

## *My Personal Virtual Assistant - ZARA*

```
import pyttsx3
import speech_recognition as sr
import wikipedia
import datetime
import webbrowser
import os
import smtplib
import time
import sys
import operator
import requests
import json
import phonenumbers
import socket
import random
import cv2
import face_recognition

print("initializing your personal assistant")

engine = pyttsx3.init('sapi5')
voices = engine.getProperty('voices')
engine.setProperty('voice', voices[0].id)

def speak(audio):
    engine.say(audio)
    engine.runAndWait()

from pygame import mixer
mixer.init()
mixer.music.load("C:\\\\Users\\\\path_to_your_file")
mixer.music.play()

time.sleep(15)

def wishMe():
    speak(' i am your personal assistant ZARA with tremendous speed')
    speak('initializing myself getting started')
    speak('i am online now')
    res = requests.get('https://ipinfo.io/')
    data = res.json()
    city = data['city']
    speak(f'your current location is {city}')
    speak(f'sir current weather conditions here are:')
```

```
MAIN = "https://openweathermap.org/data/2.5/weather?q="
API = "your_api_key"
URL = MAIN + city + "&appid=" + API
response = requests.get(URL)
```

```
if response.status_code==200:
    data = response.json()
    main = data['main']
    temprature = main['temp']
    humidity = main['humidity']
    report = data['weather']
    speak(f"teprature is { temprature } sir")
    speak(f"humidity is { humidity }")
    speak(f"weather report is {report[0]['description']}")
```

```
hour = int(datetime.datetime.now().hour)
if hour>=0 and hour<12:
    speak("Good Morning sir")
    speak("welcome")
elif hour>=12 and hour<16:
    speak("Good Afternoon sir")
    speak("welcome")
elif hour>=16 and hour<24:
    speak("Good Evening sir")
    speak("welcome")
speak("How may i help you")
```

```
def takeCommand():
    r = sr.Recognizer()
    with sr.Microphone() as source:
        print("Listening...")
        r.energy_threshold = 4000
        r.pause_threshold = 1
        audio = r.listen(source)

    try:
        print("Recognizing...")
        query = r.recognize_google(audio, language = 'en-in')
        print(f"You: {query.lower()}\n")

    except Exception as e:
        speak("Sorry, Please say that again")
        return "None"

    return query
```

```
def sendEmail(do, content):
    server = smtplib.SMTP('smtp.gmail.com', 587)
```

```
server.ehlo()
server.starttls()
server.login('your_mail@gmail.com', 'your_mail_password')
server.sendmail('your_mail@gmail.com', to, content)
server.close()
```

```
def moveOn():
    speak('sir, do you want something more or should i sleep sir')
```

```
def flightCompare():
    print(f'welcome to flight price compare corner!!!!')
    speak('Enter the source destination:')
    start_dest = input('Enter the source destination:')
```

```
start_dest = start_dest.lower()
```

```
if start_dest == "agartala":
    start_code = "IXA"
if start_dest == "agatti":
    start_code = "AGX"
if start_dest == "agra":
    start_code = "AGR"
if start_dest == "akola":
    start_code = "AKD"
if start_dest == "allahabad":
    start_code = "IXD"
if start_dest == "arunachal pradesh":
    start_code = "IXV"
if start_dest == "aurangabad":
    start_code = "IXU"
if start_dest == "siliguri":
    start_code = "IXB"
if start_dest == "bareilly":
    start_code = "BEK"
if start_dest == "hyderabad":
    start_code = "HYD"
if start_dest == "belgaum":
    start_code = "IXG"
if start_dest == "bellary":
    start_code = "BEP"
if start_dest == "bathinda":
    start_code = "BUP"
if start_dest == "bhavnagar":
    start_code = "BHU"
if start_dest == "bhuj":
    start_code = "BHJ"
```

```
if start_dest == "bhubneshwar":
    start_code = "BBI"
if start_dest == "bilaspur":
    start_code = "PAB"
if start_dest == "ranchi":
    start_code = "IXR"
if start_dest == "kolkata":
    start_code = "CCU"
if start_dest == "car nicobar":
    start_code = "CBD"
if start_dest == "chandigarh":
    start_code = "IXC"
if start_dest == "lucknow":
    start_code = "LKO"
if start_dest == "chennai":
    start_code = "MAA"
if start_dest == "mumbai":
    start_code = "BOM"
if start_dest == "kochi":
    start_code = "COK"
if start_dest == "coimbatore":
    start_code = "CJB"
if start_dest == "goa":
    start_code = "GOI"
if start_dest == "daman and diu":
    start_code = "NMB"
