```
from __future__ import print_function
import pickle
import os.path
from googleapiclient.discovery import build
from google_auth_oauthlib.flow import InstalledAppFlow
from google.auth.transport.requests import Request
import webbrowser
import pyttsx3
import datetime
import speech_recognition as sr
import wikipedia
import os
import smtplib
import time
import pyjokes
import subprocess
import winshell
import wolframalpha
from twilio.rest import Client
import pytz

SCOPES = ['https://www.googleapis.com/auth/calendar.readonly']
MONTHS = ["january","february","march","april","may","june","july","august","september","october","november",
"december"]
DAYS = ["monday","tuesday","wednesday","thrusday","friday","saturday","sunday"]
DAYS_EXTENSIONS = ["rd","th","st","nd"]
engine = pyttsx3.init('sapi5')
voices = engine.getProperty('voices')
engine.setProperty('voice', voices[0].id)
engine.setProperty("rate",170)

def speak(audio):

    engine.say(audio)
    engine.runAndWait()

def takeCommand():

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

    except Exception as e:
        print("say that again please....")
        return "None"
```

```
    return query

def wishme():

    query = takeCommand().lower()
    hour = int(datetime.datetime.now().hour)

    if 'jarvis' in query:

        if hour >= 0 and hour < 12:
            speak("Good morning sir!")
            speak("how may i help you")

        elif hour >= 12 and hour < 18:
            speak("Good afternoon sir")
            speak("how may i help you")

        else:
            speak("Good Evening sir")
            speak("how may i help you")


    else:

        exit()

def sendEmail(to,content):

    server = smtplib.SMTP("smtp.gmail.com",587)
    server.ehlo()
    server.starttls()
    server.login("ayushsharma4122002@gmail.com" , 'jaisiyaram0912')
    server.sendmail("ayushsharma4122002@gmail.com" , to , content)
    server.close()

def authenticate_google():

    creds = None

    if os.path.exists('token.pickle'):
        with open('token.pickle', 'rb') as token:
            creds = pickle.load(token)

    if not creds or not creds.valid:

        if creds and creds.expired and creds.refresh_token:
            creds.refresh(Request())

        else:
            flow = InstalledAppFlow.from_client_secrets_file(
                'credentials.json', SCOPES)
            creds = flow.run_local_server(port=0)

        with open('token.pickle', 'wb') as token:
```

```
            pickle.dump(creds, token)

    service = build('calendar', 'v3', credentials=creds)

    return service

def get_events(day,service):

    date = datetime.datetime.combine(day, datetime.datetime.min.time())
    end_date = datetime.datetime.combine(day, datetime.datetime.max.time())
    utc = pytz.UTC
    date = date.astimezone(utc)
    end_date = end_date.astimezone(utc)
    events_result = service.events().list(calendarId='primary', timeMin=date.isoformat(),timeMax=end_date.isoformat
(),
                                          singleEvents=True,
                                          orderBy='startTime').execute()
    events = events_result.get('items', [])

    if not events:
        speak("No upcoming events found.")

    else:
        speak(f"you have {len(events)} on this day")

        for event in events:
            start = event['start'].get('dateTime', event['start'].get('date'))
            print(start, event['summary'])
            start_time = str(start.split("T")[1].split("-")[0])

            if int(start_time.split(":")[0])<12:
                start_time = start_time+"am"

            else:
                start_time = str(int(start_time.split(":")[0])-12)
                start_time = start_time+"pm"
            speak(event["summary"]+"at"+start_time)

def get_date(text):

    text = text.lower()
    today = datetime.date.today()

    if text.count("today")>0:

        return today

    day = -1
    day_of_month = -1
    month = -1
    year = today.year

    for word in text.split():
```

```
        if word in MONTHS:
            month = MONTHS.index(word)+1

        elif word in DAYS:
            day_of_week = DAYS.index(word)

        elif word.isdigit():
            day = int(word)

        else:

            for ext in DAYS_EXTENSIONS:
                found = word.find(ext)

                if found>0:
                    try:
                        day = int(word[:found])

                    except:
                        pass

    if month < today.month and month !=-1:
        year = year+1

    if day < today.day and month == -1 and day != -1:
        month = month+1

    if month == -1 and day == -1 and day_of_week != -1:
        current_day_of_week = today.weekday()
        dif = day_of_week-current_day_of_week

        if dif<0:
            dif+=7

            if text.count("next")>=1:
                dif+=7

        return today+datetime.timedelta(dif)

    if month == -1 or day == -1:

        return None

    return datetime.date(month=month,day=day,year=year)


if __name__ == '__main__':

    SERVICE = authenticate_google()
    wishme()

    while True:

        query = takeCommand().lower()
```

```
if 'wikipedia' in query:
    speak("Searching wikipedia...")
