```
if __name__ == '__main__':
        clear = lambda: os.system('cls')

        # This Function will clean any
        # command before execution of this python file
        clear()
        wishMe()
        username()

        while True:

                query = takeCommand().lower()

                # All the commands said by user will be
                # stored here in 'query' and will be
                # converted to lower case for easily
                # recognition of command
                if 'wikipedia' in query:
                        speak('Searching Wikipedia...')
                        query = query.replace("wikipedia", "")
                        results = wikipedia.summary(query, sentences = 3)
                        speak("According to Wikipedia")
                        print(results)
                        speak(results)

                elif 'open youtube' in query:
                        speak("Here you go to Youtube\n")
                        webbrowser.open("youtube.com")

                elif 'open google' in query:
                        speak("Here you go to Google\n")
```

```
            webbrowser.open("google.com")

        elif 'open stackoverflow' in query:
            speak("Here you go to Stack Over flow.Happy coding")
            webbrowser.open("stackoverflow.com")

        elif 'play music' in query or "play song" in query:
            speak("Here you go with music")
            # music_dir = "G:\\Song"
            music_dir = "C:\\Users\\GAURAV\\Music"
            songs = os.listdir(music_dir)
            print(songs)
            random = os.startfile(os.path.join(music_dir, songs[1]))

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

        elif 'open opera' in query:
            codePath = 
r"C:\\Users\\GAURAV\\AppData\\Local\\Programs\\Opera\\launcher.exe"
            os.startfile(codePath)

        elif 'email to gaurav' in query:
            try:
                speak("What should I say?")
                content = takeCommand()
                to = "Receiver email address"
                sendEmail(to, content)
                speak("Email has been sent !")
            except Exception as e:
```

```
        print(e)
        speak("I am not able to send this email")

elif 'send a mail' in query:
    try:
        speak("What should I say?")
        content = takeCommand()
        speak("whome should i send")
        to = input()
        sendEmail(to, content)
        speak("Email has been sent !")
    except Exception as e:
        print(e)
        speak("I am not able to send this email")

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 "change my name to" in query:
    query = query.replace("change my name to", "")
    assname = query

elif "change name" in query:
    speak("What would you like to call me, Sir ")
    assname = takeCommand()
    speak("Thanks for naming me")
```

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

elif 'exit' in query:
        speak("Thanks for giving me your time")
        exit()

elif "who made you" in query or "who created you" in query:
        speak("I have been created by Gaurav.")

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

elif "calculate" in query:

        app_id = "Wolframalpha api id"
        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 'search' in query or 'play' in query:

        query = query.replace("search", "")
        query = query.replace("play", "")
        webbrowser.open(query)
        elif "who i am" in query:
            speak("If you talk then definitely your human.")

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

        elif 'power point presentation' in query:
            speak("opening Power Point presentation")
            power = r"C:\\Users\\GAURAV\\Desktop\\Minor
Project\\Presentation\\Voice Assistant.pptx"
            os.startfile(power)

        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 Gaurav")

        elif 'reason for you' in query:
            speak("I was created as a Minor project by Mister Gaurav ")

        elif 'change background' in query:
            ctypes.windll.user32.SystemParametersInfoW(20,

    0,

    "Location of wallpaper",

    0)
            speak("Background changed successfully")
```

```
elif 'open bluestack' in query:
    appli = r"C:\\ProgramData\\BlueStacks\\Client\\Bluestacks.exe"
    os.startfile(appli)

elif 'news' in query:

    try:
        jsonObj = urlopen('''https://newsapi.org / v1 / articles?source = the-times-of-india&sortBy = top&apiKey =\\times of India Api key\\''')
        data = json.load(jsonObj)
        i = 1

        speak('here are some top news from the times of india')
        print('''=============== TIMES OF INDIA ============'''+ '\n')

        for item in data['articles']:

            print(str(i) + '. ' + item['title'] + '\n')
            print(item['description'] + '\n')
            speak(str(i) + '. ' + item['title'] + '\n')
            i += 1
    except Exception as e:

        print(str(e))


elif 'lock window' in query:
        speak("locking the device")
        ctypes.windll.user32.LockWorkStation()

elif 'shutdown system' in query:
```

```
                speak("Hold On a Sec ! Your system is on its way to shut down")
                subprocess.call('shutdown / p /f')