if start_dest == "dehradun":
    start_code = "DED"
if start_dest == "indore":
    start_code = "IDR"
if start_dest == "dhanbad":
    start_code = "DBD"
if start_dest == "dibrugarh":
    start_code = "DIB"
if start_dest == "dimapur":
    start_code = "DMU"
if start_dest == "nagpur":
    start_code = "NAG"
if start_dest == "bihar":
    start_code = "GAY"
if start_dest == "gorakhpur":
    start_code = "GOP"
if start_dest == "gwalior":
    start_code = "GWL"
if start_dest == "haryana":
    start_code = "HSS"
```

```
if start_dest == "hubli":
    start_code = "HBX"
if start_dest == "imphal":
    start_code = "IMF"
if start_dest == "delhi":
    start_code = "DEL"
if start_dest == "new delhi":
    start_code = "DEL"
if start_dest == "madhya pradesh":
    start_code = "JLR"
if start_dest == "jaipur":
    start_code = "JAI"
if start_dest == "jaisalmer":
    start_code = "JSA"
if start_dest == "jammu":
    start_code = "IXJ"
if start_dest == "jamnagar":
    start_code = "JGA"
if start_dest == "jodhpur":
    start_code = "JDH"
if start_dest == "jorhat":
    start_code = "JRH"
if start_dest == "kadapa":
    start_code = "CDP"
if start_dest == "kailashahar":
    start_code = "IXH"
if start_dest == "kandla":
    start_code = "IXY"
if start_dest == "kangra":
    start_code = "DHM"
if start_dest == "kannur":
    start_code = "CNN"
if start_dest == "kanpur":
    start_code = "KAN"
if start_dest == "durgapur":
    start_code = "RDP"
if start_dest == "banglore":
    start_code = "BLR"
if start_dest == "bengaluru":
    start_code = "BLR"
if start_dest == "keshod":
    start_code = "IXK"
if start_dest == "khajuraho":
    start_code = "HJR"
if start_dest == "ajmer":
    start_code = "KQH"
```

```
if start_dest == "maharashtra":
    start_code = "KLH"
if start_dest == "kota":
    start_code = "KTU"
if start_dest == "bhuntar":
    start_code = "KUU"
if start_dest == "varanasi":
    start_code = "VNS"
if start_dest == "leh":
    start_code = "IXL"
if start_dest == "aizwal":
    start_code = "AJL"
if start_dest == "patna":
    start_code = "PAT"
if start_dest == "guwahati":
    start_code = "GAU"
if start_dest == "ludhiana":
    start_code = "LUH"
if start_dest == "madurai":
    start_code = "IXM"
if start_dest == "udaipur":
    start_code = "UDR"
if start_dest == "manglore":
    start_code = "IXE"
if start_dest == "latur":
    start_code = "LTU"
if start_dest == "muzaffarpur":
    start_code = "MZU"
if start_dest == "bikaner":
    start_code = "BKB"
if start_dest == "nanded":
    start_code = "NDC"
if start_dest == "nasik":
    start_code = "ISK"
if start_dest == "pathankot":
    start_code = "IXP"
if start_dest == "pondicherry":
    start_code = "PNY"
if start_dest == "porbandar":
    start_code = "PND"
if start_dest == "pune":
    start_code = "PNQ"
if start_dest == "bhopal":
    start_code = "BHO"
if start_dest == "rajkot":
    start_code = "RAJ"
```

```
if start_dest == "ratnagiri":
    start_code = "RTC"
if start_dest == "ahemdabad":
    start_code = "AMD"
if start_dest == "srinagar":
    start_code = "SXR"
if start_dest == "shillong":
    start_code = "SHL"
if start_dest == "shirdi":
    start_code = "SXR"
if start_dest == "silchar":
    start_code = "IXS"
if start_dest == "jamshedpur":
    start_code = "IXW"
if start_dest == "amritsar":
    start_code = "ATQ"
if start_dest == "surat":
    start_code = "STV"
if start_dest == "tiruchirappaly":
    start_code = "TRZ"
if start_dest == "thiruvanthpuram":
    start_code = "TRV"
if start_dest == "vadodra":
    start_code = "VGA"

speak('Enter the end destination:')
end_dest = input('Enter the end destination:')
```

```
end_dest = end_dest.lower()
if end_dest == "agartala":
    end_code = "IXA"
if end_dest == "agatti":
    end_code = "AGX"
if end_dest == "agra":
    end_code = "AGR"
if end_dest == "akola":
    end_code = "AKD"
if end_dest == "allahabad":
    end_code = "IXD"
if end_dest == "arunachal pradesh":
    end_code = "IXV"
if end_dest == "aurangabad":
    end_code = "IXU"
if end_dest == "siliguri":
    end_code = "IXB"
```

```
if end_dest == "bareilly":
    end_code = "BEK"
if end_dest == "hyderabad":
    end_code = "HYD"
if end_dest == "belgaum":
    end_code = "IXG"
if end_dest == "bellary":
    end_code = "BEP"
if end_dest == "bathinda":
    end_code = "BUP"
if end_dest == "bhavnagar":
    end_code = "BHU"
if end_dest == "bhuj":
