

J.I.D.

An Artificial Brilliant

“*A project based on simple python coding*”

**Project Team:AyushBadgujar,Shaivi Mishra, Mansi Ghodke, Anmol Dureja**

INDEX

| Sr. No. | Chapters | Page no. |
| --- | --- | --- |
| 1 | Certificate | 3 |
| 2 | Acknowledgement | 4 |
| 3 | Introduction | 5 |
| 4 | Objective | 6 |
| 5 | Requirements  -System requirements  -Python Edition  -Speak function  -Take Command function  -Username function  -Date and time function  -Wishing function | 7-11 |
| 6 | Working | 12-13 |
| 7 | Code | 14-28 |
| 8 | Output | 29-33 |
| 9 | Modules Used | 34 |
| 10 | Features | 35 |
| 11 | Conclusion | 36 |
| 12 | Bibliography | 37 |



**CERTIFICATE**

This is to certify that “*Ayush Badgujar”, “Anmol Dureja”, “Mansi Ghodke”, “Shaivi Mishra”* of class XI-A of Lokmanya Tilak International School, Koparkhairane, Navi Mumbai have successfully completed the computer science project on “14th December 2021” under my supervision and to my complete satisfaction during academic year 2021-2022.

Date: 14/12/2021

Teacher:

*Mrs. Poonam Gupta* (Computer Science teacher)

ACKNOWLEDGEMENTS

We would like to express our sincere gratitude to our educator/instructor Mrs. Poonam Gupta (Computer Science Teacher) for her guidance which led to the completion of this project.

We thank her for her patience, enthusiasm and the opportunity to work on such an educational project. We would also like to thank our group mates for their contribution and their ideas. We are grateful for the hours of work they put in this project. Without their knowledge and expertise on their respective fields this project would not have been possible.

Lastly, we would like to thank our fellow classmates and all the people associated, which helped and guided us through this journey. Everybody's efforts and time are greatly valued. We are whole heartedly thankful for everything.

JID Project Team

INTRODUCTION

A virtual assistant, also called AI assistant or digital assistant is an application program that understands natural language voice commands and completes tasks for the user. 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. The birth of the first virtual assistant; however, began with IBM Simon in the early 1990s. It was a digital speech recognition technology that became a feature of the virtual computer.

The Siri voice assistant was released as an app for iOS in February 2010. It was the first modern digital virtual assistant installed on a smartphone. In this project we have tried to build a similar virtual assistant to do day to day basic tasks. But it makes you wonder “WHY DO WE EVEN NEED THIS?” or “WHAT ALL CAN IT DO?”.

Let us take a look over this in the coming section.

OBJECTIVE

For most of us, the ultimate luxury would be an assistant who always listens for your call, anticipates your every need, and takes action when necessary. That luxury is now available, thanks to voice assistants.

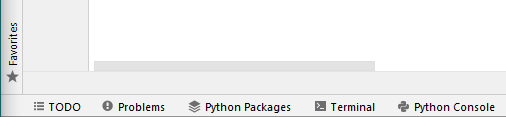
Voice assistants can perform a variety of actions after hearing a command. They can turn on lights, answer questions, play music, place online orders and many more things. Voice assistants are a very powerful tool for the blind people. As they can’t type anything on the keyboard, a voice assistant helps them to connect with the internet world as well as with their device easily.

With the addition of numerous apps in our phones, our voice can be a type of remote control for our lives. As of 2017, the capabilities and usage of virtual assistants are expanding rapidly, with new products entering the market and a strong emphasis voice user interfaces.

Therefore, keeping in mind of this usage, we have given a start to a voice assistant which may be upgraded in future and work very similar to the developed ones like Amazon Echo, Apple’s Siri and Google’s Google Assistant.

REQUIREMENTS

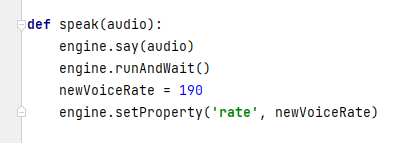
In this project, we will be using Python (version 3.9.0). And we would be working in PyCharm IDE (PyCharm Community Edition 2020.2.3) which can be downloaded from the link given in the end of the file.

There are no specific system requirements but, this code requires an exemplary ram space to run. 4 Gb or more is preferred. A lower RAM storage may result in slow running of this code. 

This project requires range of modules that can to be installed through the internet. As we are using PyCharm as a python editor, installation of the required modules written at the end of project, is advised.

To build our virtual assistant we would require few functions that are necessary for working of our PA. These do not include special functions used for specific tasks, these contain only simple yet important ones.

FUNCTION TO MAKE COMPUTER SPEAK

Firstly we would want the computer to speak. For that we would be making a speak(audio) function which would speak whatever command it has been given.

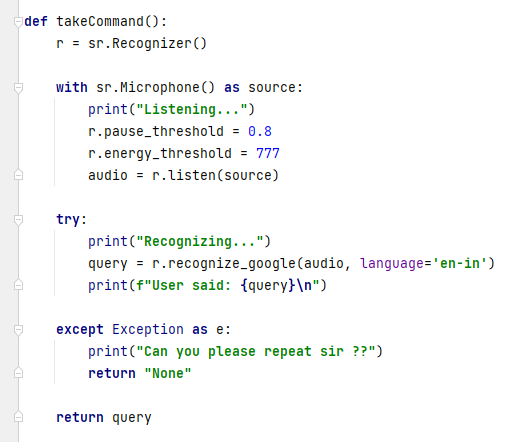
FUNCTION TO ENABLE COMPUTER UNDERSTAND WHAT WE SAY

Speech recognition module has been imported to make the computer understand our commands and reply to them.

