import pyttsx3 #for text to speech synthesis

import datetime #for datetime

import speech\_recognition as sr #for speech recognition

import wikipedia #for using wikipedia to search

import smtplib #for sending email

import webbrowser #for funnctionns related to web browser/ serching

import psutil #for cpu and battery

import pyjokes as pj #for jokes

import os #for accessing over the system

import pyautogui as ss #for invoking screenshot and other GUI related features

import json #for saving the news obtained from NewsApi.org

import requests #for fetching data from internet

from urllib.request import urlopen #for opening the link from newsapi.org

import time #for sleep function

import re #for extracting only number from string

vc=pyttsx3.init()

def speak(audio):

    vc.say(audio)

    vc.runAndWait()

def time\_():

    t=datetime.datetime.now().strftime("%H:%M:%S")#%H for 12hrs format and %I for 24 hrs fromat

    speak("the current time is")

    print("the current time is ")

    print(t)

    speak(t)

def month\_change(a): #function to change digit if month to corresponding word

    months={

        1:"January",

        2:"Feburary",

        3:"March",

        4:"April",

        5:"May",

        6:"June",

        7:"July",

        8:"August",

        9:"September",

        10:"October",

        11:"November",

        12:"December",

    }

    return months.get(a)

def date\_():  #function to get the cuurrent date

    yr=datetime.datetime.now().year

    mn=datetime.datetime.now().month

    dt=datetime.datetime.now().day

    speak("the date is")

    print("the date is ")

    speak(dt)

    print(dt,"/",mn,"/",yr)

    s=month\_change(mn)

    speak(s)

    speak(yr)

def greet(): #function to greet according to the time

    hour=datetime.datetime.now().hour

    if hour>=6 and hour<12:

        speak("Good morning, i am Bro")

        print("Good morning, i am Bro")

    elif hour>=12 and hour<=18:

        speak("Good afternoon, i am Bro")

        print("Good afternoon, i am Bro")

    elif hour>=18 and hour<24:

        speak("Good Evening, i am Bro")

        print("Good Evening, i am Bro")

    else:

        speak("Good Night, i am Bro")

        print("Good Night, i am Bro")

    time\_()

    date\_()

def send\_email(receiver,mail\_content): #function to get the simple mail transfer protocol to send a mail

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

    server.ehlo() #helo()

    server.starttls()

    server.login('pantsarla725@gmail.com','9878839980')

    server.sendmail('pantsarla725@gmail.com',receiver,mail\_content)

    server.close()

def cpu():  #function to get the cpu usage percent and battery percentage

    usage=str(psutil.cpu\_percent())

    speak("CPU percentage is"+usage)

    battery=psutil.sensors\_battery()

    speak("Battery is at ")

    speak(battery.percent)

def screenshot(): #function to get screenshot

    image=ss.screenshot()

    image.save('B:/project screenshot/screenshot.png')

    speak('screenshot taken and saved in project screenshot folder')

def take\_commands(): #function to get the commands from user

    r=sr.Recognizer()

    with sr.Microphone() as s:

        print("say command \n")

        speak("say command")

        r.pause\_threshold=1

        r.adjust\_for\_ambient\_noise(s)

        audio=r.listen(s)

    try:

        print("Recognizing... \n")

        speak("Recognizing")

        query=r.recognize\_google(audio,language="en-US")

        print("Input : "+query)

        return query

    except Exception as e:

        print(e)

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

        speak("Say it Again please...")

        return "None"

    return query

if \_\_name\_\_=="\_\_main\_\_":

    greet()

    while True:

        query=take\_commands().lower()

        if 'time' in query:

            time\_()

        elif 'date' in query:

            date\_()

        elif (('wikipedia' in query) or('search' in query) or ('definition' in query)):

            speak("Searching on web")

            result=wikipedia.summary(query,sentences=4)

            speak(result)

        elif 'search in browser'in query:

            speak("what do you want to search?")

            browser\_path="C:/Program Files (x86)/Google/Chrome/Application/chrome.exe %s"

            search=take\_commands().lower()

            webbrowser.get(browser\_path).open\_new\_tab(search)

        elif 'youtube' in query:

            speak("what do you want to search on youtube?")

