# VOICE ASSISTANT

**A Project Work**

*Submitted in the partial fulfillment for the award of the degree of*

**BACHELOR OF TECHNOLOGY**

**IN**

**DATA SCIENCE AND ARTIFICIAL**

**INTELLIGENCE**

**Submitted by:**

**K.PAVAN KALYAN(19\_251)**

**T.RUDRA DE(19\_348)**

**Under the Supervision of :**

**Dr.BRAHMA NAIDU**



**MARCH ,2022**

# DECLARATION

We, K.PAVAN KALYAN, T.RUDRA DEV students of **BACHELOR OF TECHNOLOGY**, Data science and Artificial Intelligence (2019-2023), of IFHE HYDERABAD hereby declare that the work presented in this Project Work entitled ‘VOICE ASSISTANT**’** is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. It contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

**Date :-APRIL 28, 2022**

# Table of Contents

[VOICE ASSISTANT 1](#_Toc13636)

[DECLARATION 2](#_Toc24316)

[Table of Contents 3](#_Toc31174)

ABSTRACT……………………………………………………………………………5

1.Introduction…………………………………………………………………………..6

* 1. Present System…………………………………………………………………..7
  2. Proposed System……………………………………………………………...…7
  3. Overview………………………………………………………………………..8

2.Architecture of Voice Assistant……………………………………………………..9

3.Software Details ……………………………………………………………………10

3.1 Software Requirements………………………………………………………..12

3.2 Modules Required ……..………………………………………………………12

3.3 Libraries Required……………………………………………………………..15

4.Literature Review……………………………………………………………….….16

5.How the voice assistant work?………………………………………………….….24

6. Implementation of work Details……………………………………………….…..25

6.1 Real Life Application………………………………………………………….25

6.2 Data Implementation and Program Execution………………………………...25

6.2.1 Libraries and Packages………………………………………………….26

6.2.2 Functions……………………………………………………………….26

7.Source Code for voice Assistant……………………………………………………27

8.Input/Output Screenshot……………………………………………………………42

8.1 How to install modules in pycharm ………………………………………….50

9. System Testing……………………………………………………………………..52

9.1 Functionality………………………………………………………………….52

9.2 Usability ……………………………………………………………………..53

9.3 Security……………………………………………………………………….54

9.4 Stability……………………………………………………………………….54

10. Overall Contribution …………………………………………………………..55

11.Conclusion………………………………………………………………………56

11.1Limitations…………………………………………………………………..56

11.2 Scope for future Work………………………………………………………56

12. References ……………………………………………………………………………………………………..57

**ABSTRACT**

Voice assistants are software agents that can interpret human speech and respond via synthesized voices. Apple’s Siri, Amazon’s Alexa, Microsoft’s Cortana, and Google’s Assistant are the most popular voice assistants and are embedded in smartphones or dedicated home speakers. Users can ask their assistants questions, control home automation devices and media playback via voice, and manage other basic tasks such as

email, to-do lists, and calendars with verbal commands. This column will explore the basic workings and common features of today’s voice assistants. It will also discuss some of the privacy and security issues inherent to voice assistants and some

potential future uses for these devices. As voice assistants become more widely used, librarians will want to be familiar with their operation and perhaps consider them as a means to deliver library services and materials.Particularly, we focus on developing skills (or apps) for the Alexa assistant, as it is the most widespread. It is mainly focus on the time complexity ,because voice assistant avoids the typing and it can do multi tasks at a time.Now a days technology improves and every person uses smart phone ,so we are doing this project and it has a lot of features.

1. **INTRODUCTION**

Artificial Intelligence when used with machines, it shows us the capability of

thinking like humans. In this, a computer system is designed in such a way that typically requires interaction from human. As we know Python is an emerging language so it becomes easy to write a script for Voice Assistant in Python. The instructions for the assistant can be handled as per the requirement of user. Speech recognition is the Alexa, Siri, etc. In Python there is an API called Speech Recognition which allows us to convert speech into text. It was an interesting task to make my own assistant. It became easier to send emails without typing any word, Searching on Google without opening the browser, and performing many other daily tasks like playing music, opening your favorite IDE with the help of a single voice command. In the current scenario,

advancement in technologies are such that they can perform any task with same

effectiveness or can say more effectively than us. By making this project, I realized that the concept of AI in every field is decreasing human effort and saving time. As the voice assistant is using Artificial Intelligence hence the result that it is providing are highly accurate and efficient. The assistant can help to reduce human effort and consumes time while performing any task, they removed the concept of typing completely and behave as another individual to whom we are talking and asking to perform task. The assistant is no less than a human assistant but we can say that this is more effective and efficient to perform any task. The libraries and packages used to make this assistant focuses on the time complexities and reduces time. The functionalities include , It can send emails, It can read PDF, It can send text on WhatsApp, It can open command prompt, your favorite IDE, notepad etc., It can play music, It can do Wikipedia searches for you, It can open websites like Google, YouTube, etc., in a web browser, It can give weather forecast, It can give desktop reminders of your choice. It can have some basic conversation. Tools and technologies used are PyCharm IDE for making this project, and I created all py files in PyCharm. Along with this I used following modules and libraries

in my project. pyttsx3, SpeechRecognition, Datetime, Wikipedia, Smtplib, pyjokes, etc.

**1.1 PRESENT SYSTEM**

We are familiar with many existing voice assistants like Alexa, Siri, Google

Assistant, Cortana which uses concept of language processing, and voice recognition. They listens the command given by the user as per their requirements and performs that specific function in a very efficient and effective manner. As these voice assistants are using Artificial Intelligence hence the result that they are providing are highly accurate and efficient. These assistants can help to reduce human effort and consumes time while performing any task, they removed the concept of typing completely and behave as another individual to whom we are talking and asking to perform task. These assistants are no less than a human assistant but we can say that they are more effective and efficient to perform any task. The algorithm used to make these assistant focuses on the time complexities and reduces time. But for using these assistants one should have an account (like Google account for Google assistant, Microsoft account for Cortana) and can use it with internet connection only because these assistants are going to work with internet connectivity. They are integrated with many devices like, phones, laptops, and speakers etc.

**1.2 PROPOSED SYSTEM**

It was an interesting task to make my own assistant. It became easier to send

emails without typing any word, Searching on Google without opening the browser, and performing many other daily tasks like playing music, opening your favorite IDE with the help of a single voice command. Jarvis is different from other traditional voice assistants in terms that it is specific to desktop and user does not need to make account to use this, it does not require any internet connection while getting the instructions to perform any specific task.

The IDE used in this project is PyCharm. All the python files were created in

PyCharm and all the necessary packages were easily installable in this IDE. For this project following modules and libraries were used i.e. pyttsx3, SpeechRecognition, Datetime, Wikipedia, Smtplib, pyjokes, etc

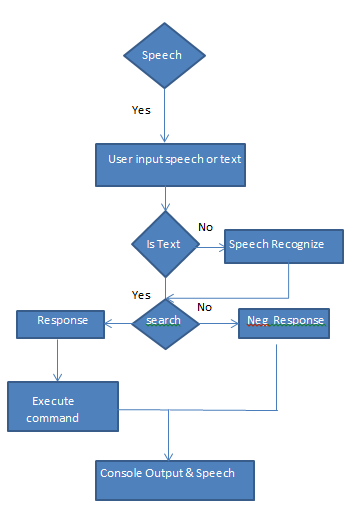
With the advancement Alexa can perform any task with same effectiveness or can say more effectively than us. By making this project, I realized that the concept of AI in every field is decreasing human effort and saving time. Functionalities of this project include, It can send emails, It can read PDF, It can send text on WhatsApp, It can open command prompt, your favorite IDE, notepad etc., It can play music, It can do Wikipedia searches for you, It can open websites like Google, YouTube, etc., in a web browser, It can give weather forecast, It can give desktop reminders of your choice. It can have some basic conversation

**1.3 Overview**

According to the overall description in the context, the purpose of the project is to develop an Android application that provides an intelligent voice assistant with the functionalities as calling services, message transformation, mail exchange, alarm, event handler, location services, music play service, checking weather, searching engine (Google, Wikipedia), camera, Bluetooth headset support.

1. **ARCHITECTURE OF VOICE ASSISTANT:**





1. **Software details**

The IDE used in this project is PyCharm. All the python files were created in

PyCharm and all the necessary packages were easily installable in this IDE. For this

project following modules and libraries were used i.e. pyttsx3, SpeechRecognition,

Datetime, Wikipedia, Smtplib, pyjokes, etc.

3.1. PYCHARM

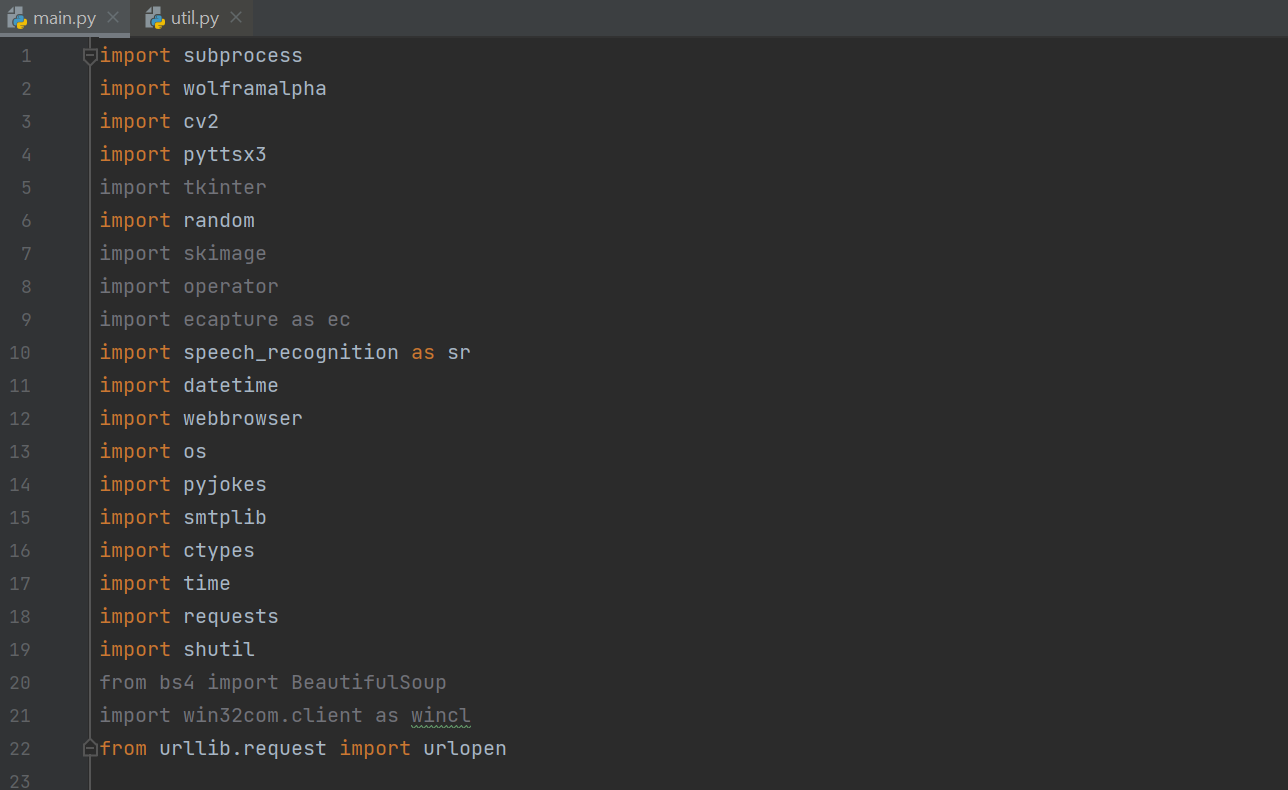
It is an IDE i.e. Integrated Development Environment which has many features

like it supports scientific tools(like matplotlib, numpy, scipy etc) web frameworks

(example Django,web2py and Flask) refactoring in Python, integrated python

debugger, code completion, code and project navigation etc. It also provides Data

Science when used with Anaconda.



**3.1 Software Requirements**

Python 3 language

Machine Learning

Artificial Intelligence

**3.2 Modules Required**

* Pyttsx3:

This module is used for the conversion of text to speech in a program it works offline.

Speech recognition

* Wikipedia:

we have used the Wikipedia module to get information from Wikipedia or to perform a Wikipedia search.

* SpeechRecognition :

Since we’re building an Application of voice assistant, one of the most important things in this is that your assistant recognizes your voice (means what you want to say/ ask).

* Sub process:

This module is used for getting system subprocess details which are used in various commands i.e Shutdown, Sleep, etc. This module comes built-in with Python.

* WolframAlpha:

It is used to compute expert-level answers using Wolfram’s algorithms, knowledgebase and AI technology. To install this module type the below command in the terminal.

* Tkinter :

This module is used for building GUI and comes inbuilt with Python. This module comes built-in with Python.

* Web browser:

To perform Web Search. This module comes built-in with Python.

* Ecapture **:**

To capture images from your Camera

* Pyjokes :

Pyjokes is used for collection Python Jokes over the Internet.

* Date Time :

Date and Time is used to showing Date and Time.

* Twilio :

Twilio is used for making call and messages.

* Beautiful Soap:

Beautiful Soup is a library that makes it easy to scrape information from web pages.