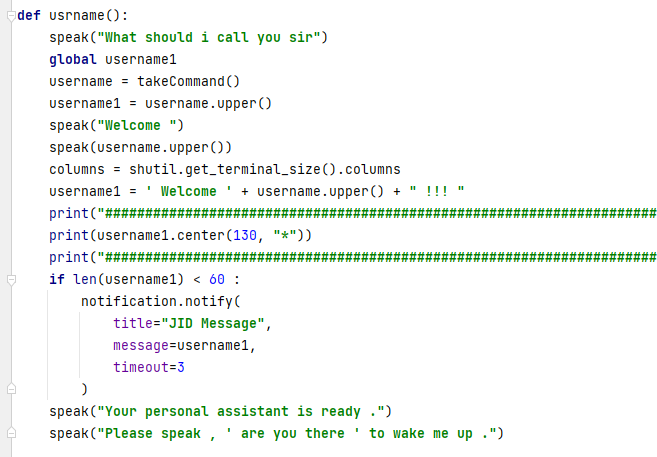
Microphone of the computer (can be inbuilt or connected through USB) will work as source of commands for our assistant.

The computer will to try recognizing what we are saying, giving us a specific time to reply. The “print(Listening…)” command is used to know when the computer is trying to reach us or is ready to listen to us.

If no sound is heard by the microphone, it would print a statement stating us to repeat.



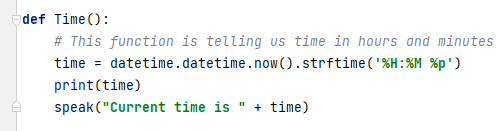
FUNCTION TO REGISTER OUR NAME

We find that many a times we want our virtual assistant use our name. the virtual assistant will ask the user for his/her name to welcome them and use it whenever required. 

FUNCTION TO TELL TIME AND DATE

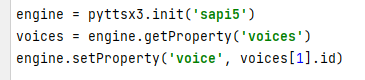
For this we have to import “time” and “datetime” module, which would enable us to get current time and date.   
Code for day – 

Code for time-



LINES OF CODE TO CHOOSE OUR FAVORITE VOICE

Everyone would want their virtual assistant to be able to speak in different voices.

For that we have to type a set of lines of code which will help us to change voice of the assistant. We would have to import ‘pyttsx3’ module to use this. We can change voice by typing different numbers in place of 1 in the last line of the code given below. 

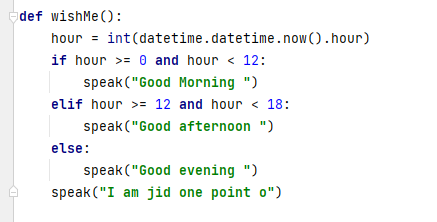
For instance if we type [1] we would get a female voice whereas if we type [0] then we will have a male voice. Note that these both are default voices. For more voices you must do this:

Go to Search option in Windows 🡪 Search Speech Settings 🡪 Under Manage Voice Title press the ‘+’ button to add voices 🡪 Now you can select your default voices 🡪 Now in the coding part write in the function “print(voices[1])” .

{Try typing different numbers like 1,0,2,3 etc. to find out when voice is registered to which number}.

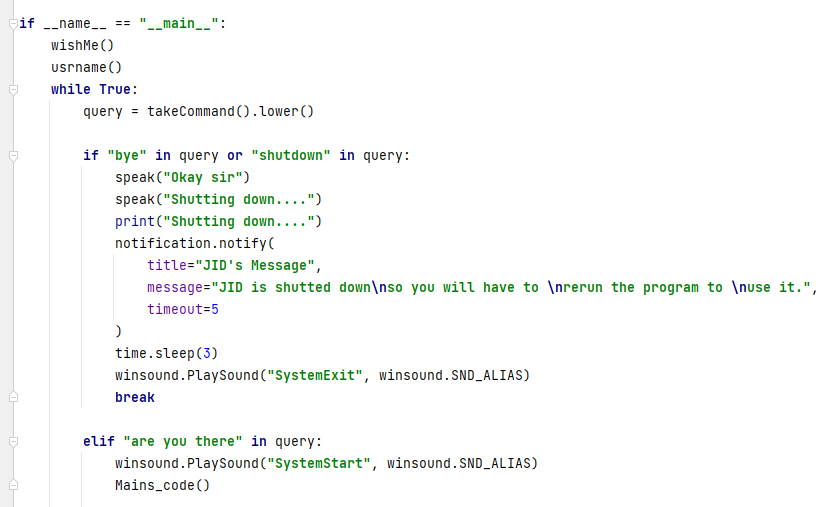
FUNCTION TO MAKE VIRTUAL ASSISTANT WISH US

Whenever we would start our program, our virtual assistant would greet us according to current time.



WORKING

Working of this entire project is very simple. The entire project is based on trial and error. Let’s see how…

Firstly, we would use initial voice, or simply extract voices from the computer and use them. Then we would run a simple while loop that we’ll show in coming pages.

if \_\_name\_\_ == “main”: is used to execute some code only if the file was run directly, and not imported.

After this we will use wishMe() and usrname() functions (explained in previous pages) to make our virtual assistant wish us according to real time and then take our name input, these are first time setup which will make our virtual assistant somewhat get to know us.

We made a while loop and took audio input from the user to wake up or shut down our virtual Assistant.

We have defined a function named Mains\_code() which recognizes our input and give answer based on that. The idea of looping is based on the fact that the program will be running in background till we stop or terminate it. If we want help of our assistant we are just need to speak “are you there” and our virtual assistant will be alerted and ready to use.

It will give us a small timespan in which we would have to speak and if we don’t, then it will stop.

As this is an Alpha version this is not capable of understand a wide range of words and variation of sentences. So, it is a requirement that one must use the appropriate triggering phrases. The code is given on the next page.

