**SOURCE CODE**

import subprocess

import wolframalpha

import pyttsx3

import tkinter

import json

import random

import operator

import speech\_recognition as sr

import datetime

import wikipedia

import webbrowser

import os

import winshell

import pyjokes

import feedparser

import smtplib

import ctypes

import time

import requests

import shutil

from twilio.rest import Client

from clint.textui import progress

from bs4 import BeautifulSoup

import win32com.client as wincl

from urllib.request import urlopen

engine = pyttsx3.init('sapi5')

voices = engine.getProperty('voices')

engine.setProperty('voice', voices[1].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 Sir !")

elif hour>= 12 and hour<18:

speak("Good Afternoon Sir !")

else:

speak("Good Evening Sir !")

assname =("Jarvis 1 point o")

speak("I am your Assistant")

speak(assname)

def usrname():

speak("What should i call you sir")

uname = takeCommand()

speak("Welcome Mister")

speak(uname)

columns = shutil.get\_terminal\_size().columns

print("#####################".center(columns))

print("Welcome Mr.", uname.center(columns))

print("#####################".center(columns))

speak("How can i Help you, Sir")

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, language ='en-in')

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

except Exception as e:

print(e)

print("Unable to Recognize your voice.")

return "None"

return query

def sendEmail(to, content):

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

server.ehlo()

server.starttls()

# Enable low security in gmail

server.login('your email id', 'your email passowrd')

server.sendmail('your email id', to, content)

server.close()

if \_\_name\_\_ == '\_\_main\_\_':

clear = lambda: os.system('cls')

# This Function will clean any

# command before execution of this python file

clear()

wishMe()

usrname()

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 'the time' in query:

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

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

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 A group of three people")

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 definately your human.")

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 "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 "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)

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 "wikipedia" in query:

webbrowser.open("wikipedia.com")

elif "Good Morning" in query:

speak("A warm" +query)

speak("How are you Mister")

speak(assname)

elif "how are you" in query:

speak("I'm fine, glad you me that")

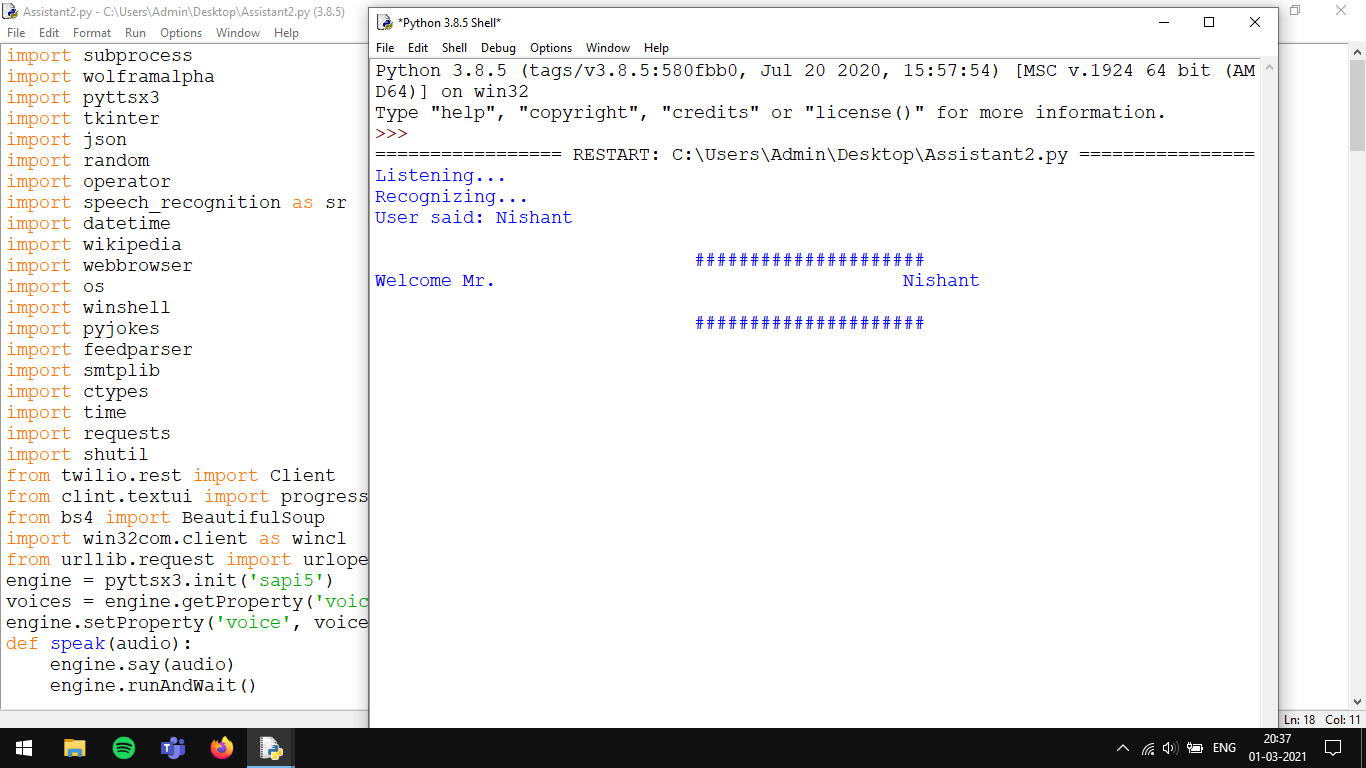
elif "i love you" in query:

