VOICE ASSISTANT
(USING PYTHON)

INDUSTRIAL TRAINING PROJECT REPORT

Submitted in partial fulfilment of the requirements for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

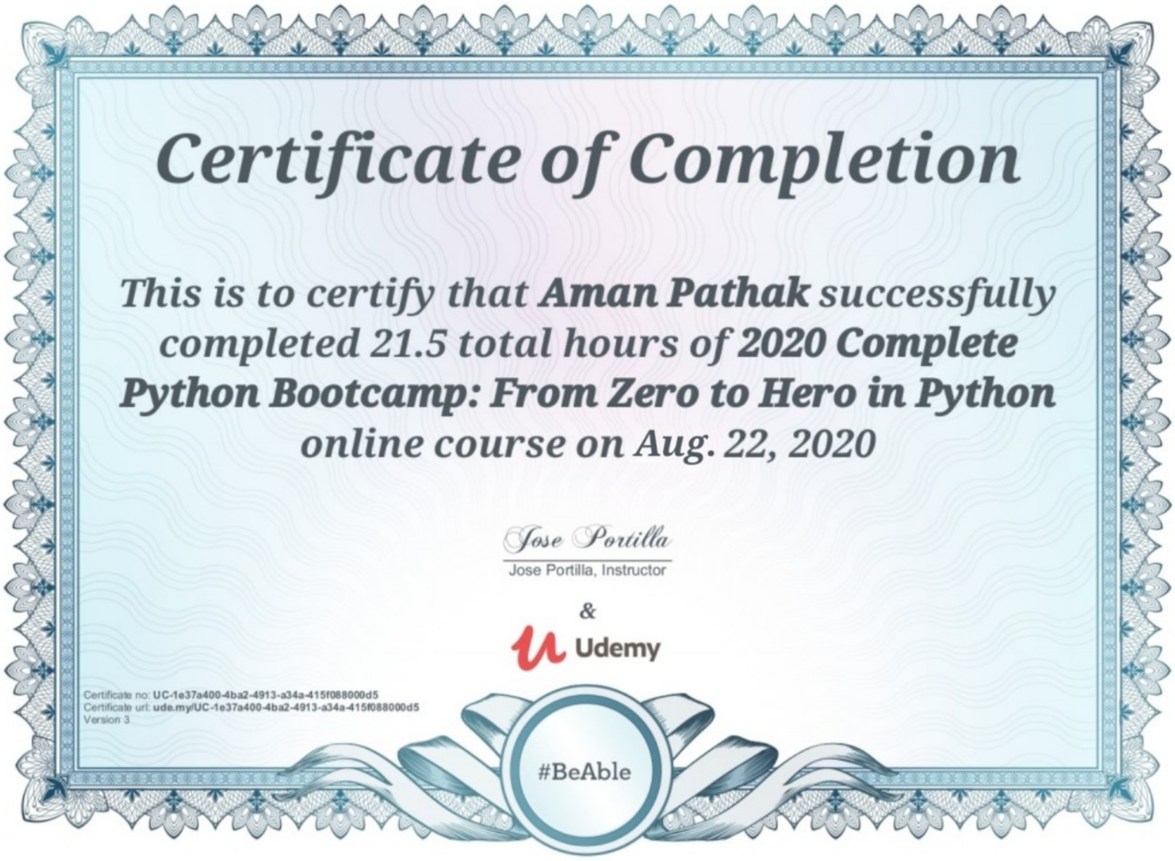
INFORMATION TECHNOLOGY

by

AMAN PATHAK
Enrolment No: 01315603118

DEPARTMENT OF INFORMATION TECHNOLGY
DR. AKHILESH DASS GUPTA INSTITUTE OF TECH & MGMT
(AFFILIATED TO GURU GOBIND SINGH INDRAPRASTHA UNIVERSITY, DELHI)
NEW DELHI – 110053

CERTIFICATE OF TRAINING



Certificate of Completion

This is to certify that **Aman Pathak** successfully completed 21.5 total hours of **2020 Complete Python Bootcamp: From Zero to Hero in Python** online course on Aug. 22, 2020

Jose Portilla

Jose Portilla, Instructor

&

Udemy

Certificate no: **UC-1e37a400-4ba2-4913-a34a-415f088000d5**
Certificate url: **ude.my/UC-1e37a400-4ba2-4913-a34a-415f088000d5**
Version 3