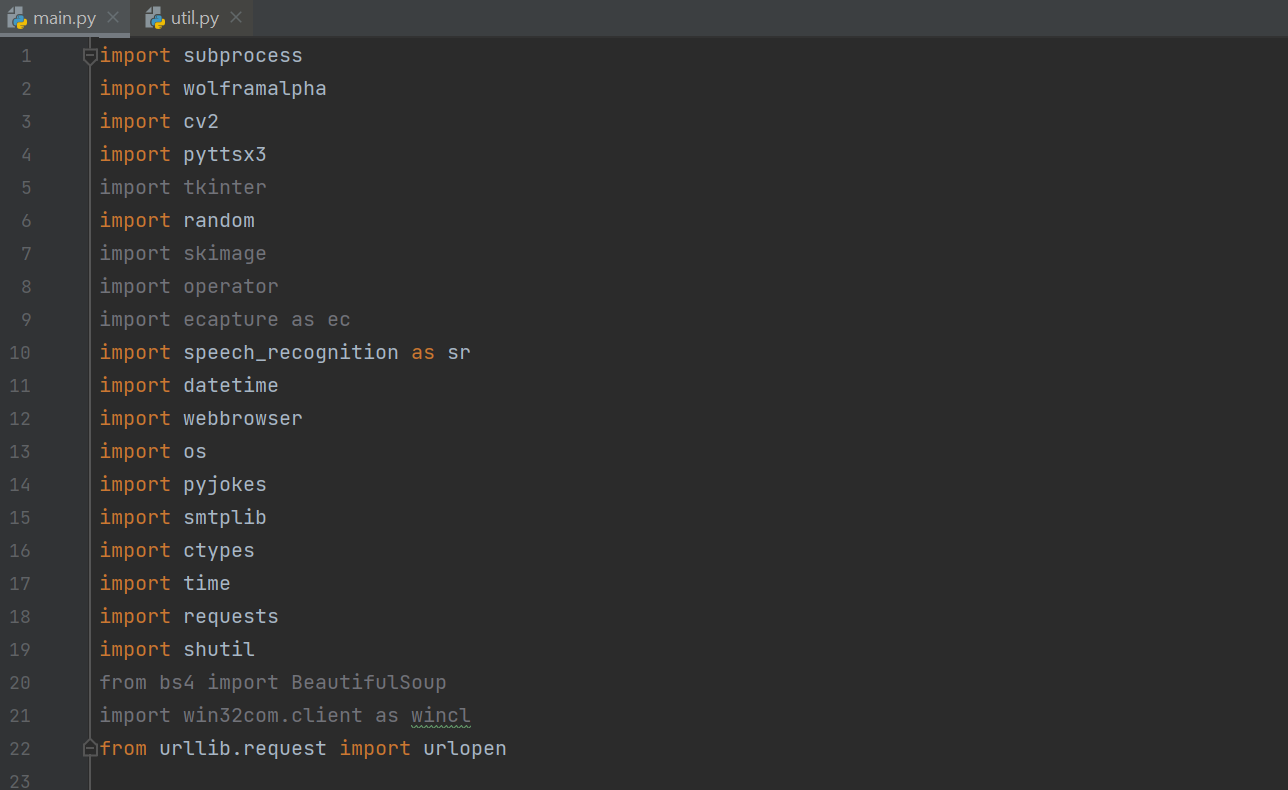
* Requests:

Requests is used for making GET and POST requests.



**Basic figure of voice assistant**

**3.3 Libraries Required** :



**Figure shows the libraries we used in this project**

* import json: The JSON module is mainly used to convert the python dictionary above into a JSON string that can be written into a file. While the JSON module will convert strings to Python datatypes, normally the JSON functions are used to read and write directly from JSON files.
* import random: import random. #Generates a random number between a given positive range
* import operator :Operators are special symbols in Python that carry out arithmetic or logical computation. The value that the operator operates on is called the operand. For example: >>> 2+3 5. Here, + is the operator that performs addition.
* import os : The OS module in Python provides functions for creating and removing a directory (folder), fetching its contents, changing and identifying the current directory, etc. You first need to import the os module to interact with the underlying operating system.
* import winshell:The winshell module is a light wrapper around the Windows shell functionality. It includes convenience functions for accessing special folders, for using the shell's file copy, rename & delete functionality, and a certain amount of support for structured storage.

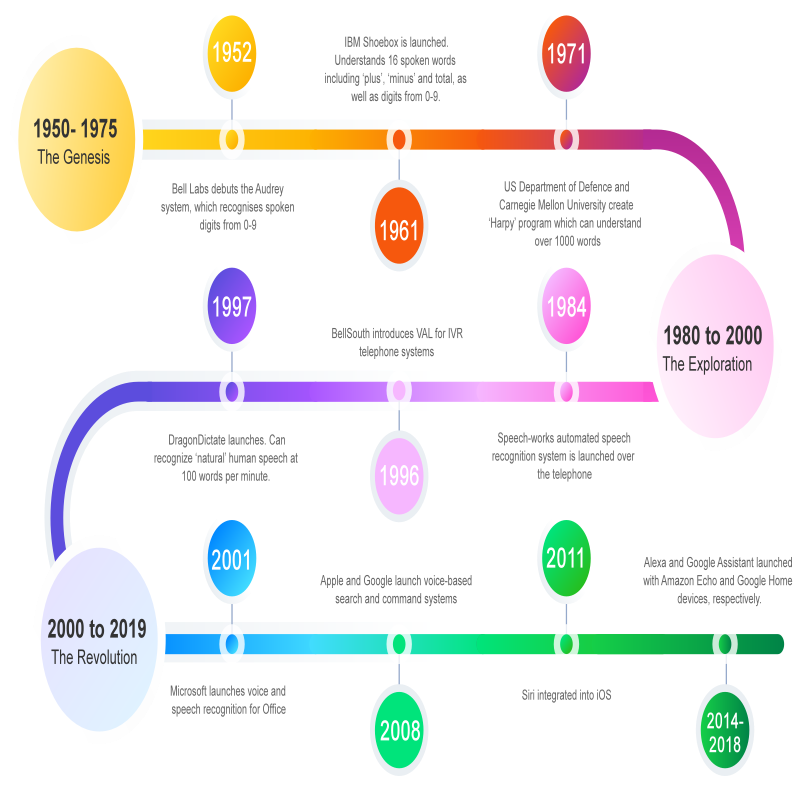
1. **Literature Review**

In this project we present a speech recognition to convert the voice into text and it will give the output .It is the ability given to machines and computers to interpret the voice of humans and then take appropriate action. In this project we use a lot of modules . For each module there is a specific task example : if we import a Wikipedia library then when you ask the voice assistant about any thing at that time it will automatically work. We need a keyword to interact with voice assistant like ‘ok google’, whenever you use the keyword then the voice assistant activates and it will respond to your query.

Speech recognition starts by taking the sound energy produced by the person speaking and converting it into electrical energy with the help of a microphone. It then converts this electrical energy from analog to digital, and finally to text. It breaks the audio data down into sounds, and it analyzes the sounds using algorithms to find the most probable word that fits that audio. All of this is done using [Natural Language Processing](https://www.simplilearn.com/tutorials/artificial-intelligence-tutorial/what-is-natural-language-processing-nlp" \o "Natural Language Processing" \t "https://www.simplilearn.com/tutorials/python-tutorial/_blank) and [Neural Networks](https://www.simplilearn.com/tutorials/deep-learning-tutorial/neural-network" \o "Neural Networks" \t "https://www.simplilearn.com/tutorials/python-tutorial/_blank). Hidden Markov models can be used to find temporal patterns in speech and improve accuracy.

**4.1 Literature Survey** :

* Dr.Kshama V. Kulhalli (2017) proposed the Most famous application of iPhone is “SIRI” which helps the end user to communicate end user mobile with voice and it also responds to the voice commands of the user. Same kind of application is also developed by the Google that is “Google Voice Search” which is used for in Android Phones. But this Application mostly works with Internet Connections. But our Proposed System has capability to work with and without Internet Connectivity. It is named as Personal Assistant with Voice Recognition Intelligence, which takes the user input in form of voice or text and process it and returns the output in various forms like action to be performed or the search result is dictated to the end user. In addition, this proposed system can change the way of interactions between end user and the mobile devices. The system is being designed in such a way that all the services provided by the mobile devices are accessible by the end user on the user's voice commands.
* Kishore Kumar R1 in 2018 presented to develop an economically effective and performance wise efficient virtual assistant using Raspberry Pi for home automation based on the concepts of Internet of Things, Speech Recognition, Natural Language Processing and Artificial Intelligence. People who are using it can give voice inputs and the device itself responds through voice commands by itself. It can fetch the date, time, weather, play your favourite music and fetch search results from the internet along with controlling the home appliances. Node MCU chips are used to control the appliances which receives the command from the Raspberry Pi. The Raspberry Pi processes the speech inputs online given by the user through the mic and converts it into text and executes the command. The whole project is put in action through a python script which includes online Speech to Text conversion and Text to Speech conversion codes written. The Node MCU is coded separately using the Arduino IDE to make it control the appliances and allow it to be accessed through its IP address. The device will respond to the user in a casual manner so that the user has a friendly experience with the device and feels it like his or her own assistant. This device makes the day by day processes easier.
* Rutuja V. Kukade in 2018 proposed there are various communication barriers for people who are blind , and they have to face various challenges. In this paper, we have discussed the implementation of a personal virtual assistant which can take the human voice commands to perform tasks which otherwise would need the dependence on others. It enables user to receive and send emails, know the weather forecast report, maintain a personal diary/Online Blog, recognize image etc, using Speech to Text Engine, Text to speech Engine, OCR (Optical character recognition) using microphone for the input and speakers for the output. International Research Journal of Engineering and Technology .
* VetonKëpuska (2018) proposed one of the goals of Artificial intelligence (AI) is the realization of natural dialogue between humans and machines. in recent years, the dialogue systems, also known as interactive conversational systems are the fastest growing area in AI. Many companies have used the dialogue systems technology to establish various kinds of Virtual Personal Assistants based on their applications and areas, such as Microsoft’s Cortana, Apple’s Siri, Amazon Alexa, Google Assistant, and Facebook’s M. However, in this proposal, we have used the multi-modal dialogue systems which process two or more combined user input modes, such as speech, image, video, touch, manual gestures, gaze, and head and body movement in order to design the Next Generation of VPA model. The new model of VPA will be used to increase the interaction between humans and the machines by using different technologies, such as gesture recognition, image/video recognition, speech recognition, the vast dialogue and conversational knowledge base, and the general knowledge base. Moreover, the new VPA system can be used in other different areas of applications, including education assistance, medical assistance, robotics and vehicles, disabilities systems, home automation, and security access control. In this proposal, we have tested the new VPA model by using IBM Watson cloud server with Python, Node Red.
* Deny Nancy (2019) presented in the Modern Era of fast moving technology we can do things which we never thought we could do before but, to achieve and accomplish these thought s there is a need for a plat form which can automate all our tasks with ease and comfort. Thus we humans developed applications like Personal Voice Assistant having the ability to inter act with the surroundings just by one of the materialistic form of human inter action i.e .Human Voice. The most famous application of android mobile phone is “Google Assistant”, “Google Voice Search” which is developed by the Google .Various applications like Microsoft Cortana, Amazon Alexa is also used as an voice assistant .The voice application of iphone is“SIRI”which helps the end user to communicate end-user mobile with voice and it also responds to the voice commands of the user. We are going to develop a web application were the voice assistant would be available for a particular website. In this proposed system we have took a college website as an example. It can change the way of inter actions between end user and the website. The Application is being designed in such a way that all the services provided by the website are accessible by the end user on the user's voice commands.
* Deepak Shende (2019) presented artificial intelligence technologies are beginning to be actively used in human life, this is facilitated by the appearance and wide dissemination of the Internet of Things (IOT). Autonomous devices are becoming smarter in their way to interact with both a human and themselves. New capacities lead to creation of various systems for integration of smart things into Social Networks of the Internet of Things. One of the relevant trends in artificial intelligence is the technology of recognizing the natural language of a human. New insights in this topic can lead to new means of natural human machine interaction, in which the machine would learn how to understand human’s language, adjusting and interacting in it. One of such tools is voice assistant, which can be integrated into many other intelligent systems. In this paper, the principles of the functioning of voice assistants are described, its main shortcomings and limitations are given. The method of creating a local voice assistant without using cloud services is described, which allows to significantly expand the applicability of such devices in the future.
* Isha S. Dubey (2019) proposed about a different combination of a reading machine (OCR), virtual assistant and domotics system using Raspberry-Pi which will be a combination of a great system. This is a helpful aid for visually impaired people and people with disabilities. OCR stands for optical character recognition where it recognizes the present text and converts them into audio speech using pre and post processing with gTTS (Google Text to Speech). Google is used as its platform for virtual assistant which can be used in day to day life activities like checking mails, weather-forecast, news etc., further using Google Assistant, and python language we implement a voice based home automation. The major objective of this project is to help visually impaired by using various fields of technology. By just voice commands tasks such as reading of document, home automation and personal assistant can be achieved.
* M. A. Jawale (2019) proposed in today’s world, many artificial intelligence applications developed using programming languages like Python, R and so on. Each language comes with its own programming structure and syntactical forms. Programmers are broadly classified into three categories namely, novice users, knowledge intermittent and expert one. For novice users, it is always a challenge to write a code without typographic errors though users know theoretical knowledge of Programming language, its structure and syntax as well as logic of program. Therefore, this paper explores use of voice recognition technique in the field of programming, specifically for writing program with Python programming language. In experimental analysis, it found helpful for new Python programmers and provide new learning curve for programmers wherein beginner can experience hassle free program writing. This paper adds new way of creating interest in beginners for judging their coding paradigm understanding and explore one of the area for user experience field for better programming Integrated Development Environment Development (IDE).
* Tushar Gharge (2019) presented the problem of user while developing a computer program. Developing a International Research Journal of Engineering and Technology .computer program is not an easy task it needs hardware resources which user have to handle. While continuous typing the code there may be possibility of injuries to the fingers of the user. To avoid the problems we are designing a system in which the computer program can developed through the voice. The voice will recognized by the system and that recognized words or word will be compared with the stored keywords in the database and if they are matched then that will be printed on editor and after this again by recognizing the specific keywords the program will be compiled and executed. This system will be easy to use, it reduce human efforts and the use of hardware resources. It would be surely useful for blind as well as novice plus knowledge intermittent users.