CODE

*#ALL MODULES***import** pyttsx3  
**import** speech\_recognition **as** sr  
**import** datetime  
**import** pyaudio  
**import** wikipedia  
**import** pywhatkit  
**import** pyjokes  
**import** googlesearch  
**import** pywhatkit **as** kt  
**import** math  
**import** os  
**import** shutil  
**import** wolframalpha  
**import** tones  
**import** webbrowser  
**import** requests  
**import** time  
**import** pytz  
**import** geocoder  
**import** pyautogui  
**import** winsound  
**from** playsound **import** playsound  
**import** pygame  
**from** tkinter **import** LabelFrame, Button, Label  
**import** tkinter **as** Tk  
**import** random  
**from** plyer **import** notification  
**from** bs4 **import** BeautifulSoup  
  
  
*#IMPORTANT LIST*how\_are\_you = [**"I'm Better On The Inside Than I Look On The Outside"**, **"Hopefully not as good as I'll ever be"**, **"Living the dream"**, **"I'm Taking Over The World"**, **"The best thing that happened to me today so far, is you asking me about my day, by the way for the record i am fine sir"**, **"I'm fine sir"**, **"Better than ever sir"**]  
  
  
*#VOICE EXTRACTING COMMANDS*engine = pyttsx3.init(**'sapi5'**)  
voices = engine.getProperty(**'voices'**)  
print(voices[2])  
engine.setProperty(**'voice'**, voices[2].id)  
  
  
*#DEFINING SOME FUNCTION FOR LATER USE***def** usrname():  
 speak(**"What should i call you sir"**)  
 **global** username1  
 **global** username  
 username = takeCommand()  
 username1 = username.upper()  
 speak(**"Welcome "**)  
 speak(username.upper())  
 columns = shutil.get\_terminal\_size().columns  
 username1 = **' Welcome '** + username.upper() + **" !!! "** print(**"##################################################################################################################################"**)  
 print(username1.center(130, **"\*"**))  
 print(**"##################################################################################################################################"**)  
 **if** len(username1) < 60 :  
 notification.notify(  
 title=**"JID Message"**,  
 message=username1,  
 timeout=3  
 )  
 speak(**"Your personal assistant is ready ."**)  
 speak(**"Please speak , ' are you there ' to wake me up ."**)  
  
**def** speak(audio):  
 engine.say(audio)  
 engine.runAndWait()  
 newVoiceRate = 150  
 engine.setProperty(**'rate'**, newVoiceRate)  
  
**def** takeCommand():  
 r = sr.Recognizer()  
  
 **with** sr.Microphone() **as** source:  
 print(**"Listening..."**)  
 r.pause\_threshold = 0.8  
 r.energy\_threshold = 777  
 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(**"Can you please repeat sir ??"**)  
 **return "None"  
  
 return** query  
  
**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 jid one point o"**)  
  
  
**def** tellDay():  
 *# This function is for telling the day of the week* day = datetime.datetime.today().weekday() + 1  
  
 *# This line tells us about the number that will help us in telling the day* Day\_dict = {1: **'Monday'**, 2: **'Tuesday'**,  
 3: **'Wednesday'**, 4: **'Thursday'**,  
 5: **'Friday'**, 6: **'Saturday'**,  
 7: **'Sunday'**}  
  
 **if** day **in** Day\_dict.keys():  
 day\_of\_the\_week = Day\_dict[day]  
 print(day\_of\_the\_week)  
 speak(**"The day is "** + day\_of\_the\_week)  
  
**def** make\_request(url):  
 response = requests.get(url)  
 **return** response.text  
  
**def** Time():  
 *# This function is telling us time in hours and minutes* time = datetime.datetime.now().strftime(**'%H:%M %p'**)  
 print(time)  
 speak(**"Current time is "** + time)  
  
*#BRAIN OF ENTIRE PROJECT***def** Mains\_code():  
 **global** query  
 query = takeCommand().lower()  
  
 **if 'tell me about' in** query:  
 speak(**"Searching on wikipedia..."**)  
 query = query.replace(**"tell me about"**, **" "**)  
 **try**:  
 results = wikipedia.summary(query, sentences = 1)  
 results2 = wikipedia.summary(query, sentences = 3)  
 speak(**'According to wikipedia'**)  
 notification.notify(  
 title=**"Search Results"**,  
 message=results,  
 timeout=5  
 )  
 speak(results)  
 print(results2)  
  
 **except** Exception **as** e:  
 print(e)  
 speak(e)  
 speak(**"try again sir once"**)  
  
 **elif 'who is' in** query:  
 speak(**"searching on google..."**)  
 query = query.replace(**"who is"**, **""**)  
 kt.search(query)  
 c = googlesearch.search(query)  
 speak(**'According to google'**)  
 print(c)  
 speak(c)  
  
 **elif 'play' in** query:  
 speak(**"Searching on youtube..."**)  
 query = query.replace(**'play'**, **''**)  
 speak(**'Playing'** + query)  
 pywhatkit.playonyt(query)  
  
 **elif 'play on youtube' in** query:  
 speak(**"Searching on youtube..."**)  
 query = query.replace(**'play on youtube'**, **''**)  
 speak(**'Playing'** + query)  
 pywhatkit.playonyt(query)  
  
 **elif 'search on youtube' in** query:  
 speak(**"Searching on youtube..."**)  
 query = query.replace(**'search on youtube'**, **''**)  
 speak(**'Playing'** + query)  
 pywhatkit.playonyt(query)  
  
 **elif 'time' in** query:  
 query = query.replace(**'time'**, **''**)  
 Time()  
  
 **elif 'your age' in** query:  
 speak(**"As a matter of fact sir age is just a number. For your knowledge my age is approximately 7 months"**)  
  
 **elif 'your name' in** query:  
 notification.notify(  
 title=**f"User said = {**query**}"**,  
 message=**"JID 1.0 Aplha Version \nJunior Intercommunication Device"** )  
 speak(**"My name is Jude one point o"**)  
  