    query = query.replace("wikipedia","")
    result = wikipedia.summary(query,sentences=3)
    speak("According to Wikipedia")
    print(result)
    speak(result)

elif 'how are you' in query:
    speak("I am fine, Thank you")
    speak("How are you, Sir")

elif 'fine' in query or "good" in query:
    speak("It's good to know that your fine")

elif "what's your name" in query or "what is your name" in query:
    speak("My friends call me")
    speak("JARVIS")
    print("My friends call me JARVIS")

elif "who made you" in query or "who created you" in query:
    speak("I have been created by AYUSH SHARMA and team.")

elif "will you be my gf" in query or "will you be my bf" in query:
    speak("I'm not sure about, may be you should give me some time")

elif "i love you" in query:
    speak("It's hard to understand")

elif "who i am" in query:
    speak("If you talk then definately your human.")

elif "why you came to world" in query:
    speak("Thanks to Ayush. further It's a secret")

elif 'is love' in query:
    speak("It is 7th sense that destroy all other senses")

elif "who are you" in query:
    speak("I am your virtual assistant created by Ayush Sharma and team")

elif 'reason for your' in query:
    speak("I was created as a Minor project by Ayush Sharma and team ")

elif 'open youtube' in query:
    speak("okay sir....")
    webbrowser.open("https://www.youtube.com/")

elif 'google' in query:
    speak("alright sir")
    webbrowser.open("https://www.google.com/")

elif 'instagram' in query:
```

```
            speak("login if you, haven't login yet")
            webbrowser.open("https://www.instagram.com/accounts/login/")

        elif 'facebook' in query:
            speak("login if you, haven't login yet")
            webbrowser.open("https://www.facebook.com/")

        elif 'WhatsApp' in query:
            speak("scan the q r code if haven't done yet")
            webbrowser.open("https://web.whatsapp.com/")

        elif 'twitter' in query:
            speak("login if haven't done yet")
            webbrowser.open("https://twitter.com/LOGIN")

        elif 'linkedin' in query:
            speak("login if haven't done yet")
            webbrowser.open("https://linkedin.com")

        elif 'check my schedule' in query:
            speak("of which day sir.")
            text = takeCommand().lower()
            get_events(get_date(text) , SERVICE)

        elif 'yahoo' in query:
            speak("okay sir")
            webbrowser.open("https://in.yahoo.com/?p=us&guccounter=1&guce_referrer=aHR0cHM6Ly93d3cuZ29vZ
2xlLmNvbS8&guce_referrer_sig=AQAAACgLcyE1_TwQyecGDIgd2Jyajkhq6p27miK0-zvnt7flKyEUVJ2ggde737
xu6xx0axScZjHm9sel1ilfDOn-ihCWOshIUEdbuQuo5Z99VdfwwAWL9OaUrNO_QuPj2LgChSFbtiRxePfbmJ6w-f
gdR_Acc2_3O8sPx95bKR-mSkGn")

        elif 'play music' in query:
            speak("how's your mood sir i'll play accordingly, or you want some specific song to be played")

        elif 'broken' in query:
            speak("dim the light the night is going to begin")
            webbrowser.open("https://music.youtube.com/watch?v=NeXbmEnpSz0&list=PLArGT-K2h4YflefxiYFruyx
-wrH5H3GvN")

        elif 'light' in query:
            speak("rolling it sir...")