* AbhayDekate (2016) et al. presented in the Modern Era of fast moving technologywe can do things which we never thought we could do before but, to achieve and accomplish these thoughts there is a need for a platform which can automate all our tasks with ease and comfort. Thus we need to develop a Personal Assistant having brilliant powers of deduction and the ability to interact with the surroundings just by one of the materialistic form of human interaction i.e. Human Voice. The Hardware device captures the audio request through microphone and processes the request so that the device can respond to the individual using in-built speaker module. For Example, if you ask the device ’what’s the weather?’ or ’how’s traffic?’ using its built-in skills, it looks up the weather and traffic status respectively and then returns the response to the customer through connected speaker.
* Voice assistant has a long history with several waves of major innovations. Voice assistant for dictation, search, and voice commands has become a standard feature on smartphones and wearable devices. The study stems from an overlooking literature review in order to present generic knowledge (theory and concepts) about voice control, virtual assistants, fields of use and more. When looking at a number of currently available intelligent programs with natural language processing capabilities, many examples can be found in everyday life filling a variety of roles. The first speech recognition system, named Audrey, was created by Bell Laboratories in 1952. Audrey was rather rudi mental and limited technology wise, understanding only ten digits - spoken by particular people . About 10 years later, IBM developed and demonstrated their Shoebox Machine. The device recognized and responded to 16 different spoken words, including all ten digits “0” to “9” as well as calculating commands such as “plus” or “minus” (IBM, 2018).Shoebox Machine recognized and responded to 16 spoken words, including the ten digits from “0” through “9”, only in English by a designated speaker. These limitations later proved to be problematic, increasing the scepticism opposing voice recognition. Mid 1970’s came the Hidden Markov Model (HMM) . The HMM considerably altered the development of a feasible speech recognition software. With the help of HMM speech recognition started using a statistical method measuring the probability of unknown sounds being words. Now, the potential to recognize an unlimited number of words became imminent due to the method allowing the number of understandable words go up to a few thousands. These choices of observation distribution in each state of the model allow accurate modelling of virtually unlimited types of data. The first mass accessible voice command system was launched by Apple Inc. as they released the virtual assistant named Siri in 2011 .The intelligent bot Siri can be found as standard on Apple mobile devices now and is considered a core component on these devices. Siri is a personal assistant that uses natural language processing to answer questions and outsource requests to web services that will then be carried out for the user. Similarly to this, the chatbot HAL was created by Zabaware Inc to function as a virtual assistant for users on computers. The bot also uses natural language processing algorithms to converse with the user and take notes from what the user is saying in an effort to organize the data given to it. IBM has invested a large amount of resources into this field and has created Watson, a system developed to compete on the TV show Jeopardy!. This system exhibits the current capabilities of intelligent systems with natural language recognition as it successfully beat the two most successful human contestants of the show. In contrast to these roles is the chatbot Kari who functions as a virtual girlfriend. This system communicates with the user and through using similar methods natural language recognition tries to provoke social conversation with the user. The software aims to give personal companionship and to replicate human interaction as accurately as possible with the assistance of algorithms designed to help the program learn from its inputs.

**5.How the voice assistant work?**

When a voice assistant receives some audio input,before it give output there are several steps to follow . The steps are the following

The voice assistant records the speech, removing any background noise and splitting the speech into its component parts.The speech is transcribed into what the assistant thinks is the most probable sentence, based on the patterns of these sounds.

The assistant determines the most important words and most likely intent of the text.

If it requires more clarification, such as the most relevant words that the assistant might understand easily. Alternatively, it may decide that it is confident enough in its answer to proceed without further information.

Finally, the commands are processed in a decision engine and the query you asked is shown.Many of these tasks rely heavily on natural language processing (NLP). This is a field of computer science that is broadly concerned with a machine’s ability to recognize what is said to it, understand its meaning and the appropriate action, and respond in language that the user will understand.Within NLP, there are two sub fields that particularly concern voice assistants: natural language understanding (NLU) and natural language generation (NLG). Unsurprisingly, NLU focuses primarily on the ability of machines to detect, comprehend and attribute a meaning to speech or text input, whereas NLG deals with the reverse process: turning computer generated responses into text or speech that a human can understand.

**6.Implementation Work Details**

Alexa , a desktop assistant is a voice assistant that can perform many daily tasks of

desktop like playing music, opening your favorite IDE with the help of a single voice command. Alexa is different from other traditional voice assistants in terms that it is specific to desktop and user does not need to make account to use this, it does not require any internet connection while getting the instructions to perform any specific task.

**6.1. REAL LIFE APPLICATION**

6.1.1. Saves time: Alexa is a desktop voice assistant which works on the voice

command offered to it, it can do voice searching, voice-activated device control and

can let us complete a set of tasks.

6.1.2. Conversational interaction It makes it easier to complete any task as it

automatically do it by using the essential module or libraries of Python, in a

conversational interaction way. Hence any user when instruct any task to it, they feel like giving task to a human assistant because of the conversational interaction for giving input and getting the desired output in the form of task done.

6.1.3. Reactive nature: The desktop assistant is reactive which means it know human language very well and understand the context that is provided by the user and gives response in the same way, i.e. human understandable language, English. So user finds its reaction in an informed and smart way.

6.1.4. Multitasking: The main application of it can be its multitasking ability. It can

ask for continuous instruction one after other until the user “QUIT” it.

6.1.5. No Trigger phase: It asks for the instruction and listen the response that is

given by user without needing any trigger phase and then only executes the task.

**6.2. DATA IMPLEMENTATION AND PROGRAM EXECUTION**

As the first step, install all the necessary packages and libraries. The command used

to install the libraries is “pip install” and then import it. The necessary packages included are as follows:

**6.2.1. LIBRARIES AND PACKAGES**

6.2.2.1. pyttsx3: It is a python library which converts text to speech.

6.2.2.2. SpeechRecognition: It is a python module which converts speech to text.

6.2.2.3. pywhatkit: It is python library to send WhatsApp message at a particular time with some additional features.

6.2.2.4. Datetime: This library provides us the actual date and time.

6.2.2.5. Wikipedia: It is a python module for searching anything on Wikipedia.

6.2.2.6. Smtplib: Simple mail transfer protocol that allows us to send mails and to route mails between mail servers.

6.2.2.7. Pyjokes: It is a python libraries which contains lots of interesting

jokes in it.

6.2.2.8. Webbrowser: It provides interface for displaying web-based documents to users.

6.2.2.9. os: It represents Operating System related functionality.

**6.2.2. FUNCTIONS**

* takeCommand(): The function is used to take the command as input

through microphone of user and returns the output as string.

* wishMe(): This function greets the user according to the time like

Good Morning, Good Afternoon and Good Evening.

* taskExecution(): This is the function which contains all the

necessary task execution definition like sendEmail(), pdf\_reader(), news()

and many conditions in if condition like “open google”, “open notepad”,

“search on Wikipedia” ,”play music” and “open command prompt” etc.

**7.Source code for voice assistant**

We can use pycharm to write code for voice assistant.

Firstly we need to install some libraries in terminal that are

* pip install pyttsx3
* pip install wikipedia
* pip install SpeechRecognition

And now we write a code

import subprocess

import wolframalpha

import pyttsx3

import tkinter

import operator

import speech\_recognition as sr

import datetime

import wikipedia

import webbrowser

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 pavan !")

elif hour>= 12 and hour<18:

speak("Good Afternoon pavan !")

else:

speak("Good Evening pavan!")

assname =("Alexa")

speak("I am your Assistant")

speak(assname)

def username():

speak("What should i call you pavan")

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, pavan")

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 password')

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()

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\\pavan kalyan\\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\\pavan\\AppData\\Local\\Programs\\Opera\\launcher.exe"

os.startfile(codePath)

elif 'email to kalyan' 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 Rudra.")

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 Rudra. further It's a secret")

elif 'power point presentation' in query:

speak("opening Power Point presentation")

power = r"C:\\Users\\Pavan\\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 Pavan")

elif 'reason for you' in query:

speak("I was created as a Minor project by Mister Pavan ")

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 "Alexa" in query:

wishMe()