 **elif 'i am tired' in** query:  
 IAT = **"It's okay to get tired sir, you can take rest if you want."** notification.notify(  
 title=**f"User said = {**query**}"**,  
 message=IAT,  
 timeout=2  
 )  
 speak(IAT)  
  
 **elif 'joke' in** query:  
 JK = pyjokes.get\_joke()  
 print(JK)  
 notification.notify(  
 title=**"JOKE ! "**,  
 message=JK,  
 timeout=3  
 )  
 speak(JK)  
 print(**" "**)  
 speak(**"Do you want me to tell you another joke sir??"**)  
 joke\_reply1 = takeCommand()  
 **if "yes" in** joke\_reply1:  
 JK = pyjokes.get\_joke()  
 print(JK)  
 notification.notify(  
 title=**"JOKE ! "**,  
 message=JK,  
 timeout=3  
 )  
 speak(JK)  
 **elif "no" in** joke\_reply1:  
 speak(**"Okai sir "**)  
 **else**:  
 **pass  
  
 elif 'search on google' in** query:  
 speak(**"searching on google..."**)  
 query = query.replace(**"search on google"**, **""**)  
 kt.search(query)  
 G\_search = googlesearch.search(query)  
 speak(**'According to google'**)  
 print(G\_search)  
 speak(G\_search)  
  
 **elif 'how are you' in** query:  
 speak(random.choice(how\_are\_you))  
  
 **elif 'add' in** query:  
 query = query.replace(**"add"**, **""**)  
 query = query.replace(**"and"**, **"+"**)  
 query = query.replace(**"to"**, **"+"**)  
 query = query.replace(**"one"**, **"1"**)  
 query = query.replace(**"two"**, **"2"**)  
 query = query.replace(**"three"**, **"3"**)  
 query = query.replace(**"four"**, **"4"**)  
 query = query.replace(**"five"**, **"5"**)  
 query = query.replace(**"six"**, **"6"**)  
 query = query.replace(**"seven"**, **"7"**)  
 query = query.replace(**"eight"**, **"8"**)  
 query = query.replace(**"nine"**, **"9"**)  
 query = query.replace(**"zero"**, **"0"**)  
 addition = float(eval(query))  
 notification.notify(  
 title=**"Solution"**,  
 message=str(addition),  
 timeout=2  
 )  
 speak(**"answer is "**)  
 speak(addition)  
  
 **elif 'subtract' in** query:  
 query = query.replace(**"subtract"**, **"-"**)  
 query = query.replace(**"from"**, **"+"**)  
 query = query.replace(**"one"**, **"1"**)  
 query = query.replace(**"two"**, **"2"**)  
 query = query.replace(**"three"**, **"3"**)  
 query = query.replace(**"four"**, **"4"**)  
 query = query.replace(**"five"**, **"5"**)  
 query = query.replace(**"six"**, **"6"**)  
 query = query.replace(**"seven"**, **"7"**)  
 query = query.replace(**"eight"**, **"8"**)  
 query = query.replace(**"nine"**, **"9"**)  
 query = query.replace(**"zero"**, **"0"**)  
 subtraction = float(eval(query))  
 notification.notify(  
 title=**"Solution"**,  
 message=str(subtraction),  
 timeout=2  
 )  
 speak(**"answer is "**)  
 speak(subtraction)  
 print(subtraction)  
  
 **elif 'multiply' in** query:  
 query = query.replace(**"multiply"**, **""**)  
 query = query.replace(**"and"**, **"\*"**)  
 query = query.replace(**"to"**, **"\*"**)  
 query = query.replace(**"one"**, **"1"**)  
 query = query.replace(**"two"**, **"2"**)  
 query = query.replace(**"three"**, **"3"**)  
 query = query.replace(**"four"**, **"4"**)  
 query = query.replace(**"five"**, **"5"**)  
 query = query.replace(**"six"**, **"6"**)  
 query = query.replace(**"seven"**, **"7"**)  
 query = query.replace(**"eight"**, **"8"**)  
 query = query.replace(**"nine"**, **"9"**)  
 query = query.replace(**"zero"**, **"0"**)  
 multiplication = float(eval(query))  
 notification.notify(  
 title=**"Solution"**,  
 message=str(multiplication),  
 timeout=2  
 )  
 speak(**"answer is "**)  
 speak(multiplication)  
 print(multiplication)  
  
 **elif 'divide' in** query:  
 query = query.replace(**"divide"**, **""**)  
 query = query.replace(**"by"**, **"/"**)  
 query = query.replace(**"and"**, **"/"**)  
 query = query.replace(**"one"**, **"1"**)  
 query = query.replace(**"two"**, **"2"**)  
 query = query.replace(**"three"**, **"3"**)  
 query = query.replace(**"four"**, **"4"**)  
 query = query.replace(**"five"**, **"5"**)  
 query = query.replace(**"six"**, **"6"**)  
 query = query.replace(**"seven"**, **"7"**)  
 query = query.replace(**"eight"**, **"8"**)  
 query = query.replace(**"nine"**, **"9"**)  
 query = query.replace(**"zero"**, **"0"**)  
 division = float(eval(query))  
 notification.notify(  
 title=**"Solution"**,  
 message=str(division),  
 timeout=2  
 )  
 speak(**"answer is "**)  
 speak(division)  
 print(division)  
  