#BeAble

ABSTRACT

Industrial training is an important phase of a student life. A well planned, properly executed and evaluated industrial training helps a lot in developing a professional attitude. It develops an awareness of industrial approach to problem solving, based on a broad understanding of process and mode of operation of organization. Besides software training, it builds self confidence among students and let students know the technical knowledge and professionalism. The aim and motivation of this industrial training is to receive discipline, skills, teamwork and technical knowledge through a proper training environment, which will help me, as a student in the field of Information Technology, to develop a responsiveness of the self-disciplinary nature of problems in information and communication technology. Throughout this industrial training, I have been learned new libraries of python programming language that were required for the system, the process of the production lines and able to implement what I have learnt for the past year as a Bachelor in Information Technology student at ADGITM, GGSIP University, Dwarka, Delhi. During the development of this Project, most of the theoretical knowledge gained during the course of studies was put to test, Various effort and processes involved in designing of a various logic was studied and understood during the training. During this, I undertook courses – "Python Bootcamp –from zero to hero" from josh portilla offered by UDEMY.. As of today, many people suffer from Visual Impairment. It one of the biggest problem's humankind is facing today. Frequent assistance is required for them to be able to perform day to day tasks. Earlier Braille type of paper was used to read. Although it's still in use today it is gradually becoming outdated as it is restricted to only reading and cannot suffice the needs and perform other tasks of the visually impaired. In today's technologically advanced world Computer Vision based solutions and android based solutions are emerging as one of the most promising due to their accessibility and the technology used which enables them to perform a wide range of tasks. This paper discusses the latter solution. The solution aims to create an application for visually impaired in which text and speech commands are accepted from the user xiv and the tasks are performed. The application is created by using PYTHON programming language. The training gave me a good experience from the view of implementing my theoretical knowledge in practical aspects. It gave me first-hand experience of working as an engineering professional. It helped me in improving my technical, interpersonal and communication skills, both oral and written. Overall, it is such a great experience to have industrial training and I truly believe that it will help me in building a successful career.

ACKNOWLEDGEMENT

I take this opportunity to express my profound gratitude and deep regards to my Instructor **Prof. Josh Portille** and online educational website **UDEMY** for their exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by them time to time shall carry me a long way in the journey of life on which I am about to embark.

I take this opportunity to thank the **ADGITM** college for giving me chance to do this project.

Lastly, I would like to thank each and every person who directly or indirectly helped me in the completion of the project especially my Parents and Peers who supported me throughout my project.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

1.1 About Python

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum during 1985- 1990. Like Perl, Python source code is also available under the GNU General Public License (GPL).

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

Python is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain. I will list down some of the key advantages of learning Python:

- **Python is Interpreted** − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive** − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented** − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- **Python is a Beginner's Language** − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

1.1.1 What PYTHON implementations are available?

An "implementation" of Python should be taken to mean a program or environment which provides support for the execution of programs written in the Python language, as represented by the CPython reference implementation.

There have been and are several distinct software packages providing what we all recognize as Python, although some of those are more like distributions or variants of some existing implementation than a completely new implementation of the language.

CPython Variants

These are implementations based on the CPython runtime core (the de-facto reference Python implementation), but with extended behaviour or features in some aspects.

- CrossTwine Linker - a combination of CPython and an add-on library offering improved performance (currently proprietary)
- Stackless Python - CPython with an emphasis on concurrency using tasklets and channels (used by dspython for the Nintendo DS)
- unladen-swallow - "an optimization branch of CPython, intended to be fully compatible and significantly faster", originally considered for merging with CPython subject to PEP 3146, but now unmaintained
- wpython - a re-implementation of CPython using "wordcode" instead of bytecode
- eGenix PyRun - Python runtime (CPython + std library) compressed into a single 3-4MB binary

Reduced CPython Variants

These provide a subset of the full language + standard library, for use in embedded scenarios (see also the EmbeddedPython topic)

- ❖ Micropython - Python for microcontrollers (runs on the pyboard and the BBC Microbit)
  - Acro python - Adafruit's branch of MicroPython

Also look at the sections on Python compilers and extensions below, some of which would qualify as CPython variants.

Other Implementations

These are re-implementations of the Python language that do not depend on (or necessarily interact with) the CPython runtime core. Many of them reuse (a large part of) the standard library implementation.

Note that most of these projects have not yet achieved language compliance. However, many of these have goals and features or run in certain environments that make them interesting in their own regard. The only implementations that are known to be compatible with a given version of the language are IronPython, Jython and PyPy.