            Y\_search=take\_commands().lower()

            speak("opening youtube")

            webbrowser.open("https://www.youtube.com/results?search\_query="+Y\_search)

        elif 'google' in query:

            speak("what do you want to search on google?")

            G\_search=take\_commands().lower()

            speak("searching on google")

            webbrowser.open("https://www.google.com/search?q="+G\_search)

        elif "locate" in query :

            speak("specify the city you want to locate")

            location=take\_commands()

            speak("you asked to locate "+location)

            webbrowser.open\_new\_tab("https://www.google.com/maps/place/"+location)

        elif 'cpu status' in query:

            cpu()

        elif "joke" in query:

            speak(pj.get\_joke())

        elif "open ms word" in query:

            speak("opening MS Word")

            word=r'C:/Program Files/Microsoft Office/root/Office16/WINWORD.EXE'

            os.startfile(word)

        elif 'play songs' in query:

            dir='B:/songs'

            music\_list=os.listdir(dir)

            print(music\_list)

            speak("which  one should i play?")

            speak("please say only the number")

            req=take\_commands().lower()

            #song\_no=int(req.replace('number',''))

            #song\_no=int(re.sub('[^0-9]', '', req))

            song\_no = int ( ''.join(filter(str.isdigit, req) ) )

            os.startfile(os.path.join(dir,music\_list[song\_no]))

        elif "write a note" in query:

            speak("what should i write for you?")

            notes=take\_commands()

            file=open('notes.txt','w')

            file.write(notes)

            file.close()

            speak('done taking notes')

        elif 'show my notes' in query:

            file\_r=open('notes.txt','r')

            speak(file\_r.read())

            file\_r.close()

        elif 'take screenshot' in query:

            screenshot()

        elif 'news' in query:

            try:

                jsonObject=urlopen("http://newsapi.org/v2/top-headlines?country=us&category=business&apiKey=34c5c843292948adb8bae84329c1afa0")

                data=json.load(jsonObject)

                i=1

                speak("Here are some top headlines from Business:-")

                for item in data['articles']:

                    print(str(i)+".  "+item['title']+'\n')

                    print(item['description']+'\n')

                    speak(item['title'])

                    i +=1

            except Exception as e:

                print(str(e))

        elif 'remember that' in query:

            speak("what should i remember?")

            rem=take\_commands()

            speak("you asked me to remember that"+rem)

            memory=open('memory.txt','w')

            memory.write(rem)

            memory.close()

        elif 'do you remember anything' in query:

            i\_remember=open('memory.txt','r')

            speak('yes, i remember')

            speak(i\_remember)

            i\_remember.close()

        elif 'send an email' in query:

            try:

                speak("What do you want to write?")

                body=take\_commands()

                receiver\_mail='mukul.mk071@gmail.com' #we can also get the email from the user

                send\_email(receiver\_mail,body)

                speak(body)

                speak("email sent")

            except Exception as e:

                print(e)

                speak("Unable to send email")

        elif 'stop listening' in query:

            try:

                speak('for how many seconds should i sleep')

                ans=int(take\_commands())

                time.sleep(ans)

                print('sleeping for '+ans+' seconds')

            except Exception as e:

                print("invalid input, sleeping for ten seconds")

                time.sleep(10)

        elif (('exit' in query) or ('stop' in query) or ('ok bye' in query) or ('bye bro' in query)):

            speak("ok")

            speak("bye mate")

            exit() #quit() can also be used

        elif 'log out' in query:

            os.system("shutdown -l")

        elif 'restart' in query:

            os.system("shutdown /r /t 1")

        elif 'shutdown' in query:

            os.system("shutdown /s /t 1")