    end_code = "BHJ"
if end_dest == "bhubneshwar":
    end_code = "BBI"
if end_dest == "bilaspur":
    end_code = "PAB"
if end_dest == "ranchi":
    end_code = "IXR"
if end_dest == "kolkata":
    end_code = "CCU"
if end_dest == "car nicobar":
    end_code = "CBD"
if end_dest == "chandigarh":
    end_code = "IXC"
if end_dest == "lucknow":
    end_code = "LKO"
if end_dest == "chennai":
    end_code = "MAA"
if end_dest == "mumbai":
    end_code = "BOM"
if end_dest == "kochi":
    end_code = "COK"
if end_dest == "coimbatore":
    end_code = "CJB"
if end_dest == "goa":
    end_code = "GOI"
if end_dest == "daman and diu":
    end_code = "NMB"
if end_dest == "dehradun":
    end_code = "DED"
if end_dest == "indore":
    end_code = "IDR"
if end_dest == "dhanbad":
    end_code = "DBD"
```



```
if end_dest == "dibrugarh":
    end_code = "DIB"
if end_dest == "Dimapur":
    end_code = "DMU"
if end_dest == "nagpur":
    end_code = "NAG"
if end_dest == "bihar":
    end_code = "GAY"
if end_dest == "gorakhpur":
    end_code = "GOP"
if end_dest == "gwalior":
    end_code = "GWL"
if end_dest == "haryana":
    end_code = "HSS"
if end_dest == "hubli":
    end_code = "HBX"
if end_dest == "imphal":
    end_code = "IMF"
if end_dest == "delhi":
    end_code = "DEL"
if end_dest == "new delhi":
    end_code = "DEL"
if end_dest == "madhya pradesh":
    end_code = "JLR"
if end_dest == "jaipur":
    end_code = "JAI"
if end_dest == "jaisalmer":
    end_code = "JSA"
if end_dest == "jammu":
    end_code = "IXJ"
if end_dest == "jamnagar":
    end_code = "JGA"
if end_dest == "jodhpur":
    end_code = "JDH"
if end_dest == "jorhat":
    end_code = "JRH"
if end_dest == "kadapa":
    end_code = "CDP"
if end_dest == "kailashahar":
    end_code = "IXH"
if end_dest == "kandla":
    end_code = "IXY"
if end_dest == "kangra":
    end_code = "DHM"
if end_dest == "kannur":
    end_code = "CNN"
```

```
if end_dest == "kanpur":
    end_code = "KAN"
if end_dest == "durgapur":
    end_code = "RDP"
if end_dest == "banglore":
    end_code = "BLR"
if end_dest == "keshod":
    end_code = "IXK"
if end_dest == "khajuraho":
    end_code = "HJR"
if end_dest == "ajmer":
    end_code = "KQH"
if end_dest == "maharashtra":
    end_code = "KLH"
if end_dest == "kota":
    end_code = "KTU"
if end_dest == "bhuntar":
    end_code = "KUU"
if end_dest == "varanasi":
    end_code = "VNS"
if end_dest == "leh":
    end_code = "IXL"
if end_dest == "aizwal":
    end_code = "AJL"
if end_dest == "patna":
    end_code = "PAT"
if end_dest == "guwahati":
    end_code = "GAU"
if end_dest == "ludhiana":
    end_code = "LUH"
if end_dest == "madurai":
    end_code = "IXM"
if end_dest == "udaipur":
    end_code = "UDR"
if end_dest == "manglore":
    end_code = "IXE"
if end_dest == "latur":
    end_code = "LTU"
if end_dest == "muzaffarpur":
    end_code = "MZU"
if end_dest == "bikaner":
    end_code = "BKB"
if end_dest == "nanded":
    end_code = "NDC"
if end_dest == "nasik":
    end_code = "ISK"
```

```
if end_dest == "pathankot":
    end_code = "IXP"
if end_dest == "pondicherry":
    end_code = "PNY"
if end_dest == "porbandar":
    end_code = "PND"
if end_dest == "pune":
    end_code = "PNQ"
if end_dest == "bhopal":
    end_code = "BHO"
if end_dest == "rajkot":
    end_code = "RAJ"
if end_dest == "ratnagiri":
    end_code = "RTC"
if end_dest == "ahemdabad":
    end_code = "AMD"
if end_dest == "srinagar":
    end_code = "SXR"
if end_dest == "shillong":
    end_code = "SHL"
if end_dest == "shirdi":
    end_code = "SXR"
if end_dest == "silchar":
    end_code = "IXS"
if end_dest == "jamshedpur":
    end_code = "IXW"
if end_dest == "amritsar":
    end_code = "ATQ"
if end_dest == "surat":
    end_code = "STV"
if end_dest == "tiruchirappaly":
    end_code = "TRZ"
if end_dest == "thiruvanthpuram":
    end_code = "TRV"
if end_dest == "vadodra":
    end_code = "VGA"
```

```
speak('Enter the year in which you want to travel:')
year = input('Enter the year in which you want to travel:')
print(year)
speak('Enter the month in which you want to travel:')
month = input('Enter the month in which you want to travel:')
print(month)
speak('Enter the day in which you want to travel:')
day = input('Enter the day in which you want to travel:')
print(day)
```

```

date = (f'{year}-{month}-{day}')
print(f'your going date is : {date}')
speak(f'your going date is : {date}')