speak("Alexa 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

**8: Input/Output Screenshot :**

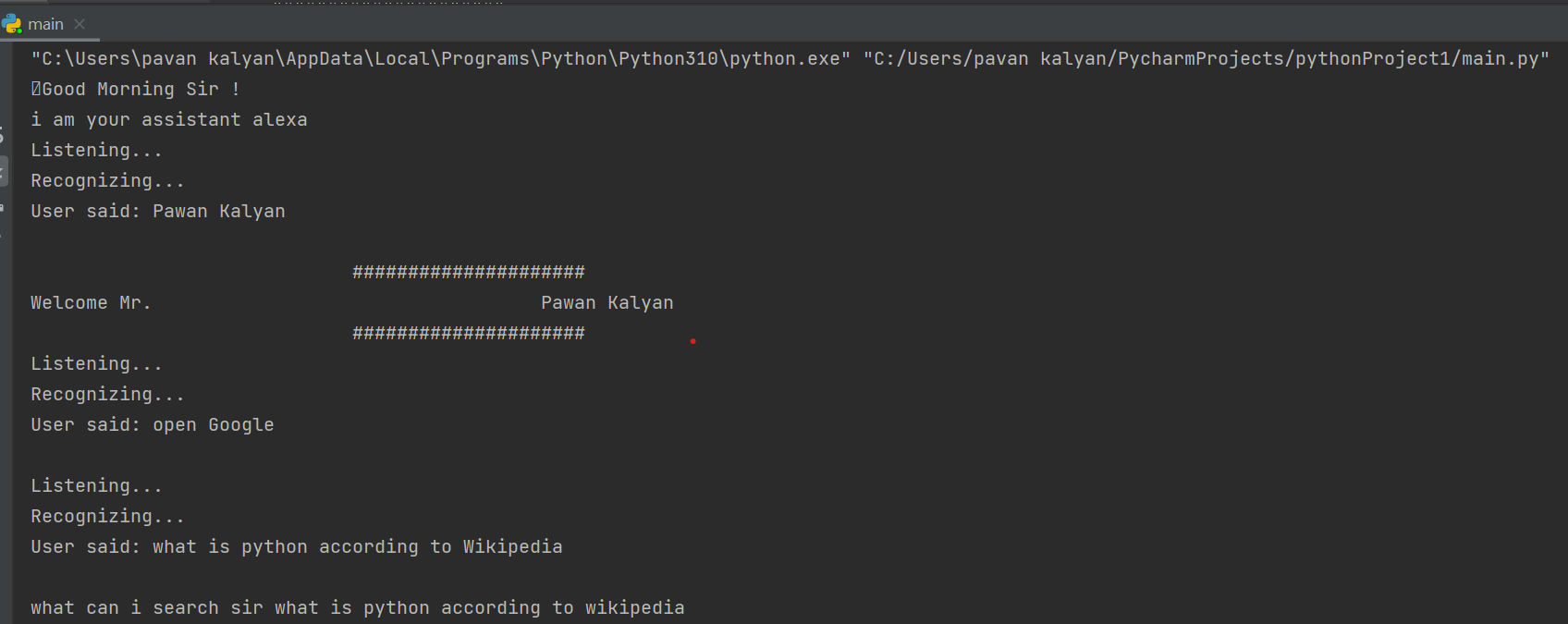


Figure 8.1 Input for Google search

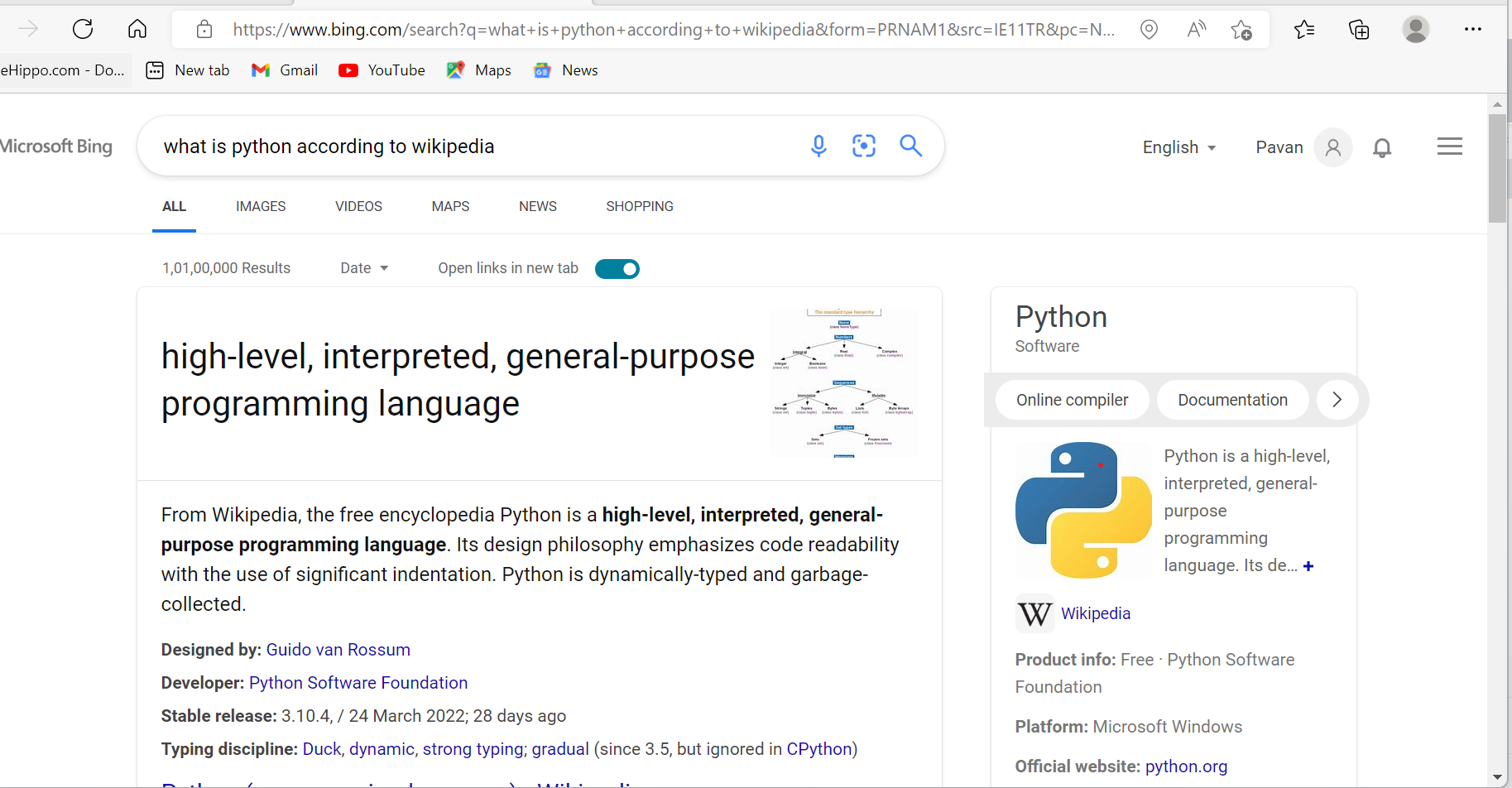


Figure 8.2 Output for Google search

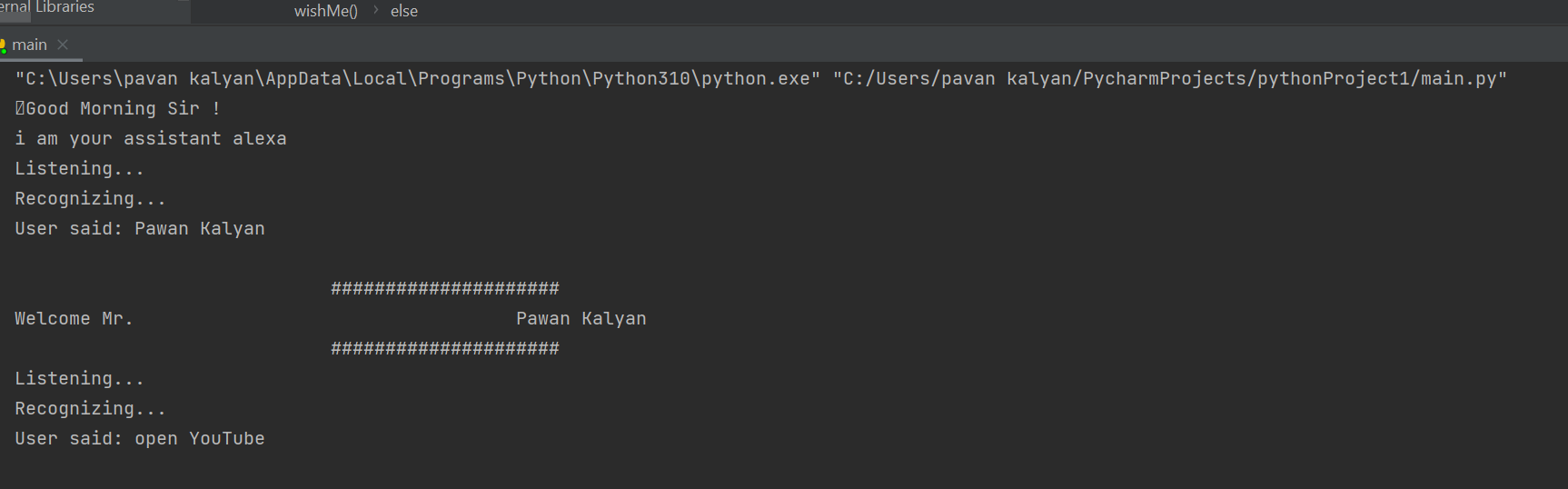


Figure 8.3 input for open Youtube

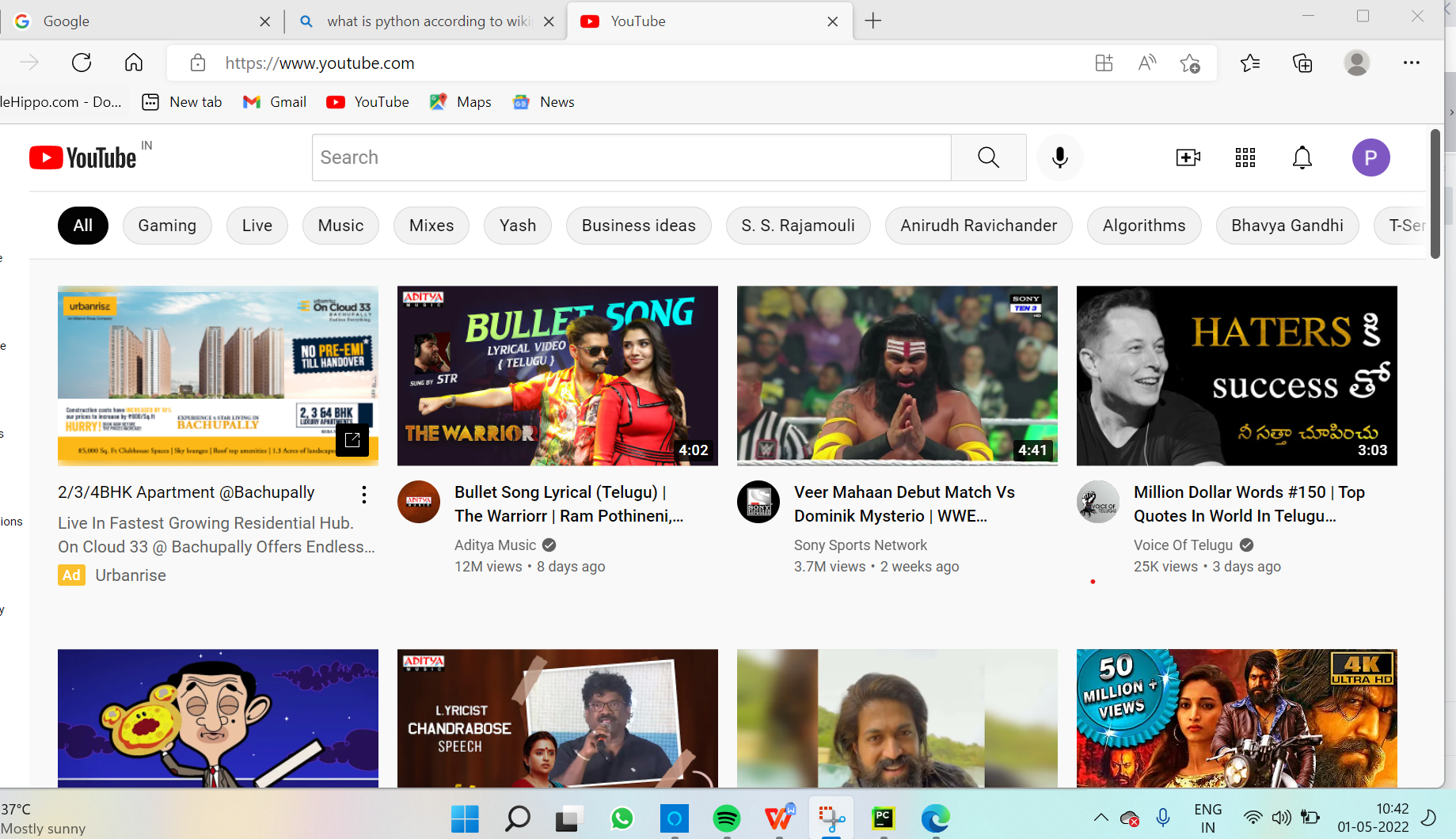


Figure 8.4 Output for open Youtbe

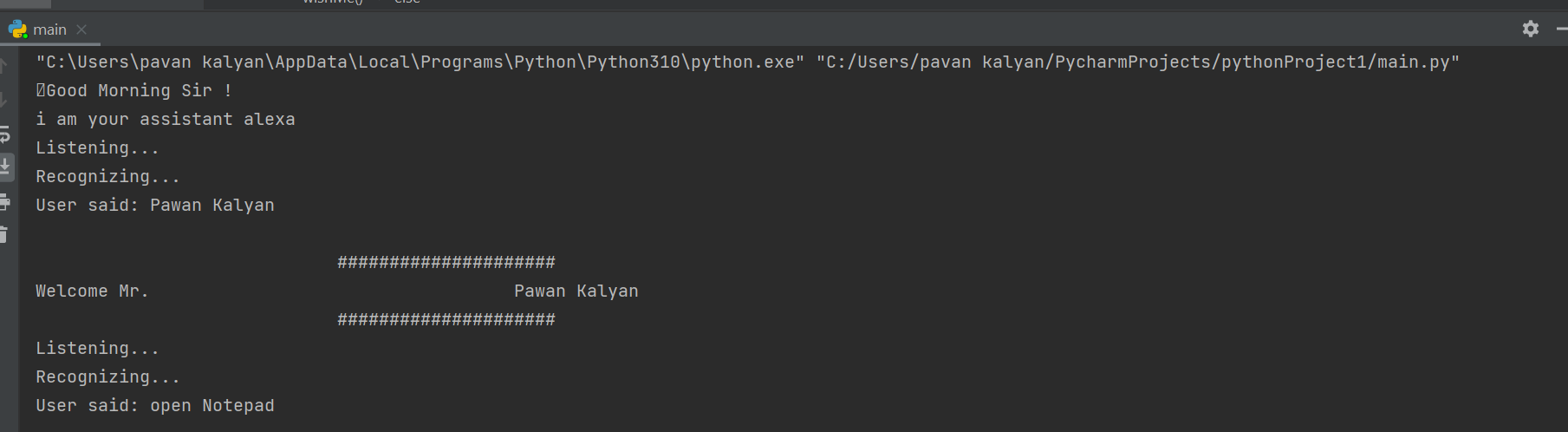


Figure 8.5 input for open notepad

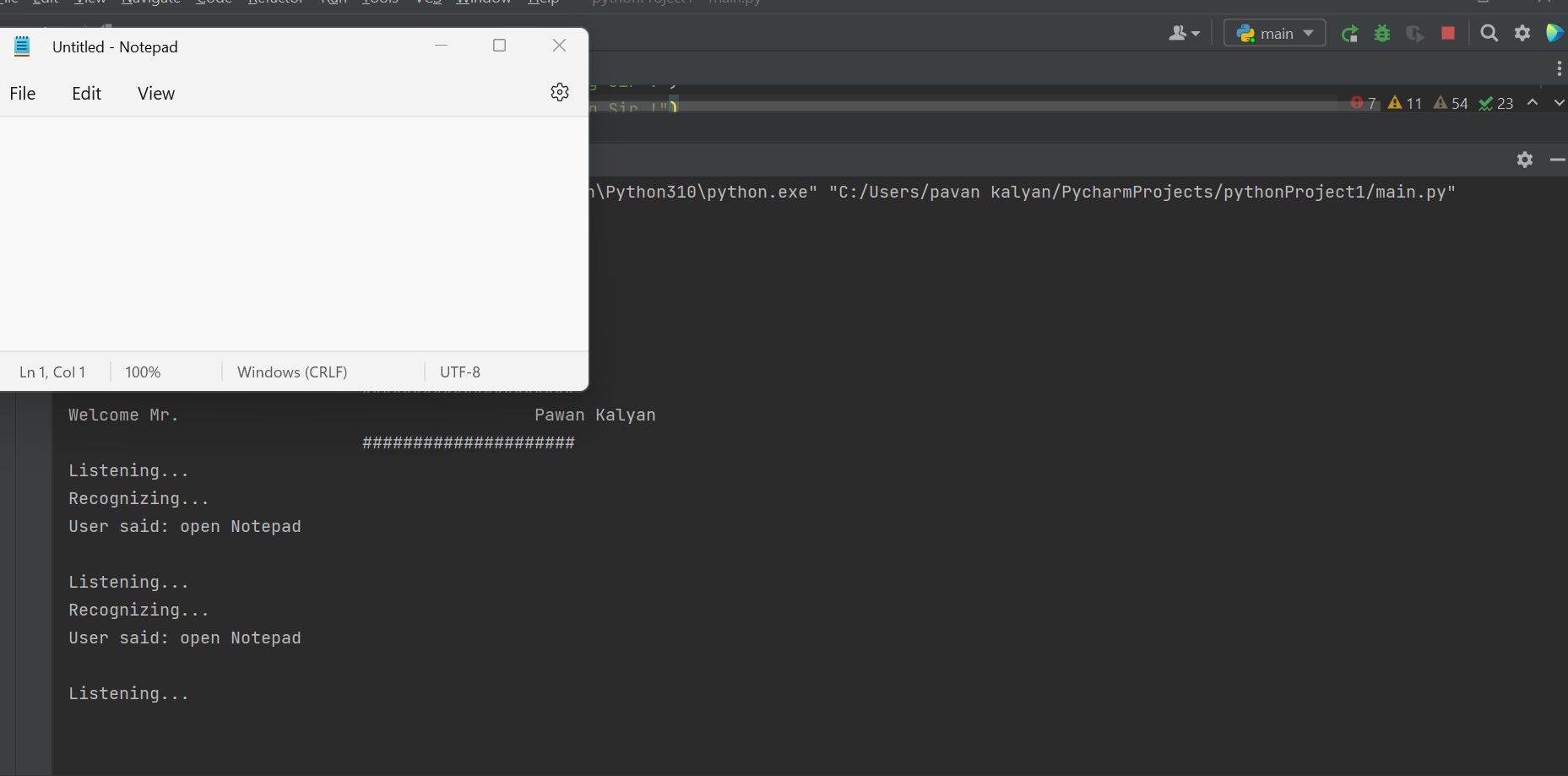


Figure 8.6 Output for open notepad

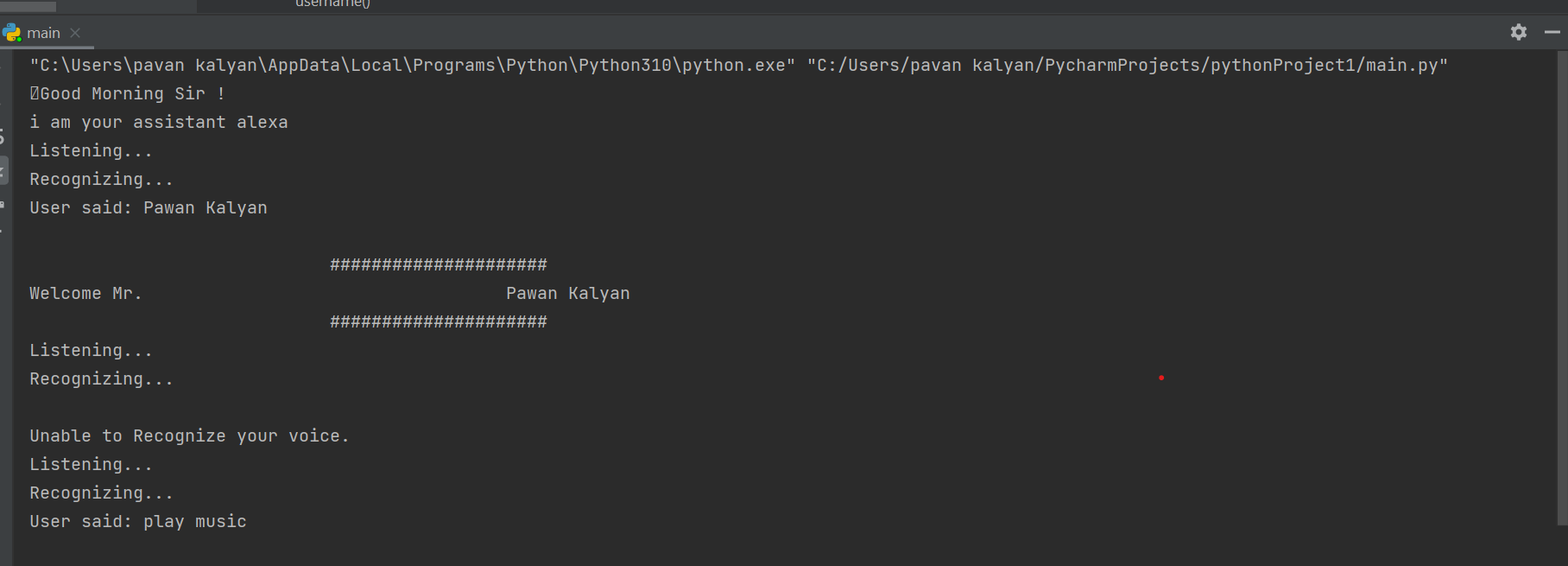


Figure 8.7 input for play music

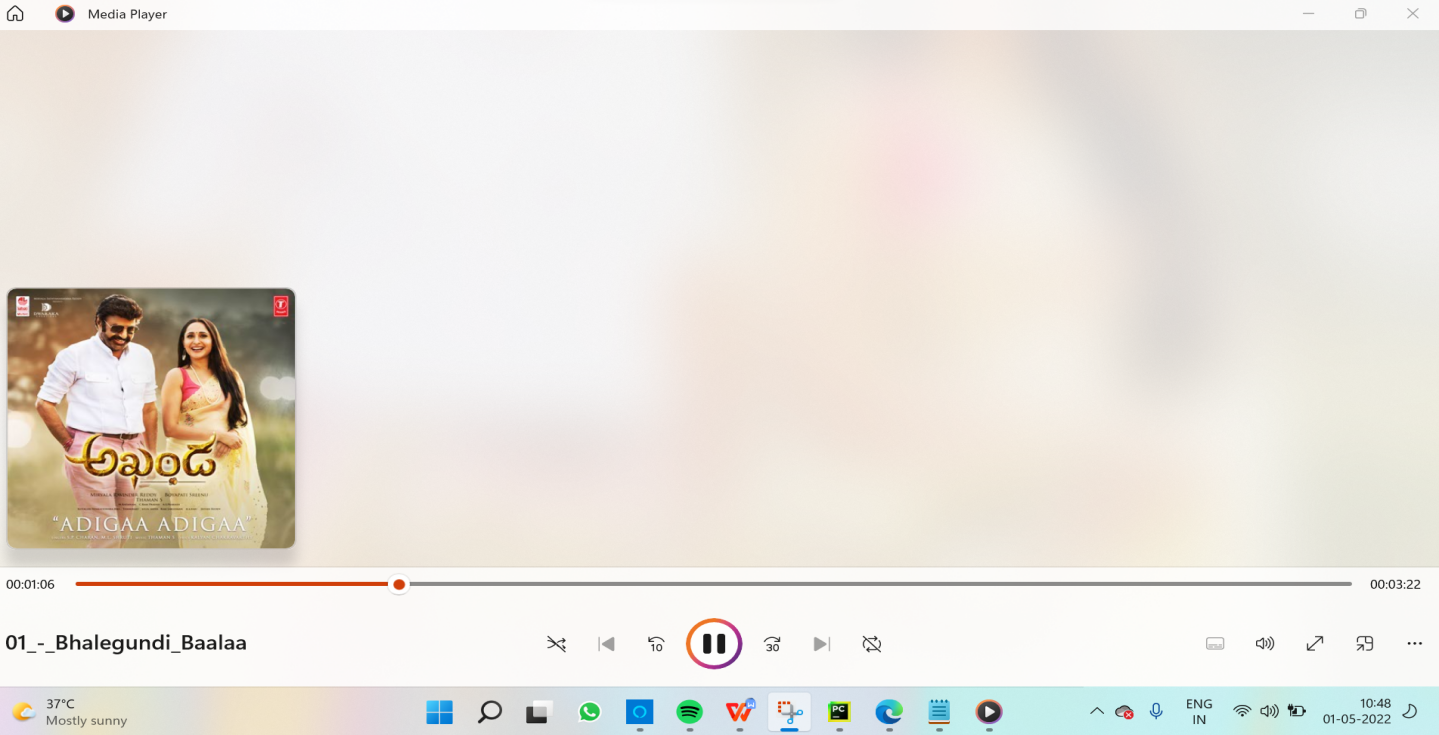