 **elif 'not feeling' in** query:  
 speak(**"Oh , which of the following things you are feeling ... can i know??..."**)  
 speak(**"Do you have headache sir"**)  
 pucha = takeCommand()  
 **if "yes" in** pucha:  
 speak(**"sir do you want me to suggest you any medicines"**)  
 pucha1 = takeCommand()  
 **if "yes" in** pucha1:  
 speak(**"Remedies that may reduce headache pain include aspirin, paracetamol and ibuprofen. Resting in a darkened room may help you sir."**)  
 notification.notify(  
 title=**"What you can do ??"**,  
 message=**"Medicines - Aspirin, Paracetamol and Ibuprofen"**,  
 timeout=2  
 )  
 **elif "no" in** pucha1:  
 speak(**"Okay sir "**)  
 **else**:  
 **pass** speak(**"Are you feeling like vomiting???"**)  
 pucha2 = takeCommand()  
 **if "yes" in** pucha2:  
 speak(  
 **"Taking small sips of oral rehydration solution (ORS) may help prevent dehydration. If fluids stay down, eating toast, crackers, gelatin or other easy-to-digest foods may ease an upset stomach."**)  
 **elif "no" in** pucha2:  
 **pass** speak(**"Are you feeling feverish ....?"**)  
 pucha3 = takeCommand()  
 **if "yes" in** pucha3:  
 speak(**"Can you please speak your temperature sir in celcius??"**)  
 fev = takeCommand()  
 a = eval(fev)  
 e = float(a)  
  
 **if** 36.6 <= e <= 37.2:  
 speak(**'You dont have fever sir no need to worry... '**)  
  
 **if** 37.2 < e <= 38.1:  
 speak(**'You have a mild fever'**)  
 speak(**'have a rest today sir'**)  
  
 **if** 38.1 < e < 45.0:  
 speak(**'Your temperature seems very high'**)  
 speak(**"do you want me to give names of some medicine sir ???"**)  
 pucha4 = takeCommand()  
 **if "yes" in** pucha4:  
 speak(**"Medications such as paracetamol and ibuprofen may help to ease your discomfort sir"**)  
 notification.notify(  
 title=**"What you can do ??"**,  
 message=**"Medicines - Paracetamol and Ibuprofen"**,  
 timeout=2  
 )  
  
 **elif 'no' in** pucha4:  
 speak(**"Okay sir no problem , take care..."**)  
  
 **elif** 35.5 <= e < 36.6:  
 speak(**'Your temperature seems little bit low'**)  
 speak(**'please eat high energitic food '**)  
  
 **elif** 30.0 < e < 35.5:  
 speak(**'Your temperature is very low '**)  
 speak(  
 **'Consult a doctor immediately or i can give you some names of medicines please ans yes or no sir ....'**)  
 pucha6 = takeCommand()  
 **if "yes" in** pucha6:  
 speak(**"Medications such as paracetamol and ibuprofen may help to ease your discomfort sir"**)  
 notification.notify(  
 title=**"What you can do ??"**,  
 message=**"Medicines - Aspirin, Paracetamol and Ibuprofen"**,  
 timeout=2  
 )  
  
 **elif 'no' in** pucha6:  
 speak(**"Okay sir no problem , take care..."**)  
  
 **else**:  
 speak(**'Your input data is wrong'**)  
 speak(**'or you are not alive rit now'**)  
 **else**:  
 speak(**"No problem sir , take care ...."**)  
  
 **elif "no" in** pucha:  
 speak(**"Are you feeling like vomiting???"**)  
 pucha2 = takeCommand()  
 **if "yes" in** pucha2:  
 speak(**"Taking small sips of oral rehydration solution (ORS) may help prevent dehydration. If fluids stay down, eating toast, crackers, gelatin or other easy-to-digest foods may ease an upset stomach."**)  
 **elif "no" in** pucha2:  
 **pass** speak(**"Are you feeling feverish ....?"**)  
 pucha3 = takeCommand()  
 **if "yes" in** pucha3:  
 speak(**"Can you please speak your temperature sir in celcius??"**)  
 fev = takeCommand()  
 a = eval(fev)  
 e = float(a)  
  
 **if** 36.6 <= e <= 37.2:  
 speak(**'You dont have fever sir no need to worry... '**)  
  
 **if** 37.2 < e <= 38.1:  
 speak(**'You have a mild fever'**)  
 speak(**'have a rest today sir'**)  
  
 **if** 38.1 < e < 45.0:  
 speak(**'Your temperature seems very high'**)  
 speak(**"do you want me to give names of some medicine sir ???"**)  
 pucha4 = takeCommand()  
 **if "yes" in** pucha4:  
 speak(**"Medications such as paracetamol and ibuprofen may help to ease your discomfort sir"**)  
 notification.notify(  
 title=**"What you can do ??"**,  
 message=**"Medicines - Aspirin, Paracetamol and Ibuprofen"**,  
 timeout=2  
 )  
  
 **elif 'no' in** pucha4:  
 speak(**"Okay sir no problem , take care..."**)  
  
  
 **elif** 35.5 <= e < 36.6:  
 speak(**'Your temperature seems little bit low'**)  
 speak(**'please eat high energitic food '**)  
  
  
 **elif** 30.0 < e < 35.5:  
 speak(**'Your temperature is very low '**)  
 speak(**'Consult a doctor immediately or i can give you some names of medicines please ans yes or no sir ....'**)  
 pucha5 = takeCommand()  
 **if "yes" in** pucha5:  
 speak(**"Medications such as paracetamol and ibuprofen may help to ease your discomfort sir"**)  
 notification.notify(  
 title=**"What you can do ??"**,  
 message=**"Medicines - Aspirin, Paracetamol and Ibuprofen"**,  
 timeout=2  
 )  
  
 **elif 'no' in** pucha5:  
 speak(**"Okay sir no problem , take care..."**)  
  
 **else**:  
 speak(**'Your input data is wrong'**)  
 speak(**'or you are not alive rit now'**)  
 **else**:  
 speak(**"No problem sir , take care ...."**)  
  
  
  
