**DAY 30**

from tkinter import \*

BACKGROUND\_COLOR = "#B1DDC6"

window = Tk()

window.title("Flashy")

window.config(padx=50, pady=50, bg=BACKGROUND\_COLOR)

canvas = Canvas(width=800, height=526)

card\_front\_img = PhotoImage(file="Image/card\_front.jpg") # Make sure to provide the correct path

canvas.create\_image(400, 263, image=card\_front\_img)

canvas.create\_text(400, 150, text="Title", font=("Arial", 40, "italic")) # Correct the font name typo ("Ariel" to "Arial")

canvas.create\_text(400, 263, text="word", font=("Arial", 60, "bold")) # Correct the font name typo ("Ariel" to "Arial")

canvas.config(bg=BACKGROUND\_COLOR, highlightthickness=0) # Fix the config argument to set the background color

canvas.grid(row=0, column=0)

window.mainloop()

**DAY 30\_1**

fruits = ["Apple", "Pear", "Orange"]

def make\_pie(index):

try:

fruit = fruits[index]

except IndexError:

print("Fruit pie")

else:

print(fruit + " pie")

make\_pie(5)

**DAY 30\_2**

facebook\_posts = [

{'Likes': 21, 'Comments': 2},

{'Likes': 13, 'Comments': 2, 'Shares': 1},

{'Likes': 33, 'Comments': 8, 'Share':3},

{'Likes': 4, 'Comments': 2},

{'Likes': 1, 'Comments': 1},

{'Likes': 19, 'Comments': 3},

]

total\_likes = 0

for post in facebook\_posts:

try:

total\_likes = total\_likes + post['likes']

except KeyError:

total\_likes += 0

print(total\_likes)

**DAY 32**

import smtplib

my\_email = "adewalep096@gmail.com"

password = "emperor11@"

try:

connection = smtplib.SMTP("smtp.gmail.com", 587) # 587 is the port number for TLS

connection.starttls()

connection.login(user=my\_email, password=password)

connection.sendmail(from\_addr=my\_email, to\_addrs="adewalep097@gmail.com", msg="Subject:This is the practical sample")

connection.quit() # Use quit() instead of close()

print("Email sent successfully!")

except smtplib.SMTPAuthenticationError:

print("Error: Authentication failed. Check your email and password.")

except smtplib.SMTPException as e:

print("Error sending email:", e)

except Exception as e:

print("An unexpected error occurred:", e)

**DAY 33**

import requests

response = requests.get(url="http://api.open-notify.org/iss-now.json")

# if response.status\_code != 200:

# raise Exception("Bad response from ISS")

# elif response.status\_code == 404:

# raise Exception("You are not authorised to access this data")

# else:

# raise Exception("You are good to go")

response.raise\_for\_status()

data = response.json()

longitude = data["iss\_position"]["longitude"]

latitude = data["iss\_position"]["latitude"]

iss\_position = (longitude, latitude)

print(iss\_position)

#response code, http

**DAY 33\_2**

import time

import requests

from datetime import datetime

import smtplib

MY\_EMAIL = "adewalep096@gmail.com"

MY\_PASSWORD = "emperor11@"

MY\_LAT = 51.507351

MY\_LONG = -0.127758

def is\_iss\_overhead():

response = requests.get(url="http://api.open-notify.org/iss-now.json")

response.raise\_for\_status()

data = response.json

iss\_latitude = float(data["iss\_position"]["latitude"])

iss\_longitude = float(data["iss\_position"]["longitude"])

#your position is within +5 and -5 degrees of the ISS position

if MY\_LAT-5 <= iss\_latitude <= MY\_LAT+5 and MY\_LONG-5 <= iss\_longitude <= MY\_LONG+5:

return True

def is\_night():

parameters = {

"lat": MY\_LAT,

"lng": MY\_LONG,

"formatted": 0,

}

response = requests.get("https://api.sunrise-sunset.org/json", param = parameters)

response.raise\_for\_status()

data = response.json()

sunrise = int(data["results"]["sunrise"].split("T")[1].split(":")[0])

sunset = int(data["results"]["sunset"].split("T")[1].split(":")[0])

time\_now = datetime.now().hour

if time\_now >= sunset or time\_now <= sunrise:

return True

while True:

time.sleep(60)

if is\_iss\_overhead() and is\_night():

connection = smtplib.SMTP("smtp.gmail.com")

connection.starttls()

connection.login((MY\_EMAIL,MY\_PASSWORD))

connection.sendmail(

from\_addr= MY\_EMAIL,

to\_addrs= MY\_EMAIL,

MSG = "Subject:Look Up \n\n The ISS is above you in the sky "

)

**DAY 33\_3**

import requests

response = requests.get(url="http://api.open-notify.org/iss-now.json")

# if response.status\_code != 200:

# raise Exception("Bad response from ISS")

# elif response.status\_code == 404:

# raise Exception("You are not authorised to access this data")

# else:

# raise Exception("You are good to go")

response.raise\_for\_status()

data = response.json()

longitude = data["iss\_position"]["longitude"]

latitude = data["iss\_position"]["latitude"]

iss\_position = (longitude, latitude)

print(iss\_position)

#response code, http

**DAY 35**

import requests

OWM\_Endpoint = "https://api.openweathermap.org/data/2.5/onecall"

api\_key = "69f04e4613056b159c2761a9d9e664d2"

weather\_params = {

"lat": 51.507351,

"lon": -0.127758,

"appid": api\_key,

}

response = requests.get(OWM\_Endpoint, params = weather\_params)

response.raise\_for\_status()

weather\_data = response.json()

weather\_slice = weather\_data["hourly"][:12]

will\_rain = False

for hour\_data in weather\_slice:

Condition\_code = hour\_data["weather"][0]["id"]

if int(Condition\_code) < 700:

will\_rain = True

if will\_rain:

print("Bring an umbrella")

# print(weather\_data["hourly"][0]["weather"][0]["id"])