice = input("Enter choice(1/2/3/4): ")

if choice in ('1', '2', '3', '4'):

num1 = float(input("Enter first number: "))

num2 = float(input("Enter second number: "))

if choice == '1':

print(num1, "+", num2, "=", add(num1, num2))

elif choice == '2':

print(num1, "-", num2, "=", subtract(num1, num2))

elif choice == '3':

print(num1, "\*", num2, "=", multiply(num1, num2))

elif choice == '4':

print(num1, "/", num2, "=", divide(num1, num2))

next\_calculation = input("Let's do next calculation? (yes/no): ")

if next\_calculation == "no":

break

else:

print("Invalid Input")

Output:

Select operation.

1.Add

2.Subtract

3.Multiply

4.Divide

Enter choice(1/2/3/4): 1

Enter first number: 2

Enter second number: 4

2.0 + 4.0 = 6.0

Let's do next calculation? (yes/no): no

Weather application using beautiful soup and requests:

from bs4 import BeautifulSoup

import requests

import time

from win10toast import ToastNotifier

headers = {'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/58.0.3029.110 Safari/537.3'}

def weather(city):

city=city.replace(" ","+")

res = requests.get(f'https://www.google.com/search?q={city}&oq={city}&aqs=chrome.0.35i39l2j0l4j46j69i60.6128j1j7&sourceid=chrome&ie=UTF-8',headers=headers)

soup = BeautifulSoup(res.text,'html.parser')

location = soup.select('#wob\_loc')[0].getText().strip()

current\_time = soup.select('#wob\_dts')[0].getText().strip()

info = soup.select('#wob\_dc')[0].getText().strip()

weather = soup.select('#wob\_tm')[0].getText().strip()

information = f"{location} \n {current\_time} \n {info} \n {weather} °C "

toaster = ToastNotifier()

toaster.show\_toast("Weather Information",

f"{information}",

duration=10,

threaded=True)

while toaster.notification\_active(): time.sleep(0.005)

city = "Chennai"

city=city+" weather"

weather(city)

Output:

Chennai,Tamilnadu

Sunday,4:00pm

Partialy cloudy

34oC

Webscraping using beautiful soup:

import requests

from bs4 import BeautifulSoup

site\_url = 'https://myanimelist.net'

response = requests.get(site\_url)

top\_anime\_url = site\_url + '/topanime.php'

response = requests.get(top\_anime\_url)

doc = BeautifulSoup(response.text,"html.parser")

row\_content = doc.find\_all('tr', {'class' : "ranking-list"})

def epi(listt):

result = []

for i in listt[:2]:

r = i.strip()

result.append(r)

return result

topanime = []

for row in row\_content:

epis = epi(row.find('div', class\_ = "information di-ib mt4").text.strip().split('\n'))

ranking = {

'Rank' : row.find('td', class\_ = "rank ac").find('span').text,

'Title': row.find('div', class\_="di-ib clearfix").find('a').text,

'Rating': row.find('td', class\_="score ac fs14").find('span').text,

'Image\_URL': row.find('td', class\_ ='title al va-t word-break').find('img')['data-src'],

'Episodes': epis[0],

'Dates': epis[1]

}

topanime.append(ranking)

def csv(items, path):

with open(path, 'w') as f:

if len(items) == 0:

return

head = list(items[0].keys())

f.write(','.join(head) + '\n')

for item in items:

value = []

for header in head:

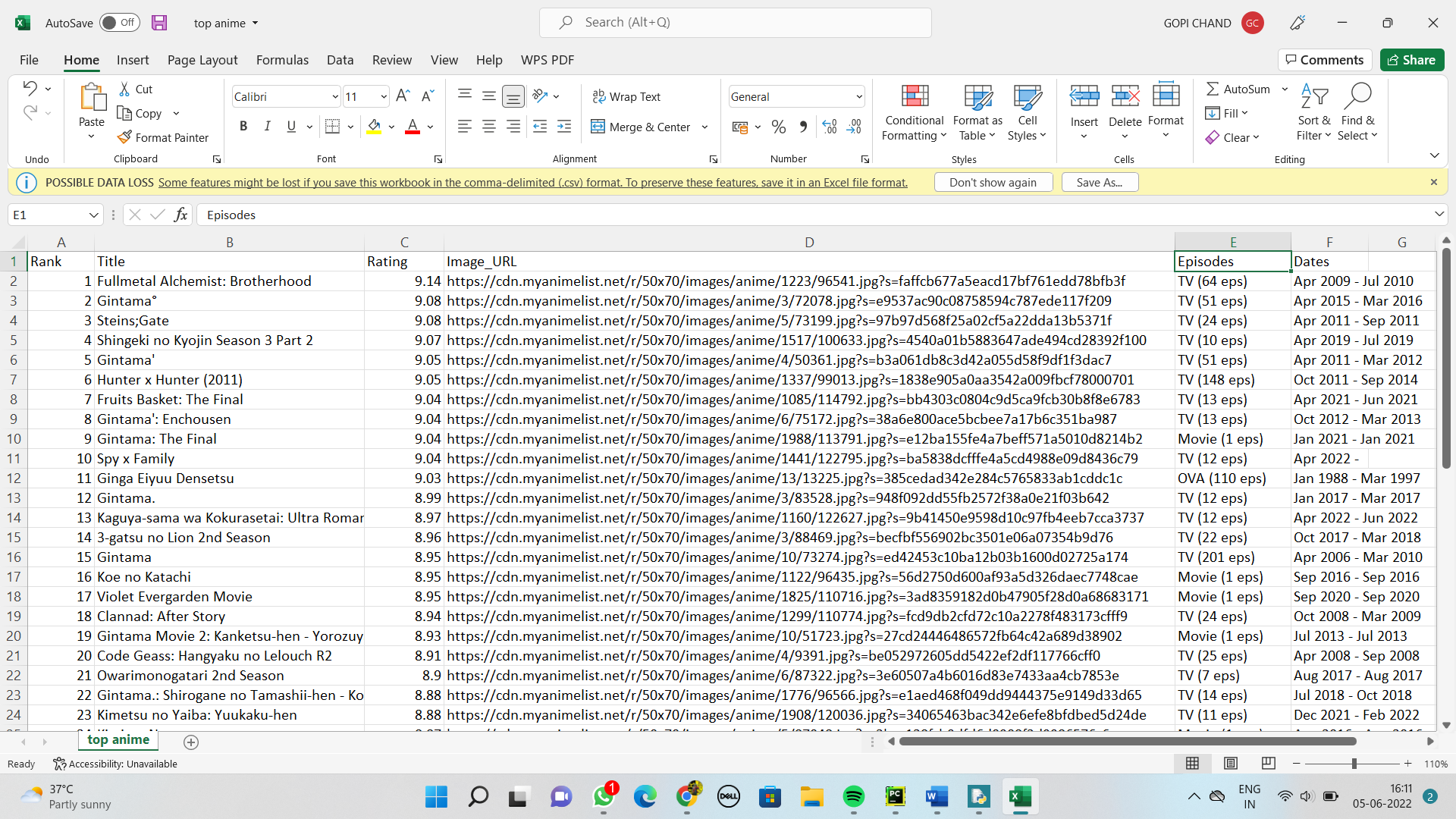
value.append(str(item.get(header, "")))

f.write(','.join(value) + "\n")

csv(topanime, 'top anime.csv')

print("your output is in a csv(excel) where you stored the python file")

Output:



Mechanical soup:

import mechanicalsoup

import os

import wget

browser = mechanicalsoup.StatefulBrowser()

url = "https://www.google.com/imghp?hl=en"

browser.open(url)

browser.select\_form()

search\_term = 'komi san'

browser["q"] = search\_term

response = browser.submit\_selected()

new\_url = browser.get\_url()

page = browser.get\_current\_page()

all\_images = page.find\_all('img')

image\_source = []

for image in all\_images:

image = image.get('src')

image\_source.append(image)

image\_source = [image for image in image\_source if image.startswith('http')]

path = os.getcwd()

path = os.path.join(path, search\_term + "s")

os.mkdir(path)