Figure 8.8 Output for play music

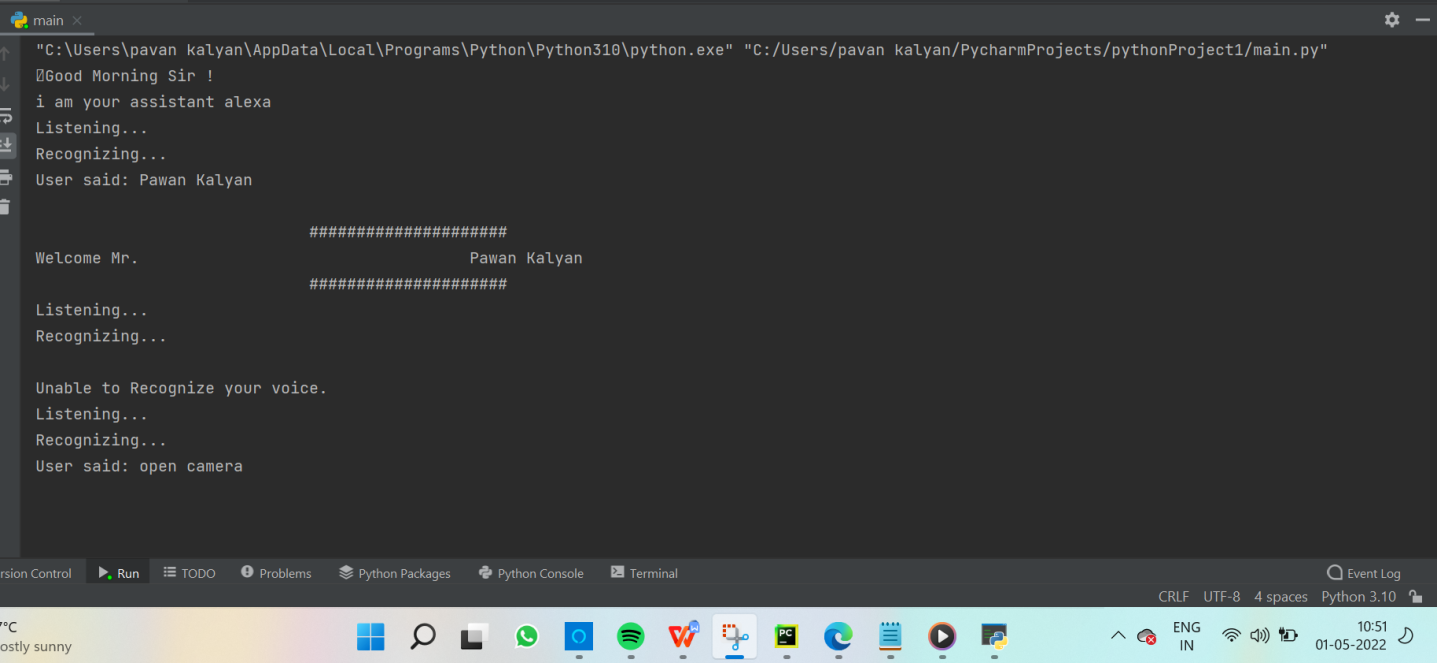


Figure 8.9 input for open camera

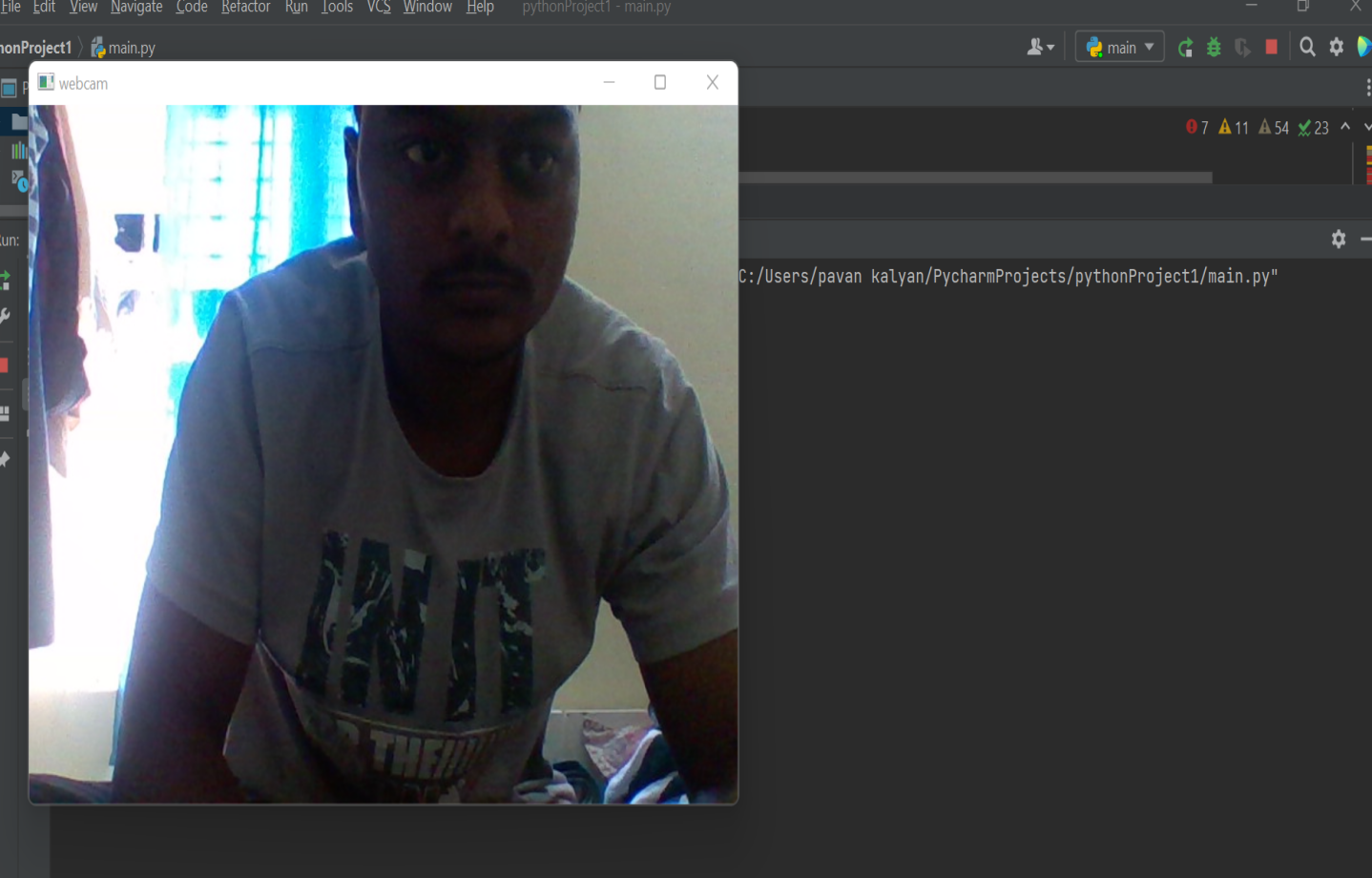


Figure 8.10 Output for open camera

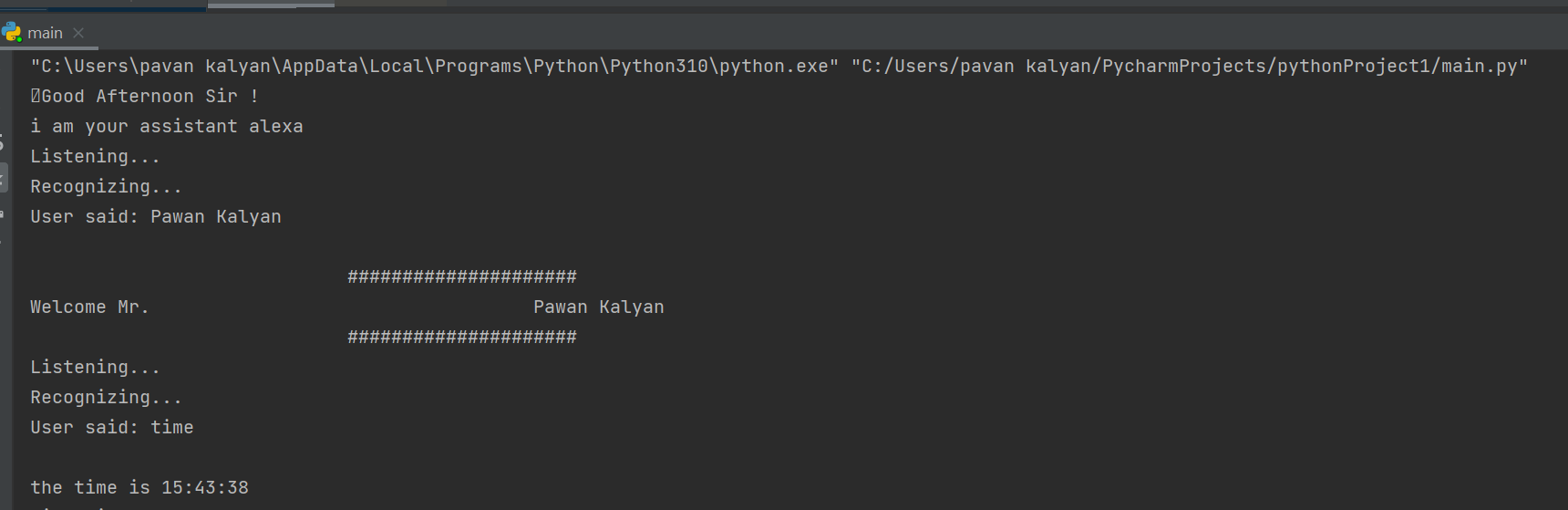


Figure 8.11 input and Output for tell me the time

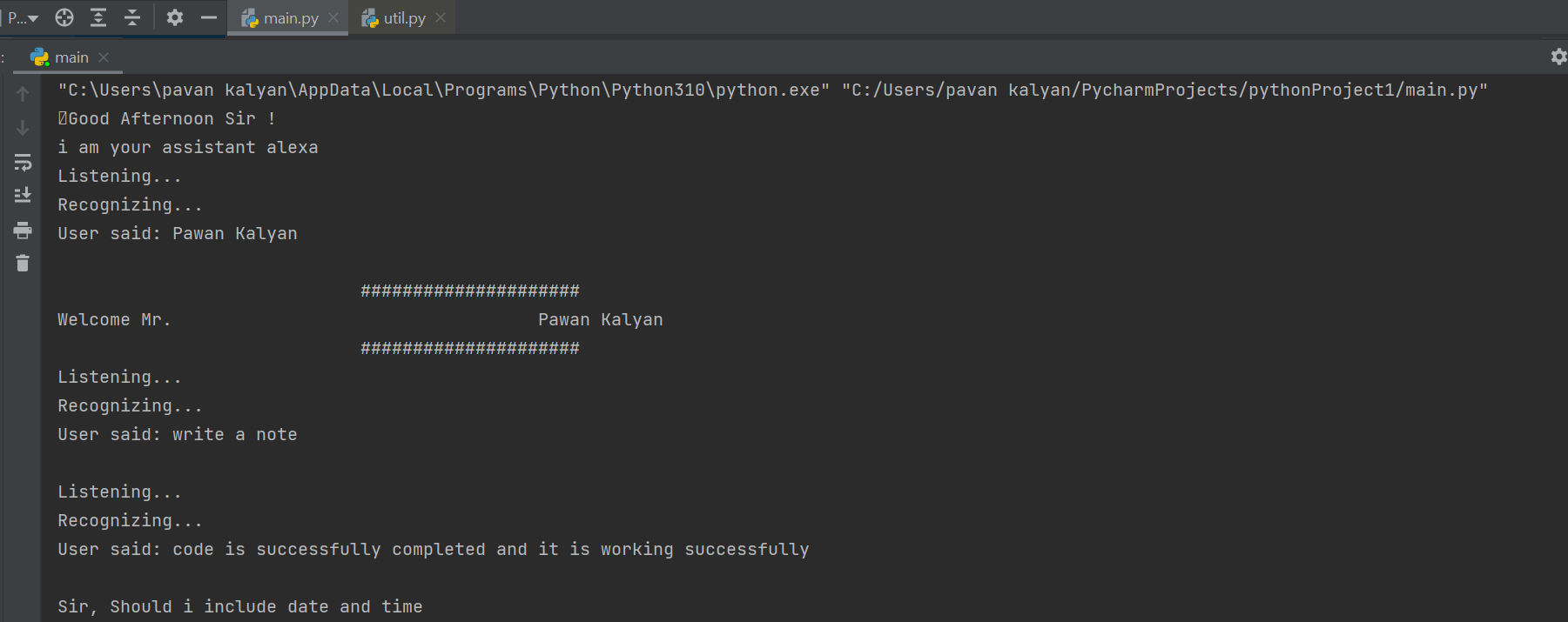


Figure 8.12 input for write a note

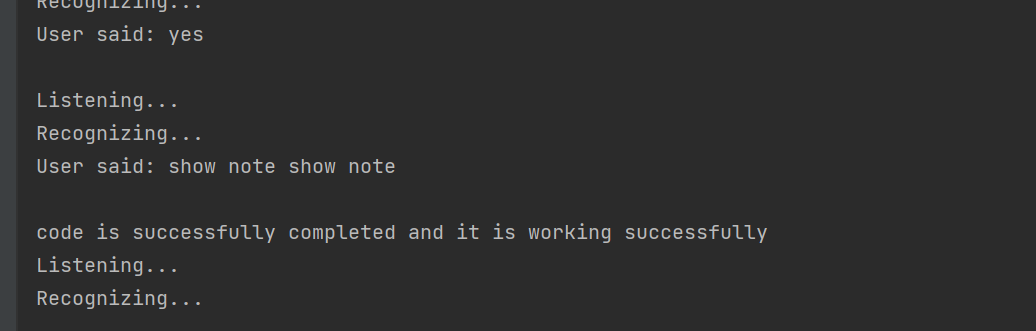


Figure 8.13 Output for showing note

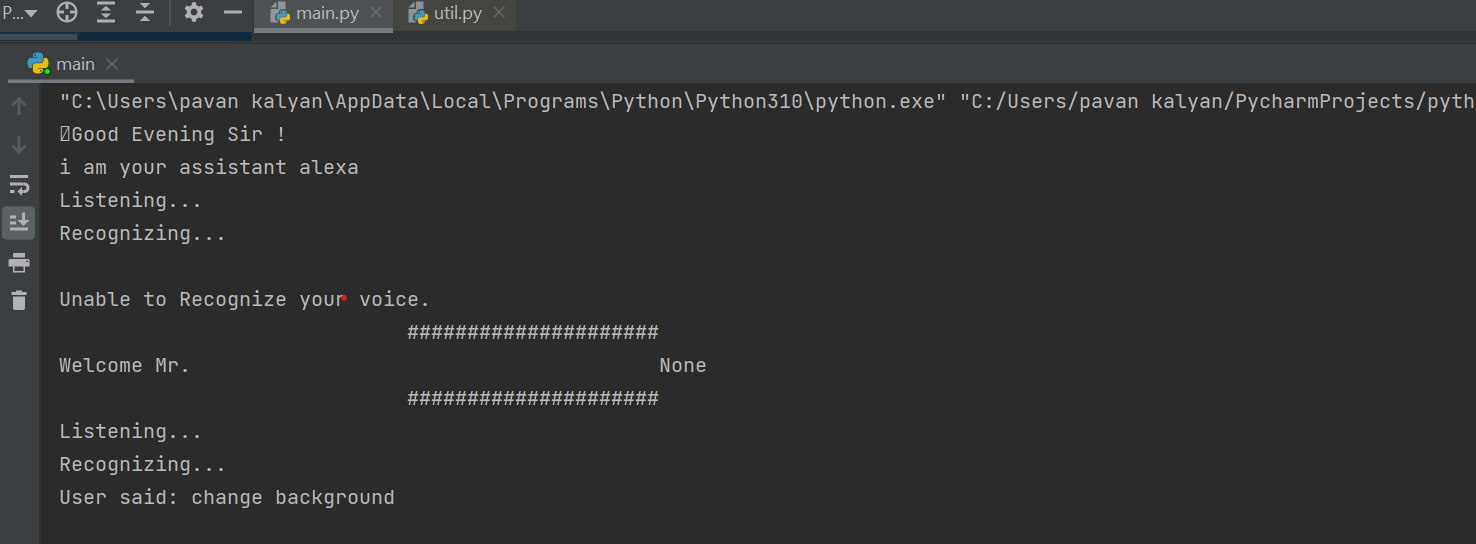


Figure 8.14 input for changing background



Figure 8.15 Output for changing background

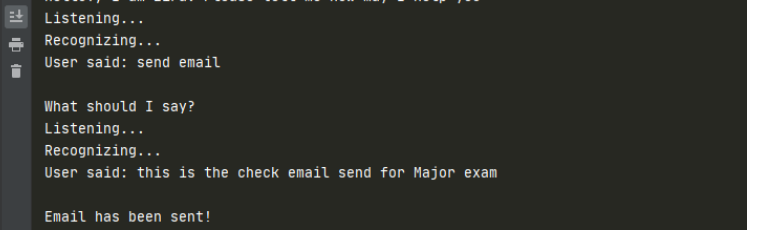


Figure 8.16 input for send mail

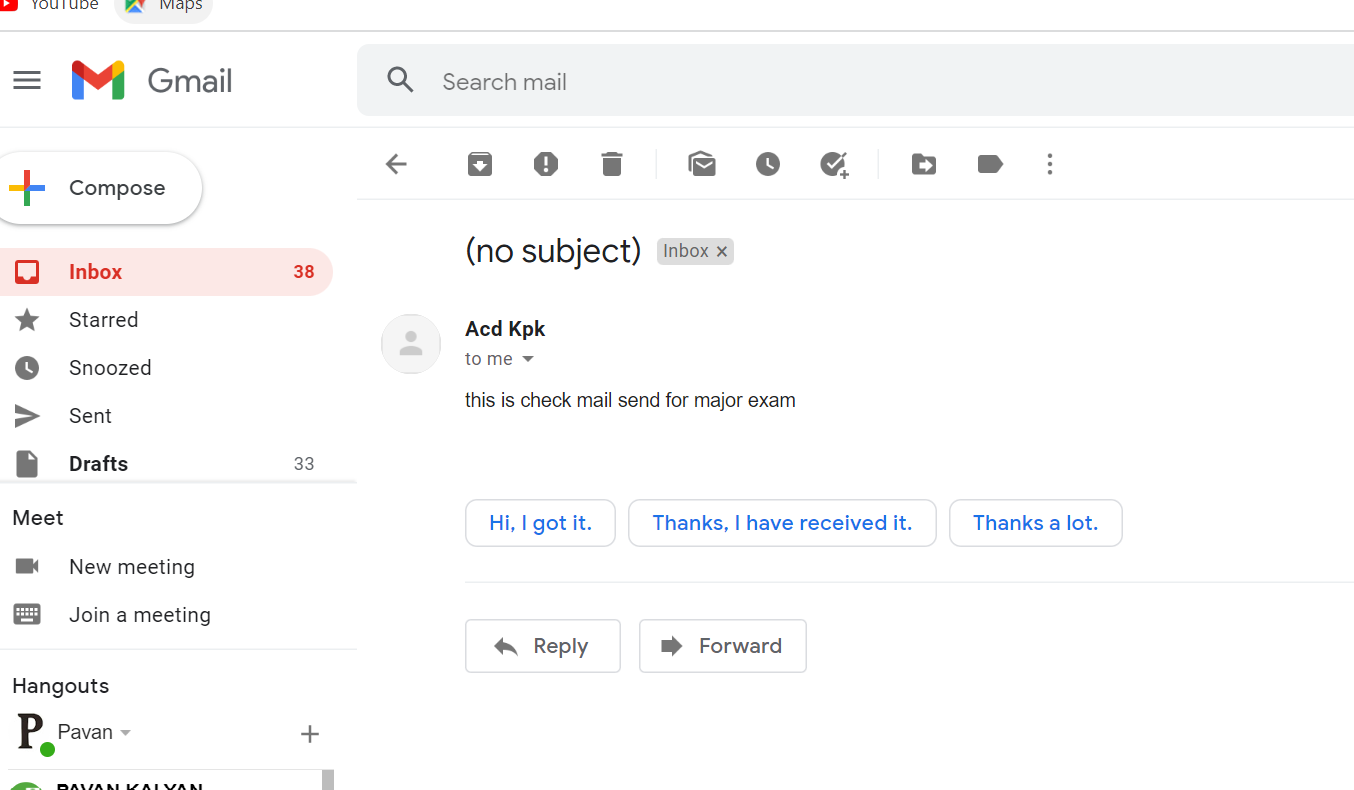


Figure 8.17 output for send mail