 **elif "stop" in** query:  
 speak(**"okay sir"**)  
  
 **elif "sine" in** query:  
 speak(**"Please say the angle below sir"**)  
 pi = math.pi  
 s1 = float(takeCommand())  
 s2 = math.sin(s1\*pi/180)  
 print(**"Sine of "**, s1, **"radians is "**, s2 )  
 s3 = **"Sine of "**, s1, **"radians is "**, s2  
 notification.notify(  
 title=**"Solution"**,  
 message=**f"={**s2**}"**,  
 timeout=2  
 )  
 speak(s3)  
  
 **elif "cos" in** query:  
 pi = math.pi  
 speak(**"Please say the angle below sir"**)  
 s4 = float(takeCommand())  
 s5 = math.cos(s4\*pi/180)  
 print(**"Cosine of "**, s4, **"radians is "**, s5 )  
 s6 = **"cosine of "**, s4, **"radians is "**, s5  
 notification.notify(  
 title=**"Solution"**,  
 message=**f"={**s5**}"**,  
 timeout=2  
 )  
 speak(s6)  
  
 **elif "tan" in** query:  
 pi = math.pi  
 speak(**"Please say the angle below sir"**)  
 s7 = float(takeCommand())  
 s8 = math.tan(s7\*pi/180)  
 print(**"Tan of "**, s7, **"radians is "**, s8 )  
 s9 = **"Tan of "**, s7, **"radians is "**, s8  
 notification.notify(  
 title=**"Solution"**,  
 message=**f"={**s8**}"**,  
 timeout=2  
 )  
 speak(s9)  
  
 **elif "coffee" in** query:  
 speak(**"Depends sir , if i am tired or not .... by the way i am tired so..... yes!!!!!!"**)  
 coffee1 = **"""  
 | |   
 |----------|   
 | |   
 | |   
 | |   
 |\_\_\_\_\_\_\_\_\_\_|   
  
 """** coffee2 = **"""  
 | |   
 | |   
 |\_\_\_\_\_\_\_\_\_\_|   
 | |   
 | |   
 |\_\_\_\_\_\_\_\_\_\_|   
  
 """** coffee3 = **"""  
 | |   
 | |   
 | |   
 | |   
 |----------|   
 |\_\_\_\_\_\_\_\_\_\_|   
  
 """** coffee4 = **"""  
 | |   
 | |   
 | |   
 | |   
 | |   
 |\_\_\_\_\_\_\_\_\_\_|   
  
 """** speak(**"Wait a sec making coffee..............oh......!!!! done !!!"**)  
 print(coffee1)  
 speak(**"can u see the glass............i am starting"**)  
 speak(**"I have started ........."**)  
 print(coffee2)  
 speak(**"Drinking..... aah!!!!!! "**)  
 speak(**"Almost done......"**)  
 print(coffee3)  
 speak(**"Almost there ....have you completed ..... you can never beat me haha ... "**)  
 print(coffee4)  
 speak(**"And there you go coffee is drunk .....entire glass empty ....."**)  
 speak(**"i am feeling much better and refresed..... "**)  
  
 **elif 'open gmail' in** query:  
 mail = webbrowser.open\_new\_tab(**"https://mail.google.com/mail/u/0/?tab=wm#inbox"**)  
 speak(**"Google Mail opening now"**)  
 time.sleep(5)  
  
 **elif 'space news' in** query:  
 news = webbrowser.open\_new\_tab(**"https://www.space.com/news"**)  
 speak(**'Here are some headlines from the Space dot com ,Happy reading'**)  
 time.sleep(6)  
  
 **elif 'search on wolfram alpha' in** query:  
 *#It can answer to computational and geographical questions and what question do you want to ask now* question = input(**"calculate: "**)  
 client = wolframalpha.Client(**"VLPYA4-3937WGJX48"**)  
 query = query.replace(**" "**, **""**)  
 res = client.query(**' '**.join(question))  
 answer = next(res.results).text  
 speak(**"calculating.."**)  
 print(answer)  
 speak(**f"The answer is {**answer**}"**)  
  
  
 **elif "weather" in** query:  
 api\_key = **"0b5e4bfab0082efe95ea6459071d78c1"** base\_url = **"https://api.openweathermap.org/data/2.5/weather?"** speak(**"weather of which city you want to know sir ??? "**)  
 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"**]  
 c\_t = float(round(current\_temperature)) - 273  
 current\_humidiy = y[**"humidity"**]  
 z = x[**"weather"**]  
 weather\_description = z[0][**"description"**]  
 speak(**" Temperature in celcius is "** +  
 str(c\_t) +  
 **"\n humidity in percentage is "** +  
 str(current\_humidiy) +  
 **"\n description "** +  
 str(weather\_description))  
 print(**" Temperature in celcius = "** +  
 str(c\_t) +  
 **"\n Humidity (in percentage) = "** +  
 str(current\_humidiy) +  
 **"\n Description = "** +  
 str(weather\_description))  
  
 **else**:  
 **pass  
  
 elif 'news related to india' in** query:  
 news = webbrowser.open\_new\_tab(**"https://timesofindia.indiatimes.com/home/headlines"**)  
 speak(**'Here are some headlines from the Times of India,Happy reading'**)  
 time.sleep(6)  
  
 **elif "day" in** query:  
 tellDay()  
  
 **elif "ganpati bappa morya" in** query:  
 speak(**"pud-chya war-shee lav-kar yaa"**)  
 print(**"Pudh Chya Varsi Laykar Ya"**)  
 notification.notify(  
 title=**"JID Message"**,  
 message=**"Pudh Chya Varsi Laykar Ya"**,  
 timeout=3  
 )  
  
 **elif "open notepad" in** query:  
 speak(**"opening notepad sir"**)  
 npath = **"C:\\ProgramData\\Microsoft\\Windows\\Start Menu\\Programs\\Accessories\\Notepad"** os.startfile(npath)  
  
 **elif "open teams" in** query **or "open Microsoft teams" in** query:  
 mpath = **"C:\\Users\\KIRAN\\Desktop\\Microsoft Teams"** os.startfile(mpath)  
  