- **IronPython** (Python running on .NET)
- **Jython** (Python running on the Java Virtual Machine)
- **PyPy** (A fast python implementation with a JIT compiler)
- Stackless Python (Branch of **CPython** supporting microthreads)

- MicroPython (Python running on micro controllers)

1.1.2 PYTHON resources

1. Udemy

Udemy provides you the best platform to learn the Python language. It offers Python courses from beginner to expert level. You can learn both versions, Python 2 and Python 3 with Udemy. It also teaches you about the advanced features of Python, including a collection module, decorators, and timestamps.

2. Learn Python the Hard Way

"Learn Python the Hard Way" is the most popular way to **get started with the Python programming language.** You are not required to have prior experience in coding. This tutorial will teach you from level 0 to higher levels. It is an open source and free tutorial, available online for a refresher as well as professional programmers.

3. Codecademy

Codecademy is an emerging online learning platform, which emphasizes improving the overall learning experience. It offers free courses on the Python programming language. It also provides a free code editor so that users can practice writing code, and a forum to discuss queries with friends and other members.

4. Python.org

Python.org contains the official documentation of Python programming language. It helps both beginners and experienced programmers to learn about Python. It is one of the free online Python tutorials.
Under Python.org, you will get a lot of references and materials to work with the language. It will teach you right from the basic concepts. And later all the top to advance level.

1.2What is VOICE ASSISTANT?

.A voice assistant is a digital assistant that uses voice recognition, speech synthesis, and natural language processing (NLP) to provide a service through an application. A Pew research centre Survey in May 2017 showed that nearly half of all adults in the United States use Voice-controlled digital assistants on their smartphones and other devices. The rise of automation, along with increased computational power, novel application of statistical algorithms, and improved accessibility to data, have resulted in the birth of the personal digital assistant market, popularly represented by Apple's Siri, Microsoft's Cortana, Google's Google Assistant, and Amazon's Alexa. While each assistant may specialize in slightly different tasks, they all seek to make the user's life easier through verbal interactions so you don't have to search out a keyboard to find answers to questions like "What's the weather today?" or "Where is Switzerland?". Despite the inherent "cool" factor that comes with using a digital assistant, you may find that a forementioned digital assistants do not cater to your specific needs.

FIGURE 1.1.VOICE ASSISTANT

1- FEATURES

1.1 Speech to Text: It converts real-time audio into recognized text using google speech-to-text from the speech service. The text is available, as its transcribed, to both your assistant implementation and your client application.

1.2 Text to Speech: Textual responses from your assistant are synthesized using text-to-speech from the speech service, the synthesis is then made available to your client application as an audio stream.

The overwhelming success of speech-enabled products like Amazon Alexa has proven that some degree of speech support will be an essential aspect of household technology for the foreseeable future. In other words, speech-enabled products would be a game changer as that offer a level of interactivity and accessibility that few technologies can match.