speak("It's hard to understand")

**(Page Number)**

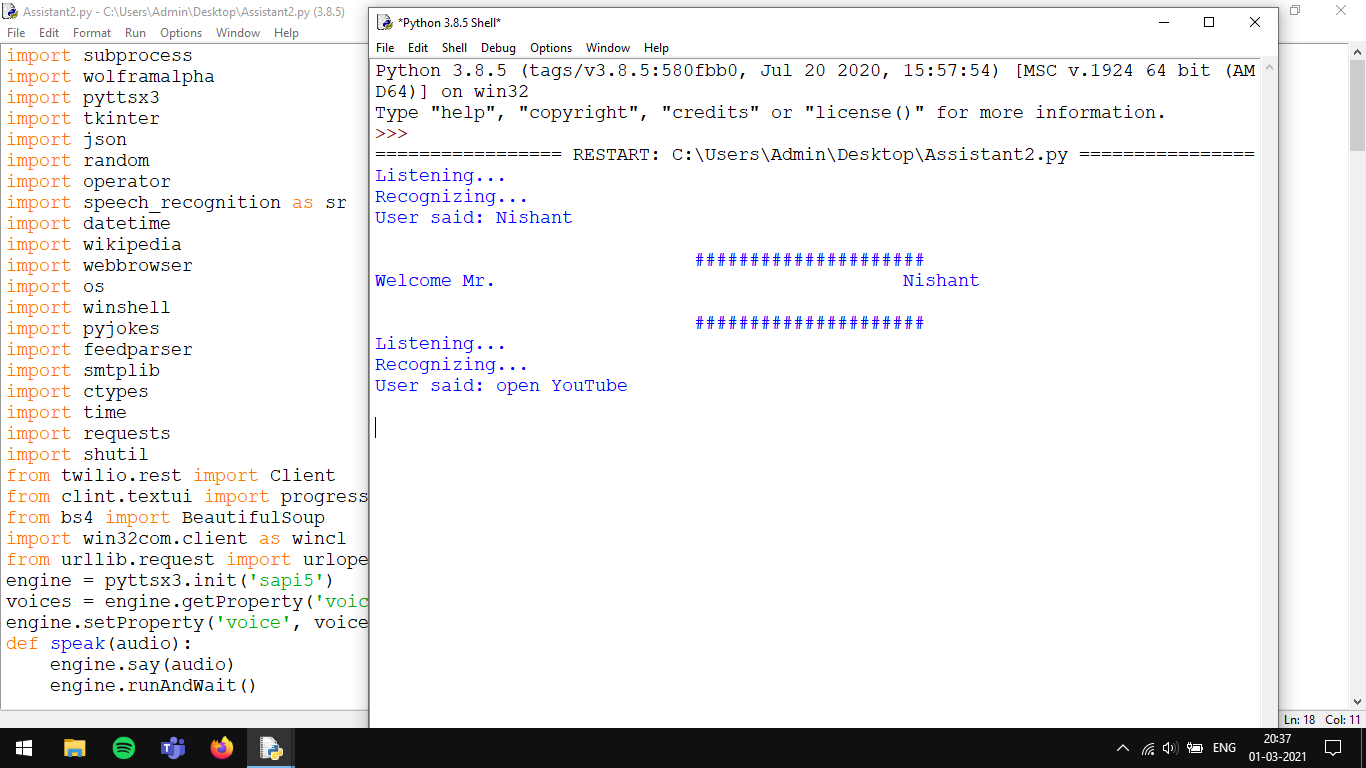
**OUTPUT**

1. **Execution Processes.**



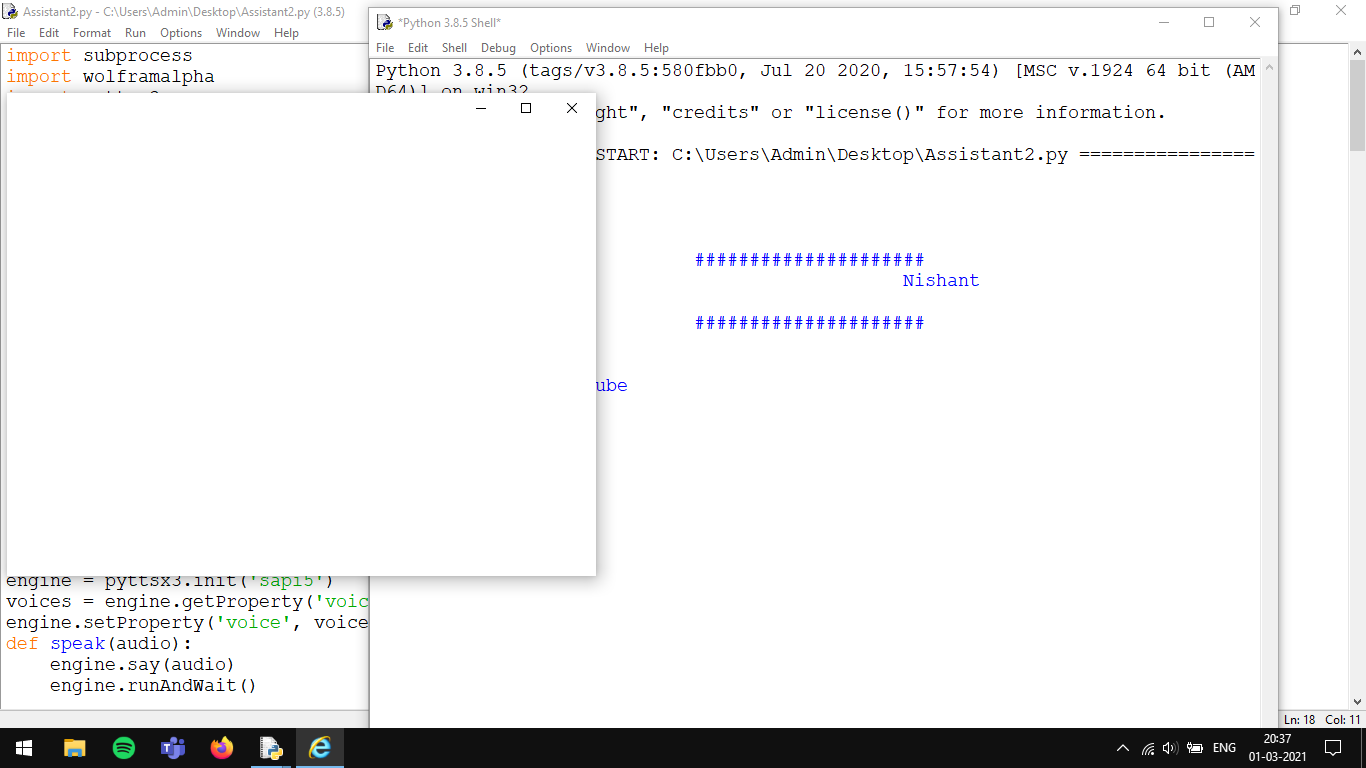
