

This is a smart PC assistant who can control your entire PC without touching you like- Gmail open,Music play , Google open,etc

```
import pyttsx3 #pip install pyttsx3
import speech_recognition as sr #pip install speechRecognition
import datetime
import wikipedia #pip install wikipedia
import webbrowser
import os
import smtplib
```

```
engine = pyttsx3.init('sapi5')
voices = engine.getProperty('voices')
# print(voices[1].id)
engine.setProperty('voice', voices[0].id)
```

```
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 Jarvis Sir. Please tell me how may I help you")
```

```
def takeCommand():
```

```
    #It takes microphone input from the user and returns string output
```

```
    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 sendEmail(to, content):

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

    server.ehlo()

    server.starttls()

    server.login('youremail@gmail.com', 'your-password')

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

    server.close()

if __name__ == "__main__":

    wishMe()

    while True:

        # if 1:

            query = takeCommand().lower()

            # Logic for executing tasks based on query

            if 'wikipedia' in query:

                speak('Searching Wikipedia...')

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

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

                speak("According to Wikipedia")

                print(results)

```

```
speak(results)
```

```
elif 'open youtube' in query:
```

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

```
elif 'open google' in query:
```

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

```
elif 'open stackoverflow' in query:
```

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

```
elif 'play music' in query:
```

```
music_dir = 'D:\\Non Critical\\songs\\Favorite Songs2'
```

```
songs = os.listdir(music_dir)
```

```
print(songs)
```

```
os.startfile(os.path.join(music_dir, songs[0]))
```

```
elif 'the time' in query:
```

```
strTime = datetime.datetime.now().strftime("%H:%M:%S")
```

```
speak(f"Sir, the time is {strTime}")
```

```
elif 'open code' in query:
```

```
codePath = "C:\\Users\\Haris\\AppData\\Local\\Programs\\Microsoft VS  
Code\\Code.exe"
```

```
os.startfile(codePath)
```

elif 'email to harry' in query:

try:

 speak("What should I say?")

 content = takeCommand()

 to = "harryyourEmail@gmail.com"

 sendEmail(to, content)

 speak("Email has been sent!")

except Exception as e:

 print(e)

 speak("Sorry my friend harry bhai. I am not able to send this email")