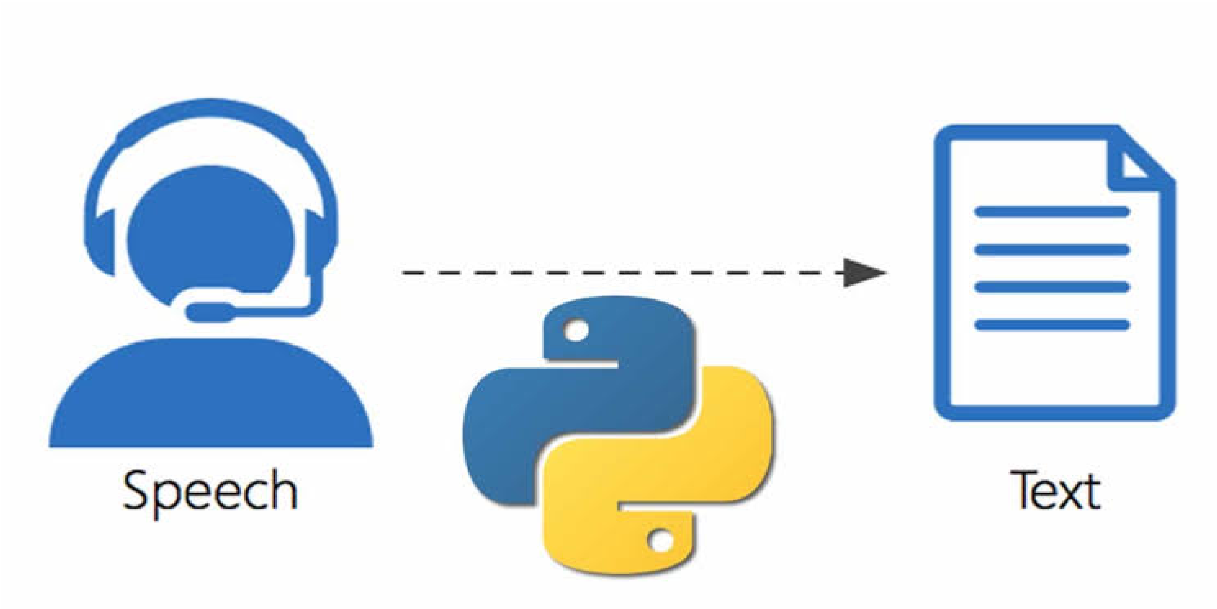


FIGURE1. 2. SPEECH TO TEXT

In other words, speech-enabled products would be a game changer as that offer a level of interactivity and accessibility that few technologies can match.

No GUI needed.

No texting needed.

It’s all about speed.

Speed is a big reason voice is poised to become the next major user interface. Each decade, we have embraced a new way to interact with technology. We have evolved from character mode to a graphical user interface, to the web, to mobile.

Voice now offers a faster and easier way to communicate and accomplish tasks than mobile apps.

We can either tell Alexa what we need (turn off the lights, adjust the thermostat, and set an alarm — or all of the above using a single utterance like “Alexa, good night”), or you can pull your phone, unlock it, open the right app, and perform the task or tasks.

“Texting will decline in the future because of Alexa”

— Gary Vaynerchuk

FIGURE 3.VOICES LET US SAY MORE FASTER

Therefore, that made me very interested in embarking on a new project to build a simple speech recognition with Python.

1.3 **Objective**

Voice Assistant project is a text to speech program which is used to search any query by giving audio command to the assistant and the assistant in return gives the result of the given input.

- It can send emails for you.
- It can play music for you.
- It can do Wikipedia searches for you.
- It is capable of opening websites like youtube,google,prime video etc in a web browser on a single line speech command.
- You can read latest news using this assistant.

CHAPTER 2: LITERATURE SERVEY

2.1 Analysis of Technology Used

2.1.1PYTHON

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.Python is used most often these days in programming world for making games as well as modern applications on mobiles.

Following are important characteristics of **Python Programming** −

- It supports functional and structured programming methods as well as OOP.
- It can be used as a scripting language or can be compiled to byte-code for building large applications.
- It provides very high-level dynamic data types and supports dynamic type checking.
- It supports automatic garbage collection.
- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

As mentioned before, Python is one of the most widely used language over the web. I'm going to list few of them here:

- **Easy-to-learn** − Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- **Easy-to-read** − Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** − Python's source code is fairly easy-to-maintain.
- **A broad standard library** − Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- **Portable** − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.

- **Extendable** − You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- **Databases** − Python provides interfaces to all major commercial databases.
- **GUI Programming** − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.

Scalable − Python provides a better structure and support for large programs than shell scripting.

.

2.1.1.1 How to Run PYTHON SCRIPT?

Here are the ways with which we can run a Python script.

1. Interactive Mode
2. Command Line
3. Text Editor (VS Code)
4. IDE (PyCharm)

1.Interactive Mode:

In Interactive Mode, you can run your script line by line in a sequence.

To enter in an interactive mode, you will have to open Command Prompt on your windows machine and type ‘python’ and press Enter.

2.Command Line

To run a Python script store in a ‘.py’ file in command line, we have to write ‘python’ keyword before the file name in the command prompt.

```
python hello.py
```

3.Text Editor (VS Code)

To run Python script on a text editor like VS Code (Visual Studio Code) then you will have to do the following:

- Go in the extension section or press ‘Ctrl+Shift+X’ on windows, then search and install the extension named ‘Python’ and ‘Code Runner’. Restart your vs code after that.

- Now, create a new file with the name ‘hello.py’ and write the below code in it:
- print('Hello World!')
- Then, right click anywhere in the text area and select the option that says ‘Run Code’ or press ‘Ctrl+Alt+N’ to run the code.

4.IDE (PyCharm)
To run Python script on a IDE (Integrated Development Environment) like PyCharm, you will have to do the following:

- Create a new project.
- Give a name to that project as ‘HELLO’ and click on Create.
- Select the root directory with the project name we specified in the last step. Right click on it, go in New and click on ‘Python file’ option. Then give the name of the file as ‘hello’ (you can specify any name as per your project requirement). This will create a ‘hello.py’ file in the project root directory.

2.1.1.2 Tools You Need

To start with, you need a text editor to write your code and a browser in this project to display the output of the command given by the user.You can use any text editor like VS code,IDE(PYCHARM).I have here used pycharm to write the required code. You can use any web browser including Google Chrome, Firefox, Microsoft Edge, Internet Explorer etc to display your output.