            webbrowser.open("https://www.youtube.com/watch?v=pe69y-u72DI&list=PLArGT-K2h4YenmKVRsjUhC
ftAc_iHe14s")

        elif 'need peace' in query:
            speak("om shanti")
            webbrowser.open("https://www.youtube.com/watch?v=BOnxCtKA1VE&list=PLArGT-K2h4YejOEYaSNd
0gY9UEfj4cetQ&index=2")

        elif 'time' in query:
            strTime = datetime.datetime.now().strftime("%H:%M:%S")
            speak(f"sir,the time{strTime}")

        elif 'open pycharm' in query:
```

```
    speak("okay sir")
    pycharmpath = "C:\\Program Files\\JetBrains\\PyCharm Community Edition 2020.2.4\\bin\\pycharm64.exe"
    os.system("pycharm")

elif 'my computer' in query:
    speak("okay sir")
    os.system("explorer.exe")

elif "calculate" in query:
    app_id = '8VHQTL-JVP32UJLVL'
    client = wolframalpha.Client(app_id)
    indx = query.lower().split().index('calculate')
    query = query.split()[indx + 1:]
    res = client.query(' '.join(query))
    answer = next(res.results).text
    print("The answer is " + answer)
    speak("The answer is " + answer)

elif "write a note" in query:
    speak("What should i write, sir")
    note = takeCommand()
    file = open('jarvis.txt', 'w')
    file.write(note)
    speak("Done sir")

elif "show the note" in query:
    speak("Showing Notes")
    file = open("jarvis.txt", "r")
    print(file.read())
    speak(file.read(6))

elif 'email' in query:

    try:
        speak("ready to send mail")
        speak("enter the mail to whom you want to send the mail")
        to = input()
        speak("Describe the text content")
        content = takeCommand()
        sendEmail(to,content)
        speak("email has been sent")

    except Exception as e:
        speak("sorry my friend i am not able to send the email")

elif 'search' in query:
    speak("what do you like to search")
    search = takeCommand()
    url = 'http://google.com/search?query='+search
    webbrowser.get().open(url)
    speak('here is what i found for'+ search)

elif 'location' in query:
    speak("Where do you wanna go")
```

```
    location = takeCommand()
    url = 'http://google.nl/maps/place/'+location+'/&amp;'
    webbrowser.get().open(url)
    speak('Here is the location of'+location)

elif 'locate my phone' in query:
    speak("Enter your phone number and look out your phone is going to ring")
    print("Enter your phone number and look out your phone is going to ring")
    account_sid = 'AC9ab5d264c120e85f0ce091957838a8ff'
    auth_token = 'bb94a86a668890b2c2a84b9aa82b58b5'
    client = Client(account_sid, auth_token)
    call = client.calls.create(
        twiml='<Response><Say>hello sir here is your phone</Say></Response>',
        to=input(),
        from_='+15126400822'
    )

elif 'calculator' in query:
    speak("Opening calculator")
    os.system("calc")

elif 'control panel' in query:
    speak("Opening control panel")
    os.system("control panel")

elif 'notepad' in query:
    speak("Opening notepad")
    os.system("notepad")

elif 'cmd' in query:
    speak("opening command prompt")
    os.system("cmd")

elif 'paint' in query:
    speak("opening paint")
    os.system("paint")

elif 'calendar' in query:
    speak("opening calendar")
    os.system("calendar")

elif "don't listen" in query or "stop listening" in query:
    speak("for how much time you want to stop jarvis from listening commands")
    a = int(takeCommand())
    time.sleep(a)
    print(a)

elif 'joke' in query:
    c = pyjokes.get_joke()
    speak(c)
    print(c)

elif "restart" in query:
    subprocess.call(["shutdown", "/r"])
```

```
elif "hibernate" in query or "sleep" in query:
    speak("Hibernating")
    subprocess.call("shutdown / h")

elif 'empty recycle bin' in query:
    winshell.recycle_bin().empty(confirm=False, show_progress=False, sound=True)
    speak("Recycle Bin Recycled")

elif 'bye' in query:
    speak("call me whenever you want, till then, goodbyee....")
    exit()
```