 **elif "open Studio" in** query **or "open o b s studio " in** query:  
 opath = **"C:\\Users\\Public\\Desktop\\OBS Studio.exe"** os.startfile(opath)  
  
 **elif "open zoom" in** query:  
 zpath = **"C:\\Users\\KIRAN\\AppData\\Roaming\\Zoom\\bin\\Zoom.exe"** os.startfile(zpath)  
  
 **elif "open telegram" in** query:  
 tpath = **"C:\\Users\\KIRAN\\AppData\\Roaming\\Telegram Desktop\\Telegram.exe"** os.startfile(tpath)  
  
 **elif "where i am" in** query **or "locate my position" in** query:  
 speak(**"wait sir, let me check !!!"**)  
 ip = requests.get(**'http://api.ipify.org/'**).text  
 location = geocoder.ip(ip)  
 loc = pytz.country\_names[location.country]  
 looc = location.city  
 speak(**f"i am not sure sir but i think you are in {**looc**} city of {**loc**}"**)  
 notification.notify(  
 title=**"LOCATION"**,  
 message=**f"City = {**looc**}\nCountry = {**loc**}"**,  
 timeout=3  
 )  
 print(location.city, pytz.country\_names[location.country])  
 print(**" "**)  
  
 **elif "who made you" in** query:  
 speak(**"Sir Ayyush Bad gujar made me with his soft hands by pressing the keyboard to type my extraordinary codes..."**)  
 notification.notify(  
 title=**"CREATOR INFO"**,  
 message=**"Creator : Ayush Badgujar"** )  
  
 **elif "take screenshot" in** query:  
 speak(**"Sir please tell me the name for this screenshot file"**)  
 name = takeCommand().lower()  
 speak(**"please sir hold the screen for few seconds, i am taking screenshot..."**)  
 time.sleep(3)  
 img = pyautogui.screenshot()  
 img.save(**f"{**name**}.png"**)  
 speak(**"i am done sir, the screenshot is saved in our main folder, now i am ready for next command"**)  
  
 **elif "your favourite animal" in** query:  
 speak(**"Uuh..., my favorite animal is dolphine, as they are cute ofcourse , they are beautiful, swift and very friendly when they come in contact with anyone, and the best quality they have is to jump in and out of water."**)  
 notification.notify(  
 title=**"Favourite animal"**,  
 message=**"Dolphine"**,  
 timeout=4  
 )  
  
 **elif "your favourite colour" in** query:  
 speak(**"my favorite colour umm... let me think , oh yes ,white, you would ask why , so white is mixer of many colours and so if your favorite colour is red , blue , green etc i include all of them... isnt it cool...!!!"**)  
 notification.notify(  
 title=**"Favourite colour"**,  
 message=**"WHITE"**,  
 timeout=4  
 )  
  
 **elif "your favourite food" in** query:  
 speak(**"veg or non veg , i will tell you both still, in veg i like electricity made from nuclear reactors and in non veg i like electricity made from electric eels..."**)  
  
 **elif 'covid stats' in** query **or "corona" in** query:  
 html\_data = make\_request(**'https://www.worldometers.info/coronavirus/'**)  
 *# print(html\_data)* soup = BeautifulSoup(html\_data, **'html.parser'**)  
 total\_global\_row = soup.find\_all(**'tr'**, {**'class'**: **'total\_row'**})[-1]  
 total\_cases = total\_global\_row.find\_all(**'td'**)[2].get\_text()  
 new\_cases = total\_global\_row.find\_all(**'td'**)[3].get\_text()  
 total\_recovered = total\_global\_row.find\_all(**'td'**)[6].get\_text()  
 print(**'Total cases : '**, total\_cases)  
 print(**'New cases'**, new\_cases[1:])  
 print(**'Total recovered'**, total\_recovered)  
 notification\_message = **f" Total cases : {**total\_cases**}\n New cases : {**new\_cases[1:]**}\n Total Recovered : {**total\_recovered**}\n"** notification.notify(  
 title=**"COVID-19 Statistics"**,  
 message=notification\_message,  
 timeout=5  
 )  
 speak(**"here are the stats for COVID-19"**)  
  
 **elif "what is my name" in** query **or "my name" in** query:  
 notification\_message = **f"Oh! Sir you only told me your name\n {**username1**}"** notification.notify(  
 title=**"IDENTITY"**,  
 message=notification\_message,  
 timeout=5  
 )  
 speak(**f"Oh! Sir you only told me your name\n {**username**}"**)  
  
 **else**:  
 speak(**"I didnt get that sir can u repeat"**)  
  
  
**if** \_\_name\_\_ == **"\_\_main\_\_"**:  
 wishMe()  
 usrname()  
 **while True**:  
 query = takeCommand().lower()  
  
 **if "bye" in** query **or "shutdown" in** query:  
 speak(**"Okay sir"**)  
 speak(**"Shutting down...."**)  
 print(**"Shutting down...."**)  
 notification.notify(  
 title=**"JID's Message"**,  
 message=**"JID is shutted down\nso you will have to \nrerun the program to \nuse it."**,  
 timeout=5  
 )  
 time.sleep(3)  
 winsound.PlaySound(**"SystemExit"**, winsound.SND\_ALIAS)  
 **break  
  
 elif "are you there" in** query:  
 winsound.PlaySound(**"SystemStart"**, winsound.SND\_ALIAS)  
 Mains\_code()

OUTPUT

C:\Users\KIRAN\PycharmProjects\JID\_Type\_1\venv\Scripts\python.exe C:/Users/KIRAN/PycharmProjects/JID\_Type\_1/venv/JID\_Type\_1.py

pygame 2.0.1 (SDL 2.0.14, Python 3.9.0)

Hello from the pygame community. https://www.pygame.org/contribute.html

<Voice id=HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Speech\Voices\Tokens\TTS\_MS\_EN-US\_ZIRA\_11.0

name=Microsoft Zira Desktop - English (United States)

languages=[]

gender=None

age=None>

Listening...