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

        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 "where is" in query:
            query = query.replace("where is", "")
            location = query
            speak("User asked to Locate")
            speak(location)
            webbrowser.open("https://www.google.nl / maps / place/" + location + "")

        elif "camera" in query or "take a photo" in query:
            ec.capture(0, "Jarvis Camera ", "img.jpg")

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

        elif "hibernate" in query or "sleep" in query:
            speak("Hibernating")
            subprocess.call("shutdown / h")
        elif "log off" in query or "sign out" in query:
            speak("Make sure all the application are closed before sign-out")
            time.sleep(5)
            subprocess.call(["shutdown", "/l"])

        elif "write a note" in query:
            speak("What should i write, sir")
            note = takeCommand()
            file = open('jarvis.txt', 'w')
            speak("Sir, Should i include date and time")
            snfm = takeCommand()
            if 'yes' in snfm or 'sure' in snfm:
                strTime = datetime.datetime.now().strftime("% H:% M:% S")
                file.write(strTime)
                file.write(" :- ")
                file.write(note)
            else:
                file.write(note)

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

        elif "update assistant" in query:
            speak("After downloading file please replace this file with the downloaded one")
            url = '# url after uploading file'
            r = requests.get(url, stream = True)
            with open("Voice.py", "wb") as Pypdf:

                total_length = int(r.headers.get('content-length'))

                for ch in progress.bar(r.iter_content(chunk_size = 2391975),
                                        expected_size
=(total_length / 1024) + 1):
                    if ch:
                    Pypdf.write(ch)

        # NPPR9-FWDCX-D2C8J-H872K-2YT43
        elif "jarvis" in query:

            wishMe()
            speak("Jarvis 1 point o in your service Mister")
            speak(assname)

        elif "weather" in query:

            # Google Open weather website
            # to get API of Open weather
            api_key = "Api key"
            base_url = "http://api.openweathermap.org / data / 2.5 / weather?"
            speak(" City name ")
            print("City name : ")
            city_name = takeCommand()
            complete_url = base_url + "appid =" + api_key + "&q =" + city_name
            response = requests.get(complete_url)
            x = response.json()

            if x["cod"] != "404":
                y = x["main"]
                current_temperature = y["temp"]
                current_pressure = y["pressure"]
                current_humidiy = y["humidity"]
                z = x["weather"]
                weather_description = z[0]["description"]
                print(" Temperature (in kelvin unit) = "
+str(current_temperature)+"\n atmospheric pressure (in hPa unit) ="+str(current_pressure) +"\n
humidity (in percentage) = " +str(current_humidiy) +"\n description = " +str(weather_description))

            else:
                speak(" City Not Found ")

        elif "send message " in query:
                # You need to create an account on Twilio to use this service
                account_sid = 'Account Sid key'
                auth_token = 'Auth token'
                client = Client(account_sid, auth_token)

                message = client.messages \
                                .create(
                                    body = takeCommand(),
                                    from_='Sender No',
                                    to ='Receiver No'
                                )

                print(message.sid)

        elif "wikipedia" in query:
            webbrowser.open("wikipedia.com")

        elif "Good Morning" in query:
```

```
    speak("A warm" +query)
    speak("How are you Mister")
    speak(assname)

# most asked question from google Assistant
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 "how are you" in query:
    speak("I'm fine, glad you me that")

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

elif "what is" in query or "who is" in query:

    # Use the same API key
    # that we have generated earlier
    client = wolframalpha.Client("API_ID")
    res = client.query(query)

    try:
        print (next(res.results).text)
        speak (next(res.results).text)
    except StopIteration:
        print ("No results")

# elif "" in query:
    # Command go here
    # For adding more commands
```