import pyttsx3

import datetime

import speech\_recognition as sr

import wikipedia

import pyfirmata

import pywhatkit

import imaplib

import email

from email.header import decode\_header

import webbrowser

import os

import smtplib

comport='COM3'

board=pyfirmata.Arduino(comport)

led1=board.get\_pin('d:8:o')

fan=board.get\_pin('d:9:o')

port1=board.get\_pin('d:10:o')

port2=board.get\_pin('d:11:o')

def led\_switch(val):

    if val==1:

        led1.write(1)

    elif val==0:

        led1.write(0)

def fan\_switch(val):

    if val==1:

        fan.write(1)

    elif val==0:

        fan.write(0)

def port1\_switch(val):

    if val==1:

        port1.write(1)

    elif val==0:

        port1.write(0)

def port2\_switch(val):

    if val==1:

        port2.write(1)

    elif val==0:

        port2.write(0)

engine = pyttsx3.init('sapi5')

voices=engine.getProperty("voices")

# print(voices[0].id)

engine.setProperty('voice', voices[0].id)

newVoiceRate = 145

engine.setProperty('rate',newVoiceRate)

def speak(audio):

    engine.say(audio)

    engine.runAndWait()

def wishMe():

    hour=int(datetime.datetime.now().hour)

    if hour>=0 and hour<12:

        speak('Good morning')

    elif hour>12 and hour<18:

        speak('Good afternoon')

    else:

        speak('Good evening')

    speak('I am Friday sir. please tell me how may I help you')

def take\_command():

    r = sr.Recognizer()

    with sr.Microphone() as source:

        print("Listening...")

        r.pause\_threshold = 1

        audio = r.listen(source)

    try:

        print("Recognizing...")

        query = r.recognize\_google(audio, language='en-in')

        print(f"User said: {query}\n")

    except Exception as e:

        # print(e)

        print("Say that again please...")

        return "None"

    return query

def youtube ():

    print(f'user want to watch {topic} ')

    pywhatkit.playonyt(f'{topic}')

def readMail():

            username="raspberry02pi2003@gmail.com"

            password="shubhankar@2003"

            N = 1

            imap = imaplib.IMAP4\_SSL("imap.gmail.com")

            imap.login(username, password)

            status, messages = imap.select("INBOX")

            messages = int(messages[0])

            for i in range(messages, messages-N, -1):

                res, msg = imap.fetch(str(i), "(RFC822)")

                for response in msg:

                    if isinstance(response, tuple):

                        # parse a bytes email into a message object

                        msg = email.message\_from\_bytes(response[1])

                        # decode the email subject

                        subject, encoding = decode\_header(msg["Subject"])[0]

                        if isinstance(subject, bytes):

                            # if it's a bytes, decode to str

                            subject = subject.decode(encoding)

                        # decode email sender

                        From, encoding = decode\_header(msg.get("From"))[0]

                        if isinstance(From, bytes):

                            From = From.decode(encoding)

                        print("Subject:", subject)

                        print("From:", From)

                        # if the email message is multipart

                        if msg.is\_multipart():

                            # iterate over email parts

                            for part in msg.walk():

                                # extract content type of email

                                content\_type = part.get\_content\_type()

                                content\_disposition = str(part.get("Content-Disposition"))

                                try:

                                    # get the email body

                                    body = part.get\_payload(decode=True).decode()

                                except:

                                    pass

                                if content\_type == "text/plain" and "attachment" not in content\_disposition:

                                    # print text/plain emails and skip attachments

                                    print(body)

                                    speak(body)

            imap.close()

            imap.logout()

def sendEmail(to,content):

    server=smtplib.SMTP('smtp.gmail.com',587)

    server.ehlo()

    server.starttls()

    server.login('raspberry02pi2003@gmail.com', 'shubhankar@2003')

    server.sendmail('raspberry02pi2003@gmail.com', to, content)

if \_\_name\_\_ =='\_\_main\_\_':

    # speak('Good afternoon')

    wishMe()

    while True:

        query=take\_command().lower()

        if 'wikipedia' in query:

            speak('searching wikipedia...')

            query=query.replace("wikipedia","")

            results=wikipedia.summary(query,sentences=2)

            speak('According to wikipedia')

            speak(results)

        elif 'light on' in query:

            print('light on......')

            speak('light on........')

            led\_switch(1)

        elif 'light off' in query:

            print('light off......')

            speak('light off........')

            led\_switch(0)

        elif 'turn on fan' in query:

            print('turning on fan.....')

            speak('turning on fan')

            fan\_switch(1)

        elif 'turn off fan' in query:

            print('turning off fan.....')

            speak('light off........')

            fan\_switch(0)

        elif 'on youtube' in query:

            speak("what you want to watch")

            topic=take\_command()

            youtube()

        elif 'open Youtube' in query:

            speak('opening youtube')

            webbrowser.open('https://www.youtube.com/')

        elif 'open google' in query:

            speak('opening google')

            webbrowser.open('https://www.google.com/')

        elif 'open gpp' in query:

            speak('opening login page')

            webbrowser.open('https://gppune.ac.in/gpp/gpp\_s20/userindex.php')

        elif 'play music' in query:

            dir='C:\\Users\\Public\\Music\\Sample Music'

            songs=os.listdir(dir)

            print(songs)

            os.startfile(os.path.join(dir,songs[-1]))

        elif 'the time' in query:

            strTime=datetime.datetime.now().strftime("%H hours %M minutes and %S seconds")

            speak(f'sir the time is{strTime}')

        elif 'stop music' in query:

            os.system("TASKKILL /F /IM wmplayer.exe")

        elif 'read my mail' in query:

            speak('reading your latest emails sir')

            readMail()

        elif 'send email to' in query:

            try:

                speak('what should I say..')

                content=take\_command()

                to='raspberry02pi2003@gmail.com'

                sendEmail(to,content)

                speak('email has been sent')

            except Exception as e:

                print(e)

                speak('sorry sir I am not able to send the mail')

        elif 'exit' in query:

            speak("well see you soon sir Thank you")

            break