Recognizing...

User said: Ayush

##################################################################################################################################

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Welcome AYUSH !!! \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

##################################################################################################################################

Listening...

Recognizing...

User said: are you there

Listening...

Recognizing...

User said: who is Amitabh Bachchan

<generator object search at 0x0000020EB675EEB0>

Listening...

Recognizing...

User said: are you there

Listening...

Recognizing...

User said: divide 110 by 11

10.0

Listening...

Recognizing...

User said: are you there

Listening...

Recognizing...

User said: multiply 15 and 16

240.0

Listening...

Recognizing...

User said: are you there

Listening...

Recognizing...

User said: play Doraemon theme song

Listening...

Recognizing...

User said: are you there

Listening...

Recognizing...

User said: tell me the time

21:51 PM

Listening...

Recognizing...

User said: are you there

Listening...

Recognizing...

User said: tell me a joke

If you put a million monkeys at a million keyboards, one of them will eventually write a Java program. The rest of them will write Perl.

Listening...

Recognizing...

User said: no

Listening...

Recognizing...

User said: are you there

Listening...

Recognizing...

User said: how are you

Listening...

Recognizing...

User said: are you there

Listening...

Recognizing...

User said: sine function

Listening...

Recognizing...

User said: 90

Sine of 90.0 radians is 1.0

Listening...

Recognizing...

User said: are you there

Listening...

Recognizing...

User said: weather

Listening...

Recognizing...

User said: Lahore

Temperature in celcius = 15.0

Humidity (in percentage) = 67

Description = smoke

Listening...

Recognizing...

User said: are you there

Listening...

Recognizing...

User said: locate my position

Mumbai India

Listening...

Recognizing...

Can you please repeat sir ??

Listening...

Recognizing...

User said: are you there

Listening...

Recognizing...

User said: what is my name

Listening...

Recognizing...

User said: shutdown

Shutting down....

MODULES USED

| MODULE NAME | Code to type in the terminal | |
| --- | --- | --- |
| pyttsx3 | | pip install pyttsx3 |
| speech\_recognition | | pip install SpeechRecognition |
| datetime | | need not to be installed |
| pyaudio | | pip install PyAudio |
| wikipedia | | pip install wikipedia |
| pywhatkit | | pip install pywhatkit |
| pyjokes  googlesearch  math os  wolframalpha  shutil | | pip install pyjokes  pip install google-search  need not to be installed  pip install os-sys  pip install wolframalpha  pip install pytest-shutil |
| tones  webbrowser  requests  time  geocoder  pyautogui | | pip install tones  pip install webbrowser  pip install requests  pip install time  pip install geocoder  pip install PyAutoGUI |
| winsound | | need not to be installed |
| playsound  pygame  random  tkinter  pytz | | pip install playsound  pip install pygame  pip install random  pip install tkinter  pip install pytz |

FEATURES

JID can do a lot of cool things, some of them being:

* Greet the user
* Tell current time and date
* Launch applications/software
* Open any website
* Tell us about weather of any city
* Tell us our current location.
* Tell us about any person (via Wikipedia)
* search anything on Google
* play any song on YouTube
* Tell us top headlines (via Times of India)
* Play music
* Open Gmail
* Calculate simple mathematical expressions (example: 3+4, 5\*6, 7-2 or 16/8)
* Answer any generic question (via Wolfram alpha)
* Take important note in notepad
* Tell us a random joke
* Can take screenshot and save it with custom filename
* Tell us covid stats
* Can calculate values of tan, sin and cos functions

And much more…

CONCLUSION

While this assistant is very basic and only is a small demonstration, it represents the beginning of something big. As technology continues to advance, the very basic stage we are at now will evolve into a more complex network of assistants and smart objects working together to make the world more efficient, safe, and sustainable.

As mentioned before, AI assistants are part of the internet of things (IoT). With continued advances in this field, humanity will be able to accomplish more in the same amount of time. IoT is being used in many fields ranging from residential to commercial. Although this technology has a long way to go, once it is ready, it will revolutionize the world.

Till then we have to keep working hard and we have to keep doing such things. And hope that we will be able to make a completely personalized assistant capable of doing numerous things that even a human can’t do hence forming a better and smart world.

BIBLOGRAPHY

1. <https://medium.com/lets-talk-ml/guide-to-your-very-own-a-i-virtual-assistant-in-python-d8120ff228a8>
2. <https://www.activestate.com/blog/how-to-build-a-digital-virtual-assistant-in-python/>
3. <https://towardsdatascience.com/how-to-build-your-own-ai-personal-assistant-using-python-f57247b4494b>
4. <https://dev.to/rohit19060/how-to-create-a-virtual-assistant-using-python-4h7l>
5. <https://en.wikipedia.org/wiki/Virtual_assistant>
6. <https://alan.app/blog/voiceassistant-2/>
7. <https://www.smartsheet.com/voice-assistants-artificial-intelligence>
8. <https://programminghistorian.org/en/lessons/installing-python-modules-pip>
9. <https://pythonprogramminglanguage.com/how-to-install-modules/>
10. <https://chatbotsjournal.com/what-benefits-in-a-personal-voice-assistant-technology-5365b31b3637>
11. <https://whatis.techtarget.com/definition/voice-assistant#:~:text=Voice%20control%2C%20also%20called%20voice,all%20processing%20is%20done%20locally>

HOW TO DOWLOAD PYTHON AND PYCHARM

1. <https://www.journaldev.com/30076/install-python-windows-10>
2. <https://www.youtube.com/watch?v=QzcaEELafkE>

THANKYOU