```

```

url = f"https://skyscanner-skyscanner-flight-search-
v1.p.rapidapi.com/apiservices/browsedates/v1.0/IN/INR/en-
US/{start_code}/{end_code}/{date}"

```

```

headers = {
    'x-rapidapi-host': "skyscanner-skyscanner-flight-search-v1.p.rapidapi.com",
    'x-rapidapi-key': "your_API_key"
}

```

```

response = requests.request("GET", url, headers=headers, params="-")
data = response.json()
price = data['Dates']['OutboundDates'][0]['Price']
updated = data['Dates']['OutboundDates'][0]['QuoteDateTime']
direct = data['Quotes'][0]['Direct']
currency = data['Currencies'][0]['Symbol']
carriers = data['Carriers'][0]['Name']
print(f'The best price for your travel from {start_dest.upper()}({start_code}) to
{end_dest.upper()}({end_code}) is:')
speak(f'The best price for your travel from {start_dest.upper()}({start_code}) to
{end_dest.upper()}({end_code}) is:')
print(f'{currency} {price}\t{carriers}')
speak(f'{currency} {price} through {carriers}')
print(f'flight is direct : {direct}')
speak(f'flight is direct : {direct}')
print(f'last updated on : {updated}')

```

```

def zodiac(day, month):
    if month == 'december':
        astro_sign = 'Sagittarius' if (day < 22) else 'capricorn'
    elif month == 'january':
        astro_sign = 'Capricorn' if (day < 20) else 'aquarius'
    elif month == 'february':
        astro_sign = 'Aquarius' if (day < 19) else 'pisces'
    elif month == 'march':
        astro_sign = 'Pisces' if (day < 21) else 'aries'
    elif month == 'april':
        astro_sign = 'Aries' if (day < 20) else 'taurus'
    elif month == 'may':
        astro_sign = 'Taurus' if (day < 21) else 'gemini'
    elif month == 'june':
        astro_sign = 'Gemini' if (day < 21) else 'cancer'