CHAPTER 3: METHODOLOGY USED

3.1 SOFTWARE REQUIREMENT SPECIFICATION(SRS)

3.1.1 Introduction to SRS

This Document layout a project plan for the development of "Voice Assistant for Blind person" open source repository by Aman Pathak. The intended readers are current and future developers working on this Voice Assistant, the plan will include, but is not restricted to, a summary of the system functionality, the scope of the project from my perspective.

3.2 **Overview**

The World Health Organization (WHO) estimates that 314 million people live with visual impairment. 217 million of those have moderate to severe vision impairment and 45 million are blind. Traditionally, Tools like stick were used from decades to avoid obstacles. Keeping pet dogs or guide dogs were one of few techniques used by the visually impaired. Some took help of their family and friends for assistance. However, these techniques and tools had drawbacks. People with trained dogs needed money and time to feed and train the dogs. In today's technologically advanced world the above techniques cannot suffice the needs of the visually impaired people. Smartphone today has become a must-have device. It performs multiple tasks like telling time, messaging, calling, etc. Tasks which visually impaired are in dire need of. Through the help of technology solutions can be created to rectify the problems visually impaired people face in day to day life. It would have a positive impact on their lives and make their day to day life easier.

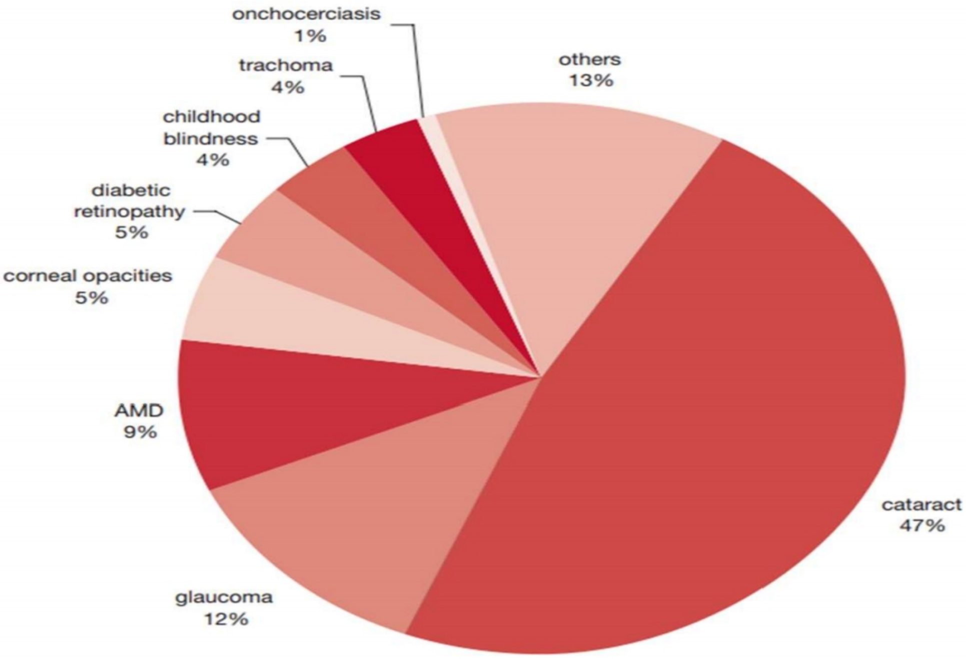
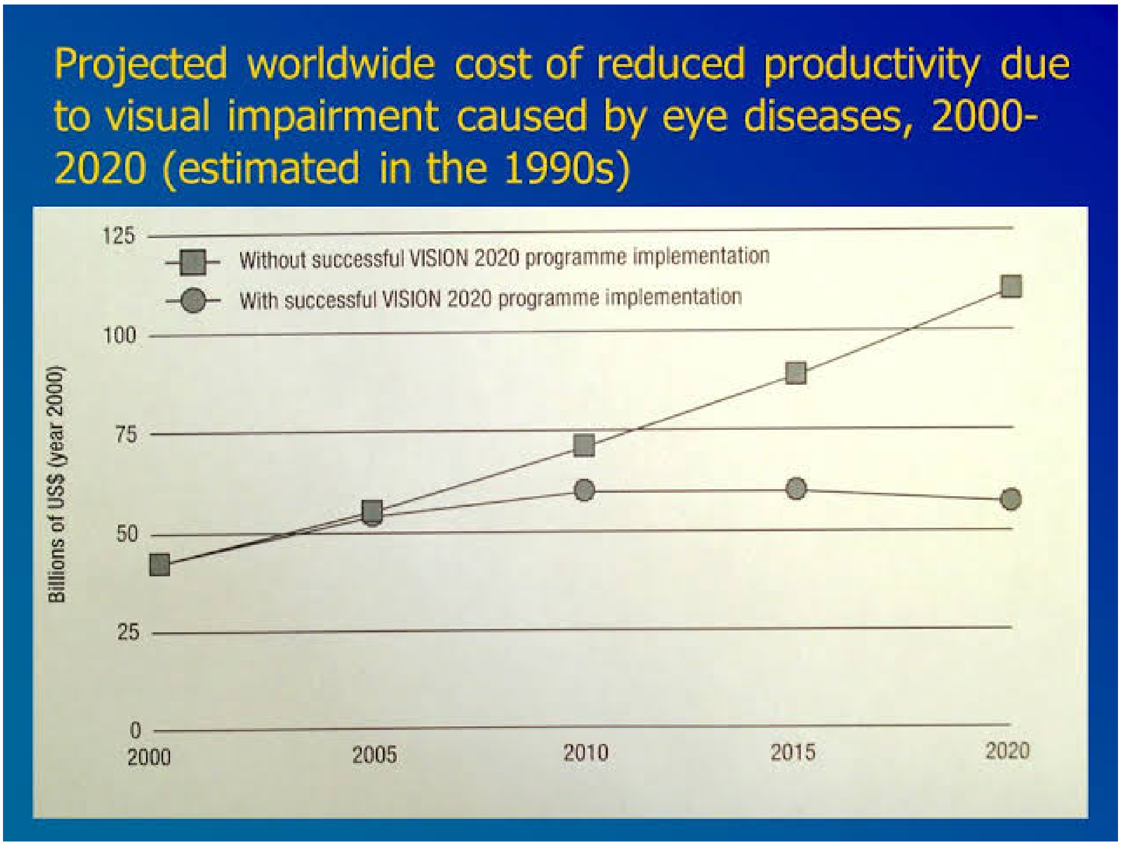