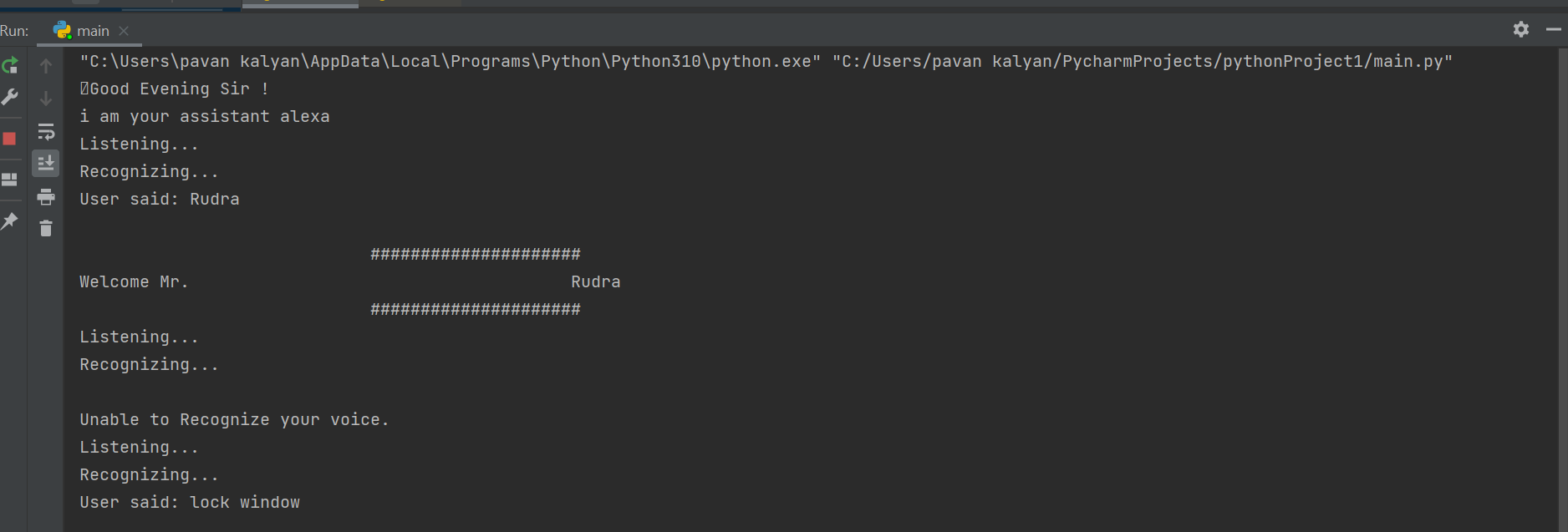


Figure 8.18 input for lock window

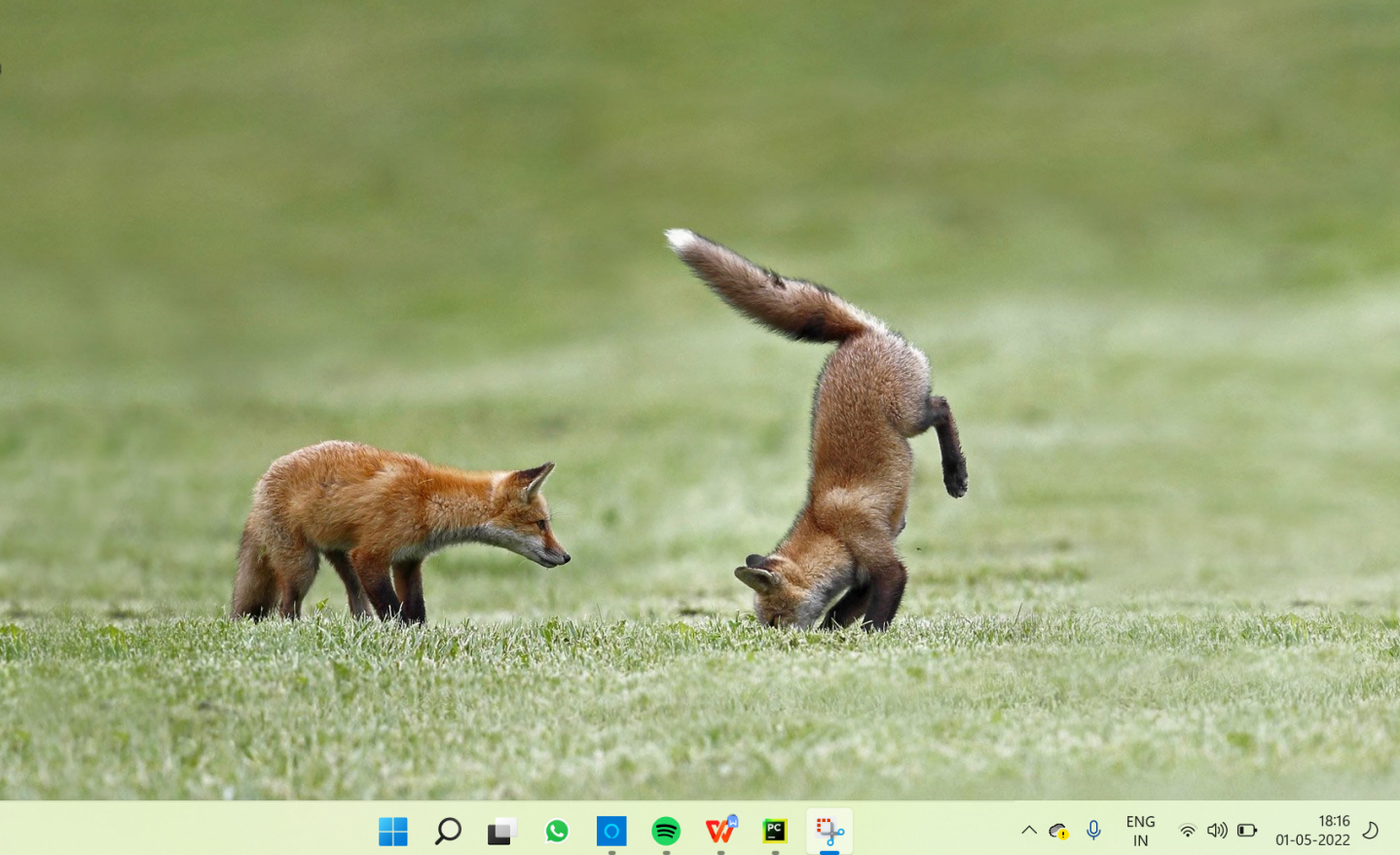
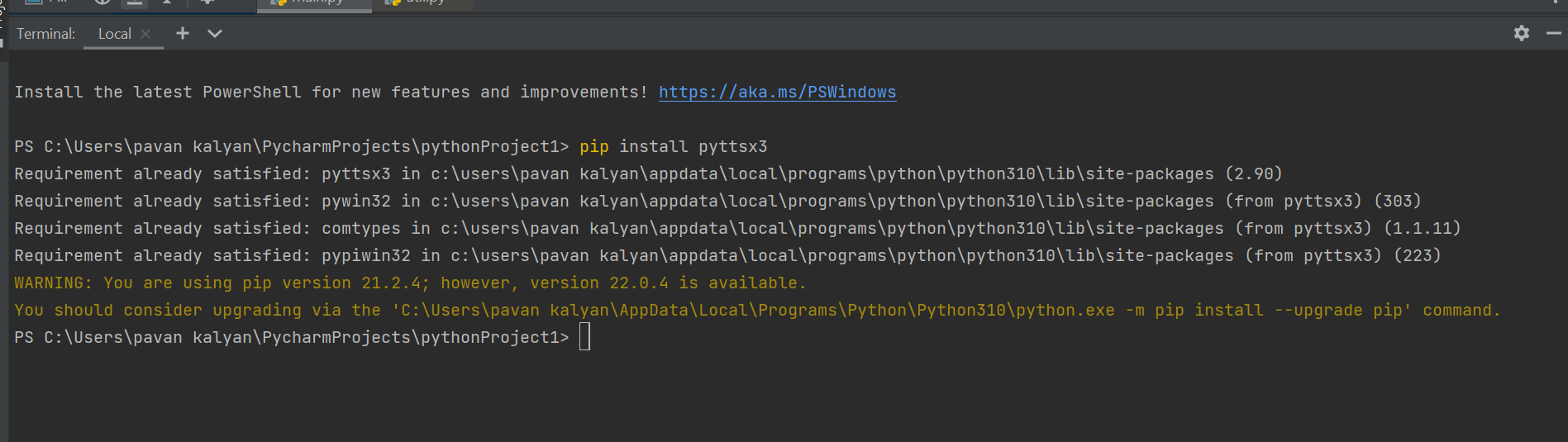


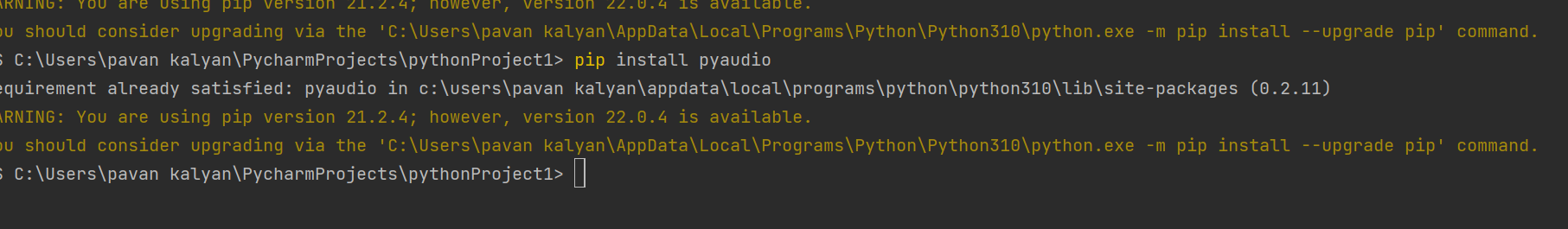
Figure 8.19 output lock window

**8.1 How to install modules in Pycharm Terminal (screenshots) :**

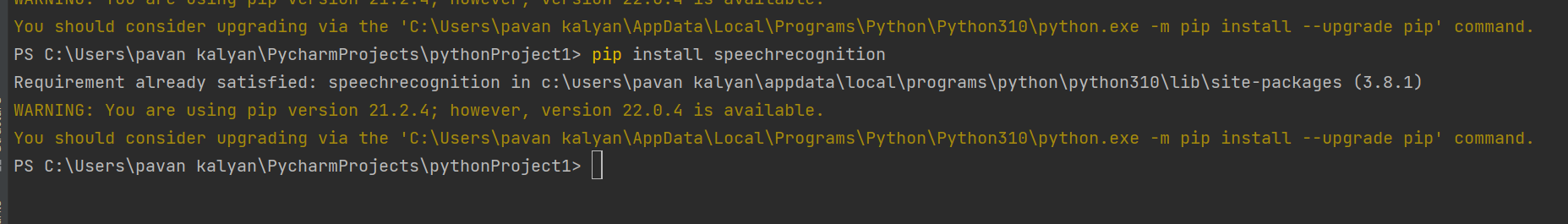
* Pyttsx3:



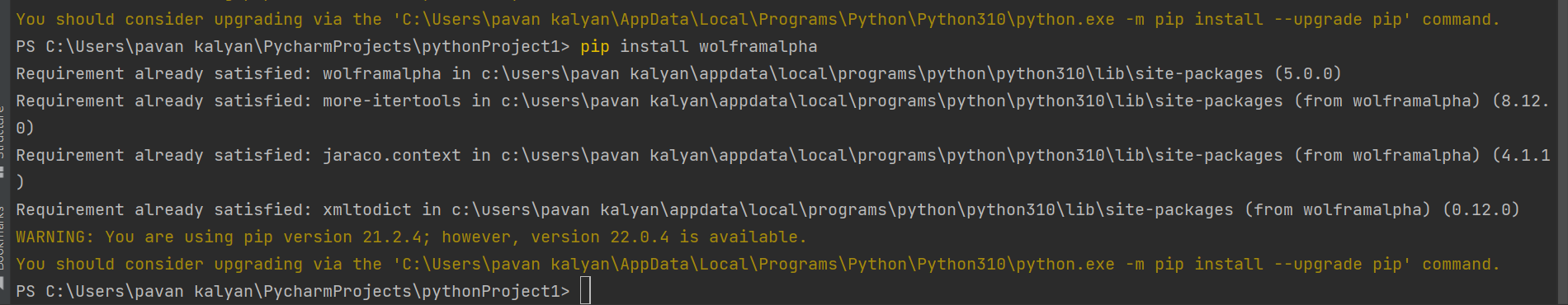
* **Pyaudio :**



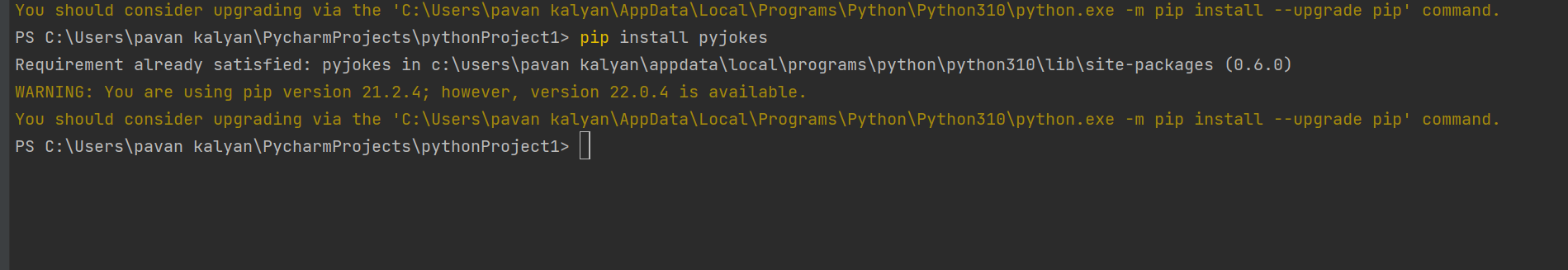
* **Speechrecognition :**



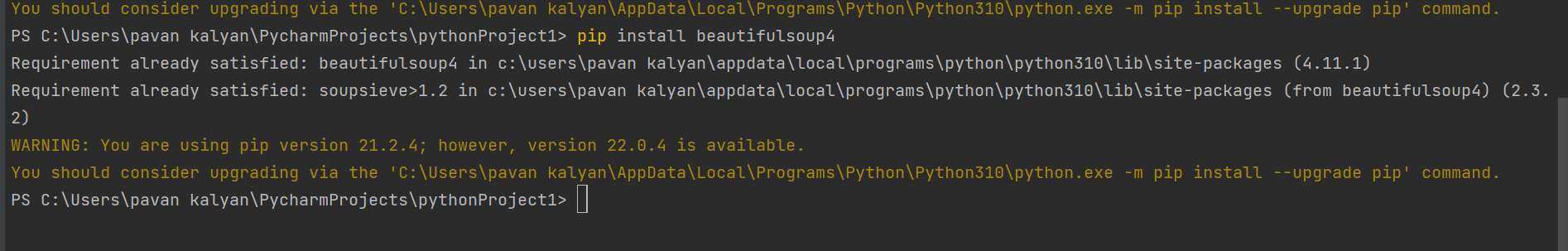
* **Wolframalpha:**



* Pyjokes:



* **Beautifulsoup4:**



1. **System Testing**

The system testing is done on fully integrated system to check whether the requirements are matching or not. The system testing for Alexa desktop assistant focuses on the following four parameters:

9.1. **FUNCTIONALITY**

In this we check the functionality of the system whether the system performs the task which it was intended to do. To check the functionality each function was checked and run, if it is able to execute the required task correctly then the system passes in that particular functionality test. For example to check whether Alexa can search on Google or not, as we can see in the figure 7.1, user said “Open Google”, then Alexa asked, ”What should I search on Google?” then user said, “What is Python”, Jarvis open Google and searched for the required input.