```

```

elif month == 'july':
    astro_sign = 'Cancer' if (day < 23) else 'leo'
elif month == 'august':
    astro_sign = 'Leo' if (day < 23) else 'virgo'
elif month == 'september':
    astro_sign = 'Virgo' if (day < 23) else 'libra'
elif month == 'october':
    astro_sign = 'Libra' if (day < 23) else 'scorpio'
elif month == 'november':
    astro_sign = 'scorpio' if (day < 22) else 'sagittarius'
print(f"Your Astrological sign is {astro_sign}")
speak(f"Your Astrological sign is {astro_sign} sir")

url = "https://sameer-kumar-aztro-v1.p.rapidapi.com/"
querystring = {"sign":astro_sign,"day":"today"}
payload = ""
headers = {
    'x-rapidapi-host': "sameer-kumar-aztro-v1.p.rapidapi.com",
    'x-rapidapi-key': "your_API_key",
    'content-type': "application/x-www-form-urlencoded"
}
res = requests.request("POST", url, data=payload, headers=headers,
params=querystring)
data = res.json()
prediction = data['description']
luckno = data['lucky_number']
close = data['compatibility']
color = data['color']
print(prediction)
speak(prediction)
time.sleep(2)
print(f' lucky number is {luckno}')
speak(f' lucky number is {luckno}')
time.sleep(2)
print(f' sir you are most comapaitable to {close} today')
speak(f' sir you are most comapaitable to {close} today')
time.sleep(1)
print(f' you should wear {color} clothes this is lucky for you it will bring charm to
you')
speak(f' you should wear {color} clothes this is lucky for you it will bring charm to
you')

def authentication():
    videoCaptureObject = cv2.VideoCapture(0)
    result = True
    while(result):

```

```

        speak('welcome sir')
        speak('please authenticate first to get started')
        speak(f'press enter to capture image')
        take = input(f'press enter to capture image:')
        print(f'get ready for image detection')
        speak(f'get ready for image detection')
        print(f'image will be captured in 3 seconds..... get ready now')
        speak(f'image will be captured in 3 seconds..... get ready now')
        time.sleep(3)
        ret,frame = videoCaptureObject.read()
        cv2.imwrite("C:\\Users\\your_ownimage_file_path",frame)
        result = False
    videoCaptureObject.release()
    cv2.destroyAllWindows()

    speak('your image is captured and processing now')
    image_of_jitu = face_recognition.load_image_file('./your_image_file.jpg')
    jitu_face_encoding = face_recognition.face_encodings(image_of_jitu)[0]

    unknown_image = face_recognition.load_image_file('./current_image_file.jpg')
    unknown_face_encoding = face_recognition.face_encodings(unknown_image)[0]

    results = face_recognition.compare_faces([jitu_face_encoding],
unknown_face_encoding)

    if results[0]:
        speak('you are authorized sir.')
        speak('welcome how are you?')
    else:
        speak('sorry you are not authorized. you are not jitender')
        speak('please try again later')
        sys.exit()

authentication()

if __name__ == "__main__":
    wishMe()
    while True:
        speak('say now sir')
        query = takeCommand()
        if 'wikipedia' in query.lower():
            speak('Searching wikipedia')
            query = query.replace("wikipedia", "")
            results = wikipedia.summary(query, sentences=4)
            speak("Acoording to wikipedia")
            print(results)