FIGURE2.1.GLOBAL CAUSE OF EYE DISEASES,EXCLUDING REFRACTIVE ERRORS

Voice Assistant are the most natural modality for people with vision impairment. These speakers really make sense for people with vision impairment and prove to be a perfect fit in their daily lives. Since, there is no screen, blind users interact on the same level as sighted people. So, it is not just about access, it is about equality. A blind person can set an alarm and play music just like the rest of us, but they can also send a text message or can construct an email.



Projected worldwide cost of reduced productivity due to visual impairment caused by eye diseases, 2000-2020 (estimated in the 1990s)

Fig2.2 projected worldwide cost of reduced productivity due to visual impairment caused by eye diseases, 2000 – 2020 (estimated in the 1990s)

3.3 GETTING STARTED WITH THE CODE

- HOW OUR VOICE ASSISTANT WORKS!

3.3.1 LIBRARIES AND MODULES USED:-

Firstly, we need two install and import two libraries, one for listening (google speech recognition) and one for responding (pyttsx3).

1.pyttsx3

1.1 Description

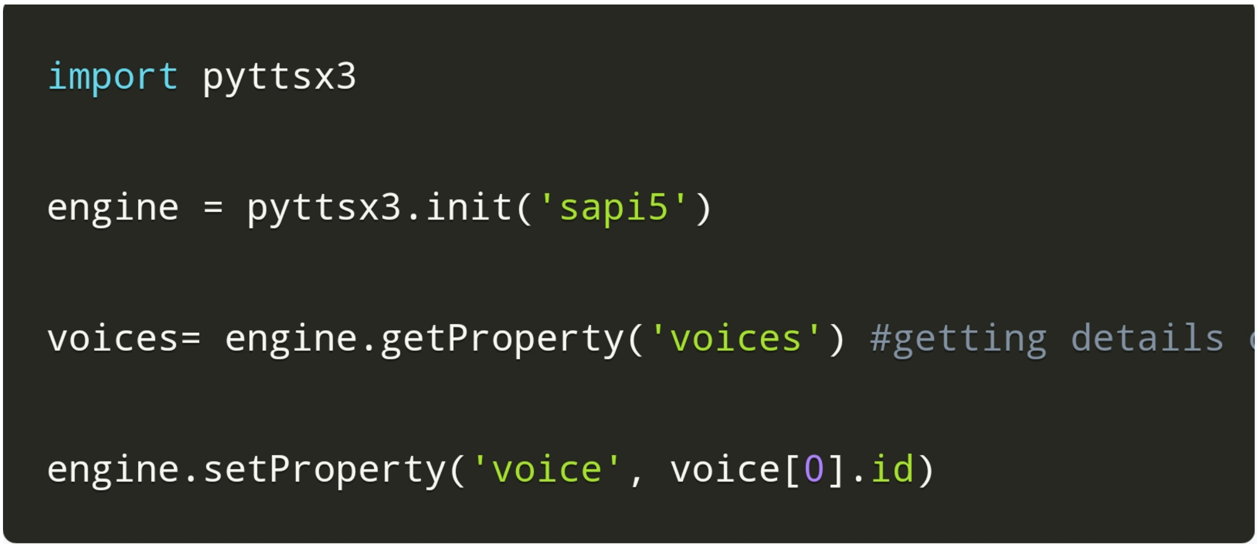
Pyttsx3 is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline, and is compatible with both Python 2 and 3

