'''       Name of students contributed in the project:

                      Rohan Dahibhate (255)

                      Ojas Phapale (255)

'''

#The code might not work in the PC of the checker not matter it is windows or apple as we explained the code to Pranav Sir..

#The reason for the code to not work in your PC will be because of the wrong indexing of the voices in your PC as we explained.

#This will happen because we have downloaded a microsoft voice and have overwritten its entry in the basic system voices by using the windows registry editor and accessing the depths of my system.

#If you want to make the code run in your PC get index as '0' in line '20'. It will give a male voice but can show error as well in some commands as our program is not centred to that voice.

import pyttsx3

import datetime

import speech\_recognition as sr

import wikipedia

import webbrowser

import os

engine = pyttsx3.init('sapi5')

voices = engine.getProperty('voices')

engine.setProperty('voice', voices[2].id) #Here is where you can change the assistant's voice to index '0'..

def speak(audio): #This is defined to avoid code clotting and repetition of the same code multiple times. We can call this entire code just by calling...

    engine.say(audio)

    engine.runAndWait()

def wishme(): #This is a function used to greet the user by using the datetime module which wishes the user according to the time he is in right now..

    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("Hello Everyone... I am Kaaya, a multi-voice assistant designed by Mr Rohan and Mr Ojas")

    voices = engine.getProperty('voices')

    engine.setProperty('voice', voices[1].id)

    speak("And myself Mark.. Whom do you want master to take your command?? Kaaya... Or myself? Please type it...")

    KorM = input("Waiting for response Master...").lower()

    #This is where we can choose who we want as our assistant. Whether it should be Kaya or Mark. It will be dependant on the user and he can type it in the terminal.

    if 'kaya' in KorM:

        voices = engine.getProperty('voices')

        engine.setProperty('voice', voices[2].id)

    elif 'mark' in KorM:

        voices = engine.getProperty('voices')

        engine.setProperty('voice', voices[1].id)

    else:

        speak("Unable to recognize master!!")

        print("Unable to recognize your input...")

def takeCommand(): #This is the function which will take the microphone input from the user and will guide assistants accordingly to return string output...

    r = sr.Recognizer()

    with sr.Microphone() as source:

        print("Listening...")

        r.phrase\_threshold = 1

        audio = r.listen(source)

    try:

        print("Recognizing...")

        global query #query is the global variable now.

        query = r.recognize\_google(audio, language= 'en-in')

        print("User said: ", query)

    except Exception as e: #This is just an extra security layer to make sure code shouldnt suddenly stop if there is some problem.

        print(e)

        print("say that again please...")

        return "None"

    return query

wishme()

speak("Okay Master Thank you for choosing me... What should I do?")

totalBill = 0

while True:

    query = takeCommand().lower()

    #This becomes the logic for search execution...

    if 'wikipedia' in query:

        speak("Going for wikipedia... Master... Just a moment")

        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")

        speak("Going for youtube... master... Just a moment")

    elif 'open google' in query:

        webbrowser.open("google.com")

        speak("Going for google ... master... Just a moment")

    elif 'open chrome' in query:

        pathCrome = "C:\\Program Files (x86)\\Google\\Chrome\\Application\\chrome.exe"

        speak("Opening Chrome Master... Just wait for a moment")

        os.startfile(pathCrome)

    elif 'open Adobe Reader' in query:

        pathReader = "C:\\Program Files (x86)\\Adobe\\Reader 11.0\\Reader\\AcroRd32.exe"

        speak("Opening Reader Master... Just wait for a moment")

        os.startfile(pathReader)

    elif 'the time' in query:

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

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

        print("The time is", strTime)

    elif 'waiter' in query:

        speak("What should I serve for you from the table below")

        print("banana: rupees 5")

        print("pizza: rupees 20")

        print("noodles: rupees 10")

        print("coffee: rupees 5")

        banana = 5

        pizza = 20

        coffee = 5

        noodles = 10

        speak("Waiting for response, master")

        order = print("Waiting for response... Master")

        order1 = takeCommand().lower()

        print('In waiter')

        print(order1)

        if 'banana' in order1:

            print("Your bill is rupees is", banana)

            totalBill = banana

            speak("Your bill is 5")

            print("Your bill is 5")

        if 'pizza' in order1:

            totalBill = totalBill + pizza

            speak("Your bill is 20")

            print("Your bill is 20")

        if 'noodle' in order1:

            totalBill = totalBill + noodles

            speak("Your bill is 10")

            print("Your bill is 10")

        if 'coffee' in order1:

            totalBill = totalBill + coffee

            speak("Your bill is 5")

            print("Your bill is 5")

    elif 'exit' in query:

        speak("Thank You Master for your time")

        exit()

The libraries we installed from pip are as follows:

pyttsx3

datetime

speech\_recognition

wikipedia

webbrowser

os