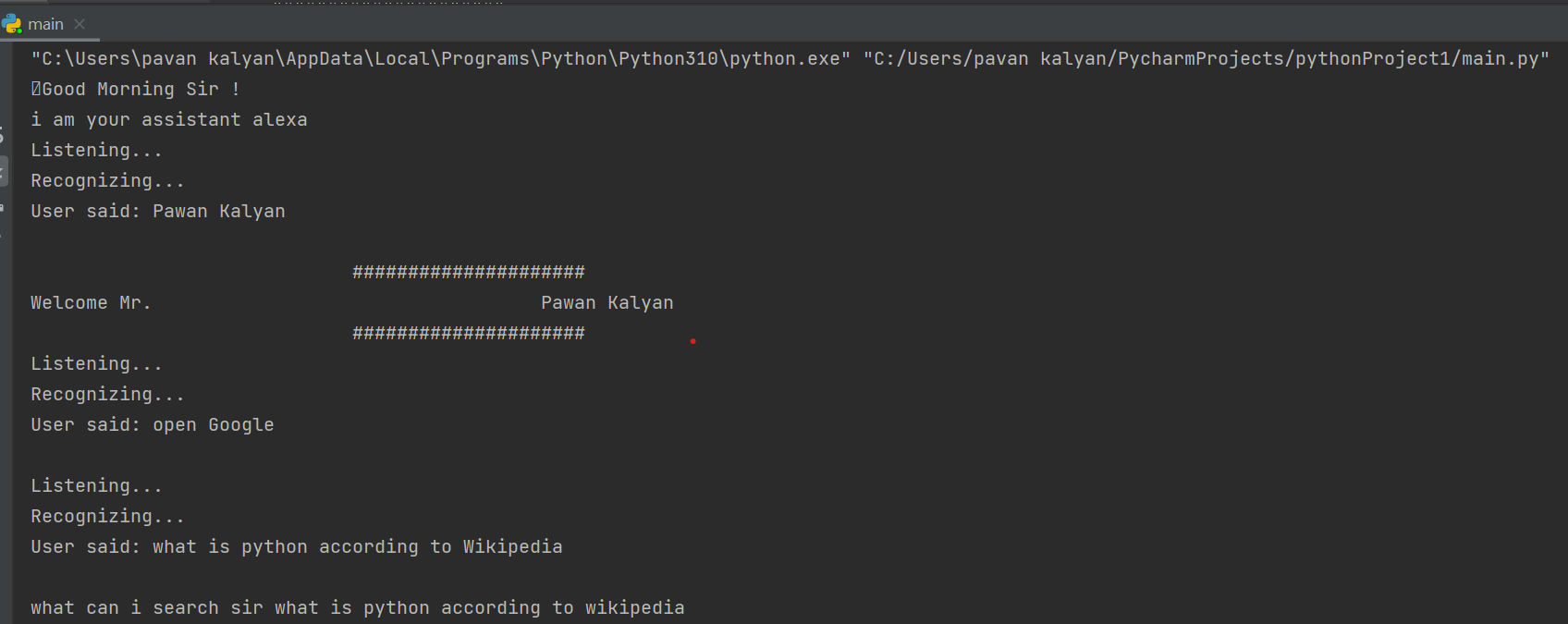


Figure 9.1 Input through voice commands

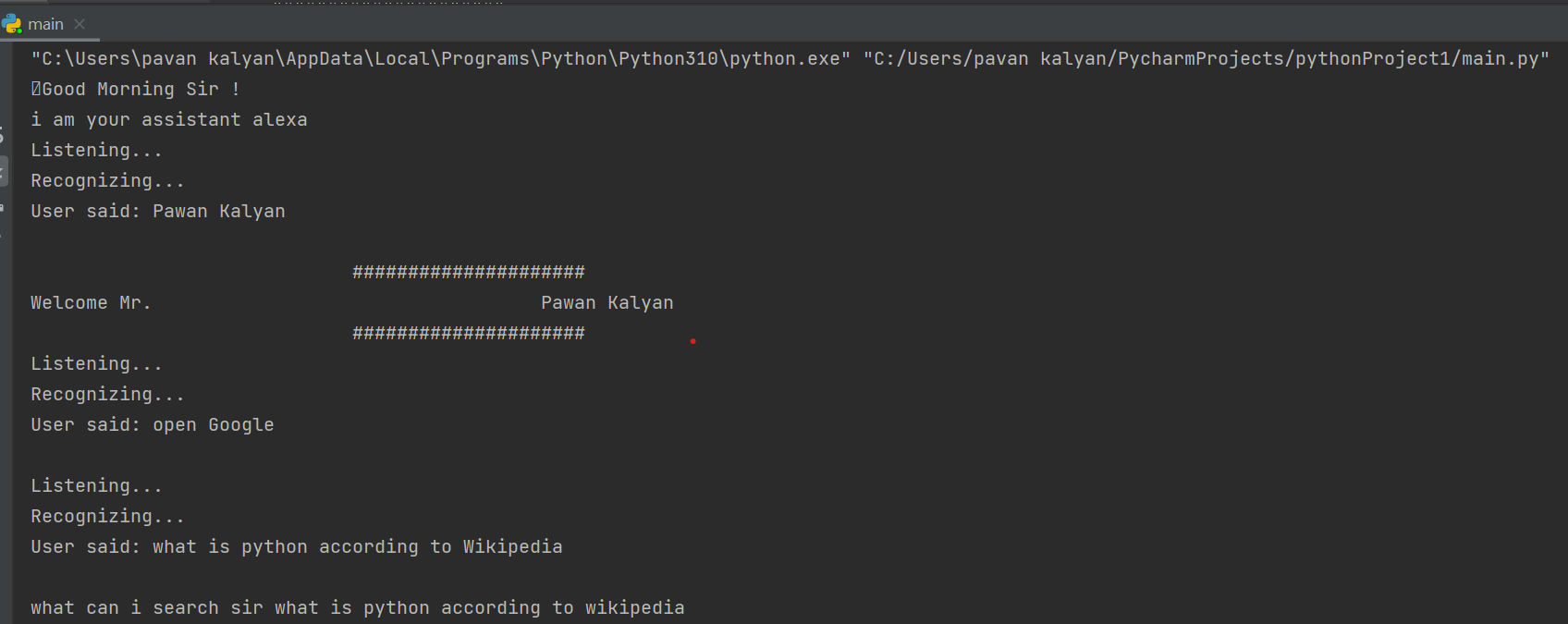


Figure 9.2 Output

9.2. **USABILITY**

Usability of a system is checked by measuring the easiness of the software and how user friendly it is for the user to use, how it responses to each query that is being asked by the user.It makes it easier to complete any task as it automatically do it by using the essential module or libraries of Python, in a conversational interaction way. Hence any user when instruct any task to it, they feel like giving task to a human assistant because of the conversational interaction for giving input and getting the desired output in the form of task done. The desktop assistant is reactive which means it know human language very well and understand the context that is provided by the user and gives response in the same way, i.e. human understandable language, English. So user finds its reaction in an informed and smart way. The main application of it can be its multitasking ability. It can ask for

continuous instruction one after other until the user “QUIT” it. It asks for the instruction and listen the response that is given by user without needing any trigger phase and then only executes the task.

9.3. **SECURITY**

The security testing mainly focuses on vulnerabilities and risks. As Alexa is a local desktop application, hence there is no risk of data breaching through remote access. The software is dedicated to a specific system so when the user logs in, it will be activated.

9.4. **STABILITY**

Stability of a system depends upon the output of the system, if the output is bounded and specific to the bounded input then the system is said to be stable. If the system works on all the poles of functionality then it is stable.

**10. Overall Contribution**

The project titled “ DESKTOP VOICE ASSISTANT: Alexa” was designed by me and Rudra. From installing of all the packages, importing, creating all the necessary functions.

I, Rudra have done all the research before making this project, designed the

requirement documents for the requirements and functionalities, wrote synopsis and all the documentation, code and made the project in such a way that it is deliverable at each stage.I and Rudra written the complete code in Python language and in PyCharm IDE from where it was very easy to install the packages and libraries, Rudra have created the functions like takeCommand(), wishMe() and taskExecution() which has the following functionalities, like takeCommand() which is used to take the command as input through microphone of user and returns the output as string, wishMe() that greets the user according to the time like Good Morning, Good Afternoon and Good Evening and taskExecution()which contains all the necessary task execution definition like sendEmail(), pdf\_reader(), news() and many conditions in if condition like “open Google”, “open notepad”, “search on Wikipedia” ,”play music” and “open command prompt” etc.While making this project I realized that with the advancement Alexa can perform any task with same effectiveness or can say more effectively than us. By making this project, I and Rudra realized that the concept of AI in every field is decreasing human effort and saving time. Functionalities of this project include, It can send emails, It can read PDF, It can send text on WhatsApp, It can open command prompt, your favorite IDE, notepad etc., It can play music, It can do Wikipedia searches for you, It can open websites like Google, YouTube, etc., in a web browser, It can give weather forecast, It can give desktop reminders of your choice. It can have some basic conversation.

At last, I and Rudra updated my report and completed it by attaching all the necessary screen captures of inputs and outputs, mentioning the limitations and scope in future of this project.

**11** : **Conclusion**

Alexa is a very helpful voice assistant without any doubt as it saves time of the user by conversational interactions, its effectiveness and efficiency. But while working on this project, there were some limitations encountered and also realized some scope of enhancement in the future which are mentioned below:

11.1. LIMITATIONS

Security is somewhere an issue, there is no voice command encryption in this project.

Background voice can interfere

Misinterpretation because of accents and may cause inaccurate results.

Alexa cannot be called externally anytime like other traditional assistants like Google Assistant can be called just by saying, “Ok Google!”

11.2 **SCOPE FOR FUTURE WORK**

* Make Alexa to learn more on its own and develop a new skill in it.
* Alexa android app can also be developed.
* Make more Alexa voice terminals.
* Voice commands can be encrypted to maintain security.

**12. References:**

[How Do Voice Assistants Work? The Basics Explained - Miquido Blog](https://www.miquido.com/blog/what-are-voice-assistants/?msclkid=f57d22a6b30411ecaf9aa853db7322d3)

[How Voice Assistants Are Changing Our Lives | Smartsheet](https://www.smartsheet.com/voice-assistants-artificial-intelligence?msclkid=8d3001c7b2f511ec88c6c7aaf8575800)

<https://www.c-sharpcorner.com/article/creating-a-voice-assistant-using-python-and-its-libraries/>

[Create Voice Assistant using Python - Python Geeks](https://pythongeeks.org/python-voice-assistant-project/" \l ":~:text= Create Voice Assistant using Python  1,Python Voice Assistance Project Output.  More )

[Creating A Voice Assistant Using Python And It's Libraries (c-sharpcorner.com)](https://www.c-sharpcorner.com/article/creating-a-voice-assistant-using-python-and-its-libraries/)

[Build Your Own Voice Assistant with Python - CodeSpeedy](https://www.codespeedy.com/voice-assistant-with-python/)

DOUGLAS O’SHAUGHNESSY, SENIOR MEMBER, IEEE, “Interacting With Computers by Voice: Automatic Speech Recognition and Synthesis” proceedings of THE IEEE, VOL. 91, NO. 9, SEPTEMBER 2003

Nil Goksel-Canbek2 Mehmet EminMutlu, “On the track of Artificial Intelligence: Learning with Intelligent Personal Assistant” International Journal of Human Sciences, 13(1), 592-601. doi:10.14687/ijhs.v13i1.3549.

Easwara Moorthy, A., Vu, K.-P.L.: Privacy Concerns for Use of Voice Activated Personal Assistant in the Public Space. International Journal of Human-Computer Interaction 31, 307–335 (2015)

Tsiao, J.C.-S., Tong, P.P., Chao, D.Y.: NaturalLanguage Voice-Activated Personal Assistant, United States Patent (10), Patent No.: US 7,216,080 B2 (45), 8 May 2007

Lopez, G., Quesada, L., Guerrero, L.A.: Alexa vs Siri vs Cortana vs Google assistant: a comparison of speech-based natural user interfaces. Conference Paper, January 2018

Kepuska, V., Bohouta, G.: Next generation of virtual personal assistants (Microsoft Cortana, Apple Siri, Amazon Alexa and Google Home). In: IEEE Conference (2018)

Gong, L.: San Francisco, CA (US) United States US 2003.01671.67A1 (12) Patent Application Publication c (10) Pub. No.: US 2003/0167167 A1 Gong (43) Pub. Date: 4 September 2003 for Intelligent Virtual Assistant

Sumitkumar Sarda, Yash Shah, Monika Das, Nikita Saibewar, ShivprasadPatil,”VPA: Virtual Personal Assistant” Published in 2017 International Journal of Computer Applications (0975 – 8887) Volume 165 – No Grabianowski, E. (2011). How Speech Recognition Works. Retrieved February 2016, from How Stuff Works: http://electronics.howstuffworks.com/gadgets/h igh-tech-gadgets/speech-recognition1.html

Nagesh Singh Chauhan,“Build Your First Voice Assistant”,March,2019 [https://towardsdatascience//com](https://towardsdatascience/com)