1.2Installation

```
Pip install pyttsx3
```

pyttsx3 installation If you receive errors such as No module named win32com.client, no module named win32, or No module named win32api, you will need to additionally install pypiwin32.

1.3Usage:



```
import pyttsx3

engine = pyttsx3.init('sapi5')

voices= engine.getProperty('voices') #getting details

engine.setProperty('voice', voice[0].id)
```

Figure 3.1pyttsx3.usage

What is Sapi5?

- Speech API developed by Microsoft.

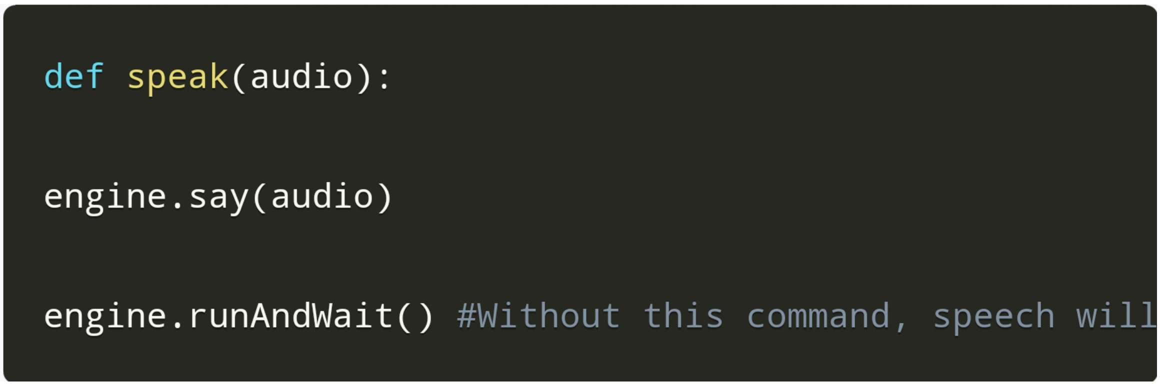
- Helps in synthesis and recognition of voice.

What Is VoiceId?

- Voice id helps us to select different voices.
- voice[0].id = Male voice
- voice[1].id = Female voice

2.*Writing Our speak() Function :*

We made a function called speak() at the starting of this tutorial. Now, we will write our speak() function so that it can convert our text to speech.



```
def speak(audio):

engine.say(audio)

engine.runAndWait() #Without this command, speech will
```

Figure 3.2speak() function

3.Creating our Main() function

Now, we will create a main() function, and inside this main() Function, we will call our speak function.

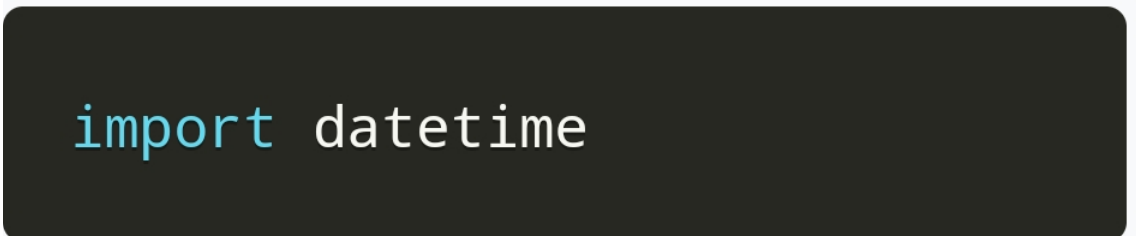
CODE:

```
if __name__=="__main__" :

speak("START SERACHING")
```

Whatever you will write inside this speak() function will be converted into speech. Congratulations! With this, our VOICE ASSISTANT has its own voice, and it is ready to speak.