```

```
speak(results)
speak('here are the results for')
speak(query)
moveOn()
continue
```

```
elif 'search youtube' in query.lower():
    speak('searching Youtube')
    new=2
    query = query.replace("search youtube", "")
    url="https://www.youtube.com/results?search_query=";
    speak('searching for you sir')
    webbrowser.open(url+query,new=new);
    speak('here are the results sir for')
    speak(query)
    moveOn()
    continue
```

```
elif 'search' in query.lower():
    new=2
    query = query.replace("search", "")
    url="http://google.com/?#q=";
    speak('searching for you sir')
    webbrowser.open(url+query,new=new);
    speak('here are the results sir for')
    speak(query)
    moveOn()
    continue
```

```
elif 'play some music' in query.lower():
    music_dir = 'C:\\Users\\your_songs_path'
    songs = os.listdir(music_dir)
    list = random.randint(0, 100)
    os.startfile(os.path.join(music_dir, songs[list]))
    moveOn()
    continue
```

```
elif 'open your code' in query.lower():
    codePath = 'C:\\Users\\your_code_path'
    speak('opening my code')
    os.startfile(codePath)
    speak('sir, do you want something more or should i sleep sir')
    continue
```

```

elif 'date and time' in query.lower():
    from datetime import date
    strTime = datetime.datetime.now().strftime("%H:%M:%S")
    strDate = date.today()
    speak(f"today is {strDate}")
    speak(f"Sir, the time is {strTime}")
    moveOn()
    continue

```

```

elif 'send email' in query.lower():
    try:
        speak("to whom do you want to send mail?")
        speak("write the mail address sir")
        to = input("enter mail address sir:")
        print(f'You entered {to}')
        speak(to)
        speak("what should i say?")
        content = takeCommand()
        sendEmail(to, content)
        speak("Email has been sent")
    except Exception as e:
        speak(" sorry, i am not able to send your mail")
    moveOn()
    continue

```

```

elif 'sleep now' in query.lower():
    speak('i am sleeping for 20 seconds sir after that i will come back to
you sir')

    time.sleep(20)
    speak('i wake up sir,please tell me what should i do...')
    continue

```

```

elif 'shutdown' in query.lower():
    speak('i am stopping now sir')
    sys.exit()

```

```

elif 'open music file' in query.lower():
    os.startfile("C:\\Users\\your_songs_path")
    speak('here are your files sir')
    moveOn()
    continue

```



```
elif 'open movies' in query.lower():
    os.startfile("your_movies_path")
    speak('here are your files sir')
    moveOn()
    continue
```

```
elif 'open images' in query.lower():
    os.startfile("your_images_path")
    speak('here are your files sir')
    moveOn()
    continue
```

```
elif 'tell me some jokes' in query.lower():
    speak('Bap: Tumhari abhi pitai karti ho tumne nails kyu nahi cut
kiye?')
    speak('Beta: main toh subha roz cut karta hoon par van ka driver itni
slow drive karta hai ke rastey mein hi nails badh jate hai. ...')
    speak('Sardar Ji: humne Mobile Marriage Bureau shuru kiya hai:
Rishtey k liye press 1, Mangni k liye press 2, Shadi k liye press 3.')
    speak('Man: mai 2nd Shadi k liye kya press karu?')
    speak('Sardar Ji: 2nd shadi k liye pehle wali ka gala press karo sir ji
..!')
    moveOn()
    continue
```

```
elif 'open gmail' in query.lower():
    speak('Opening your gmail sir')
```

```
webbrowser.open("https://mail.google.com/mail/u/0/?tab=rm&ogbl#inbox")
    moveOn()
    continue
```

```
elif 'open google' in query.lower():
    speak('opening google')
    webbrowser.open("https://www.google.com")
    moveOn()
    continue
```

```

        elif 'play some romantic track' in query.lower():
            speak('sir enjoy the music i am sleeping sir i will light up after u
enjoyed all these songs !!! have a good day sir')
            from pygame import mixer
            mixer.init()
            mixer.music.load("C:\\Users\\songs_path\\romantic.mp3")
            mixer.music.play()
            time.sleep(4000)
            speak('i m back hope you enjoyed sir what else can i do now sir for
you')

            moveOn()
            continue

```

```

        elif 'calculate' in query.lower():
            def get_operator_fn(op):
                return {
                    '+' : operator.add,
                    'add' : operator.add,
                    '-' : operator.sub,
                    'subtract' : operator.sub,
                    'minus' : operator.sub,
                    'x' : operator.mul,
                    'multiply' : operator.mul,
                    '/' : operator.truediv,
                    'divide' : operator.truediv,
                    'divided' : operator.truediv,
                    'divide by' : operator.truediv,
                }[op]