counter = 0

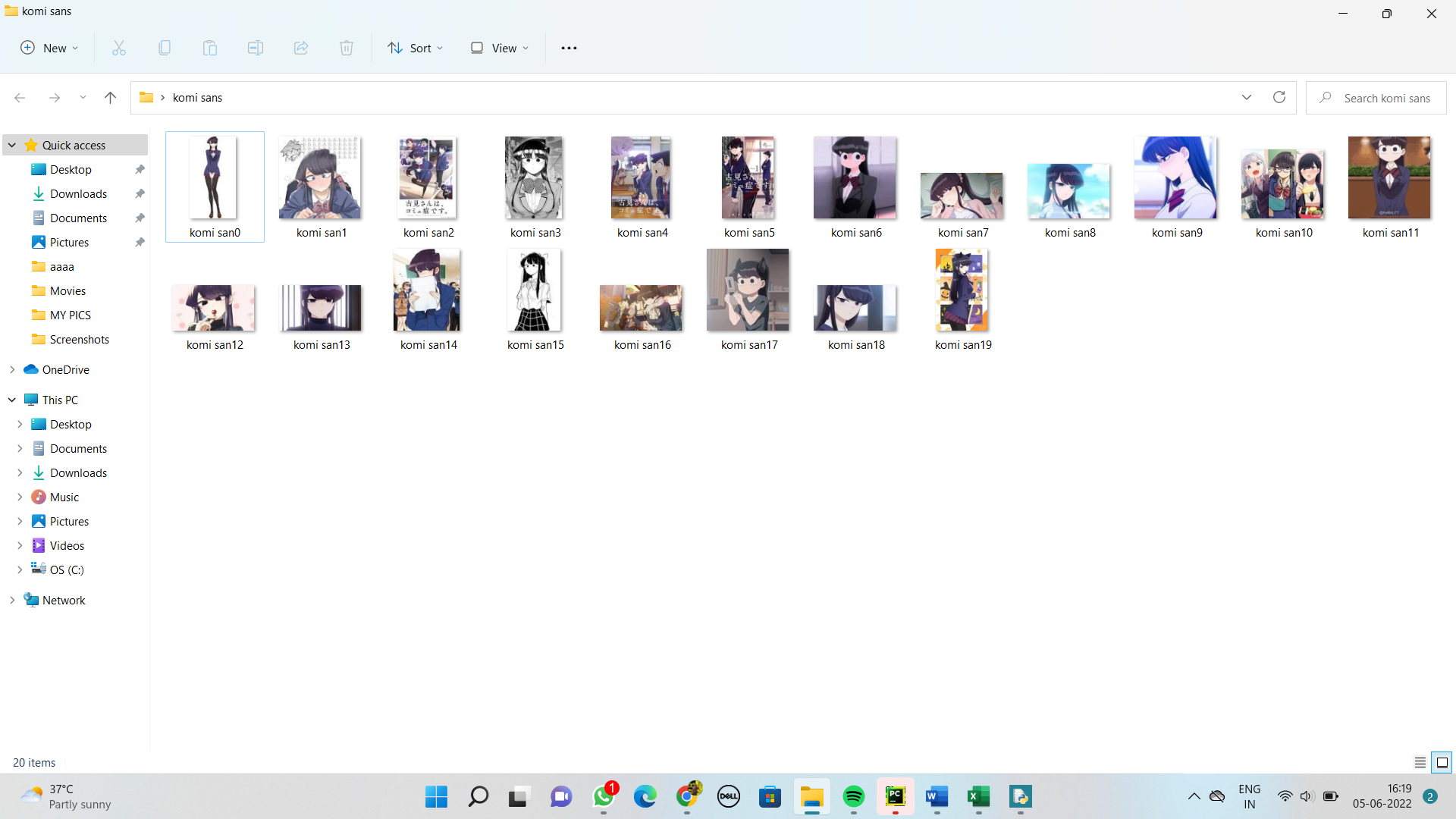
for image in image\_source:

save\_as = os.path.join(path, search\_term + str(counter) + '.jpg')

wget.download(image, save\_as)

counter += 1

Output:



Check url is alive or not:

import requests

url=input("enter url to check:")

a=requests.get(url)

if (a.status\_code == 200):

print("the site is alive")

else:

print("the site is not live")

Output:

enter url to check:https://www.google.com

the site is alive