User name spoken: Nishant

1. **Command Given.**

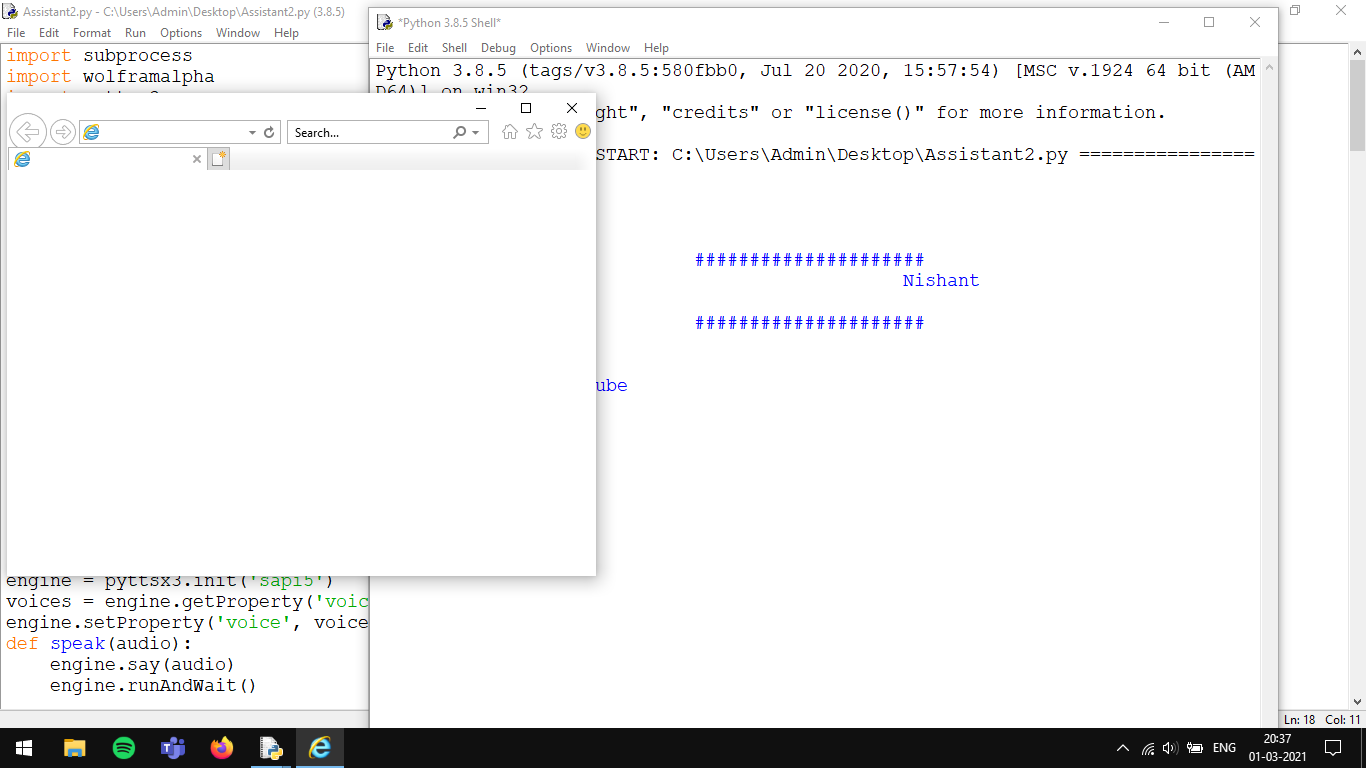


**(Page Number)**

1. **Processing Starts.**



1. **Process being transferred to another Browser.**



Reason: As Internet Explorer do not supports new websites

**(Page Number)**

1. **Execution completed successfully.**