```

```

            while query:
                query = query.replace("calculate", "")
                def eval_binary_expr(op1, oper, op2):
                    op1,op2 = int(op1), int(op2)
                    return get_operator_fn(oper)(op1, op2)
                speak('your answer is')
                speak(eval_binary_expr(*(query.split())))
                break

            moveOn()
            continue

```

```

        elif 'shutdown the system' in query.lower():
            speak('sir closing all windows and shutting down sir')
            os.system("shutdown /s /t 1");
            continue

```

```

elif 'restart the system' in query.lower():
    speak('closing all windows and starting the system')
    os.system("shutdown /r /t 1");
    continue

elif 'current weather of' in query.lower():
    speak('current weather conditions for')
    query = query.replace("current weather of", "")
    MAIN = "https://openweathermap.org/data/2.5/weather?q="
    API = "your_API_key"
    URL = MAIN + query + "&appid=" + API
    response = requests.get(URL)

    if response.status_code==200:
        data = response.json()
        main = data['main']
        temprature = main['temp']
        humidity = main['humidity']
        pressure = main['pressure']
        report = data['weather']
        speak(f"city {query} is")
        speak(f"teprature is {temprature} sir")
        speak(f"humidity is {humidity}")
        speak(f"pressure is {pressure}")
        speak(f"weather report is {report[0]['description']}")
    else:
        speak('error in your net connection sir')
    moveOn()
    continue

elif 'list c drive images' in query.lower():
    for root, dirs, files in os.walk('C:\\path_name'):
        for file in files:
            if file.endswith('.jpg'):
                print(os.path.join(root, file))
            elif file.endswith('.txt'):
                print(os.path.join(root, file))
    moveOn()
    continue

elif 'list files' in query.lower():
    for root, dirs, files in os.walk("D:\\"):
        for filename in files:
            print(filename)

```

```
moveOn()
continue
```

```
elif 'translate' in query.lower():
    from googletrans import Translator
    trans = Translator()
    speak('sir enter the text what do you want to translate')
    word = input("enter the text:")
    print(word)
    speak(word)
    print("source language is english")
    speak('languages which can be translated by me sir are french
german italian russian spanish only sir')
    speak('i can do more if my master willl alloww me')
    speak('enter the language in which you want to translate sir')
    lan = input("enter the language in which you want to translate sir:")
    if lan == 'french':
        t = trans.translate( word, src = 'en' , dest = 'fr')
    elif lan == 'german':
        t = trans.translate( word, src = 'en' , dest = 'de')
    elif lan == 'italian':
        t = trans.translate( word, src = 'en' , dest = 'it')
    elif lan == 'russian':
        t = trans.translate( word, src = 'en' , dest = 'ru')
    elif lan == 'spanish':
        t = trans.translate( word, src = 'en' , dest = 'es')
    else:
        speak('language is not present in the system sorry sir')
    speak('source language is english')
    speak('traslated language is')
    speak(lan)
    print(f'{t.origin} -----> {t.text}')
    speak('translated text is')
    speak(t.text)
    moveOn()
    continue
```

```
elif 'features' in query.lower():
    from pygame import mixer
    mixer.init()
    mixer.music.load("C:\\Users\\file_path_name\\describe.mp3")
    mixer.music.play()
    time.sleep(7)
    speak('i am jitender personal assistant zara')
    speak('i can perform various tasks')
```

```

not')
speak('i first authorize the user to check wheter he s my master or
even mirror for you')
speak('i can tell you your horoscope also, i can open notepad and
speak('i can search on google, search on youtube for your favourites
videos ,search music files,movie files,text files,and whatever you want')
speak('i can even play music files,translate english into many
languages,perform calculation')
speak('i can also predict weather, send emails, shutdown the whole
sytem and boot it again')
speak('i can find the loacation of country and service provider also
for a given mobile no')
speak('i will definitely generate the ip address of my master sytem
with his machine name it willl help him')
speak('as the covid-19 is current affair i can also give covid details')
speak('i am one of the advanced personal virtual assistant')
speak('i am still at development periods so many new tasks are
waiting for me')
speak('i will be able to perform them very soon')
speak('my master words are my commands i will do everything for
him, i care for my master')
speak('now i am going to serve my master again, i am ready for
serving')
continue

```

```

elif 'mobile' in query.lower():
    from phonenumbers import geocoder
    from phonenumbers import carrier
    mobileNo = input("enter your mobile number:")
    print(mobileNo)
    speak(mobileNo)
    phone_number = phonenumbers.parse(mobileNo)
    service_provider = phonenumbers.parse(mobileNo)

    print(geocoder.description_for_number(phone_number,'en'))
    speak(geocoder.description_for_number(phone_number,'en'))
    print(carrier.name_for_number(service_provider,'en'))
    speak(carrier.name_for_number(service_provider,'en'))
    moveOn()
    continue

```

```

elif 'my ip' in query.lower():
    hostname = socket.gethostname()
    IPAddr = socket.gethostbyname(hostname)
    print(f"Your Computer Name is: {hostname}")
    print(f"Your Computer IP Address is: {IPAddr}")

```

```

        moveOn()
        continue

    elif 'covid-19 cases' in query.lower():
        url = "https://corona-virus-world-and-india-
data.p.rapidapi.com/api_india"

        headers = {
                                'x-rapidapi-host': "corona-virus-world-and-india-
data.p.rapidapi.com",
                                'x-rapidapi-key': "your_API_key"
        }

        response = requests.request("GET", url, headers=headers).json()
        speak('enter the city of any state sir i will show you the details sir')
        querys = input("enter the name of city:")
        speak('city is')
        speak(querys)
        for each in response['state_wise']:
            if int(response['state_wise'][each]['active']) != 0:
                for city in response['state_wise'][each]['district']:
                    if city.lower() == querys.lower():
                        active_cases =
response['state_wise'][each]['district'][city]['active']
                        print(f"Total number of active
cases in {querys.lower()} is: {active_cases}")
                        confirmed_cases =
response['state_wise'][each]['district'][city]['confirmed']
                        print(f"Total number of
confirmed cases in {querys.lower()} is: {confirmed_cases}")
                        moveOn()
                continue

    elif 'notepad' in query.lower():
        osCommandString = "notepad.exe"
        os.system(osCommandString)
        moveOn()
        continue

    elif 'mirror' in query.lower():
        speak('showing you mirror sir ,if you want to close it press q for it')
        vid = cv2.VideoCapture(0)

        while(True):
            ret, frame = vid.read()
            cv2.imshow('frame', frame)

```

```

        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
    vid.release()
    cv2.destroyAllWindows()
    moveOn()
    continue

elif 'horoscope' in query.lower():
    day = int(input("enter your birth date sir:"))
    print(day)
    speak(day)
    month = input("enter your birth month sir:")
    print(month)
    speak(month)
    zodiac(day, month)
    moveOn()
    continue

elif 'flight' in query.lower():
    flightCompare()
    moveOn()
    continue

elif 'map' in query.lower():
    from gmplot import *
    gmap = gmplot.GoogleMapPlotter(28.4089, 77.3178, 16)
    gmap.draw( "C:\\Users\\your_path_name\\world_map.html" )
    map = os.startfile("C:\\Users\\your_path_name\\world_map.html")
    moveOn()
    continue

elif 'corona' in query .lower():
    import pycountry
    import pandas as pd
    import plotly.express as px

    urlDataset = r'https://raw.githubusercontent.com/datasets/covid-
19/master/data/countries-aggregated.csv'
    df = pd.read_csv(urlDataset)
    print(df.head)

    list_countries = df['Country'].unique().tolist()
    #print(list_countries)

    d_country_code = { }

```

```

        for country in list_countries:
            try:
                country_data =
pycountry.countries.search_fuzzy(country)
                country_code = country_data[0].alpha_3
                d_country_code.update({country:
country_code})
            except:
                d_country_code.update({country: ' '})
                #print(d_country_code)

        for key, value in d_country_code.items():
            df.loc[(df.Country == key), 'iso_alpha'] = value
            #print(df.head)

        figure = px.choropleth(data_frame = df,
                                locations = "iso_alpha",
                                color = "Confirmed",
                                hover_name = "Country",
                                color_continuous_scale = 'RdYlGn',
                                animation_frame = "Date")
        figure.show()
        moveOn()
        continue

```