4.Import datetime module:

Datetime module supplies classes to work with date and time. These classes provide a number of functions to deal with dates, times and time intervals.

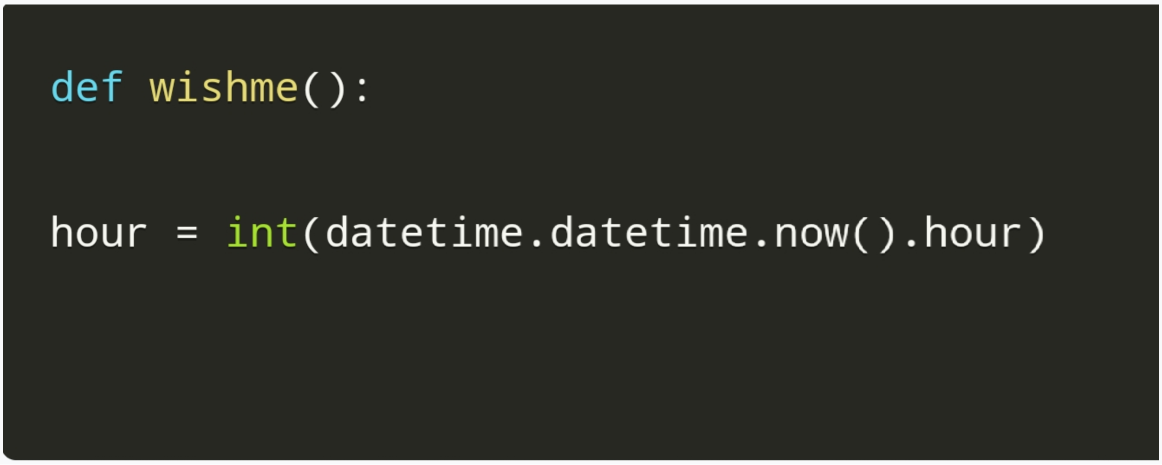


```
import datetime
```

FIGURE 3.3DATETIME MODULE

5.Defining Wish me Function :
Now, we are going to make a **wishme()** function, that will make our Assistant wish or greet the user according to the time of computer or pc. To provide current or live time to A.I., we need to import a module called datetime which we did in previous step.

Now, let's start defining the **wishme()** function:



```
def wishme():

hour = int(datetime.datetime.now().hour)
```

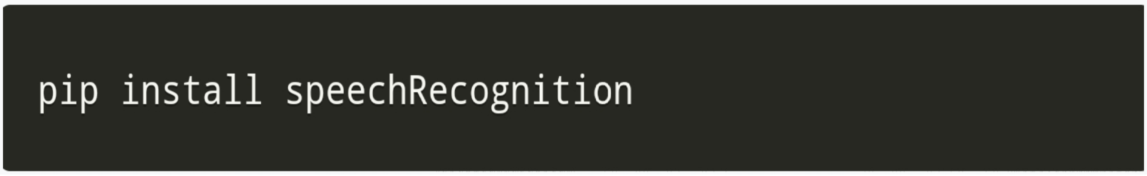
FIGURE 3.4 WISH ME FUNCTION DEFINED

Here, we have stored the integer value of the current hour or time into a variable named hour. Now, we will use this hour value inside an if-else loop.

6.Defining Take command Function :
The next most important thing for our A.I. assistant is that it should be able to take command with the help of the microphone of the user's system. So, now we will make

a **takeCommand()** function. With the help of the takeCommand() function, our A.I. assistant will be able to return a string output by taking microphone input from the user.

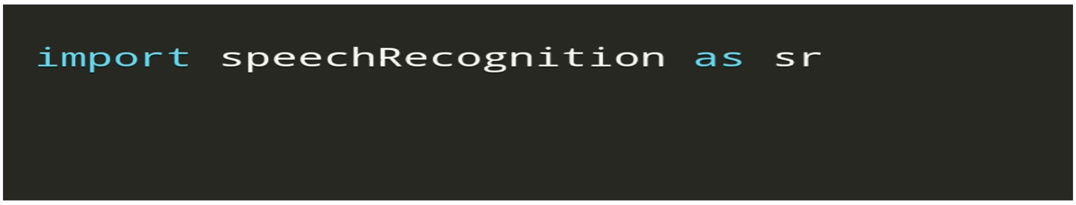
Before defining the takeCommand() function, we need to install a module called **speechRecognition.** Install this module by:



```
pip install speechRecognition
```

FIGURE3. 5 INSTALLED SPEECH RECOGINITION MODULE

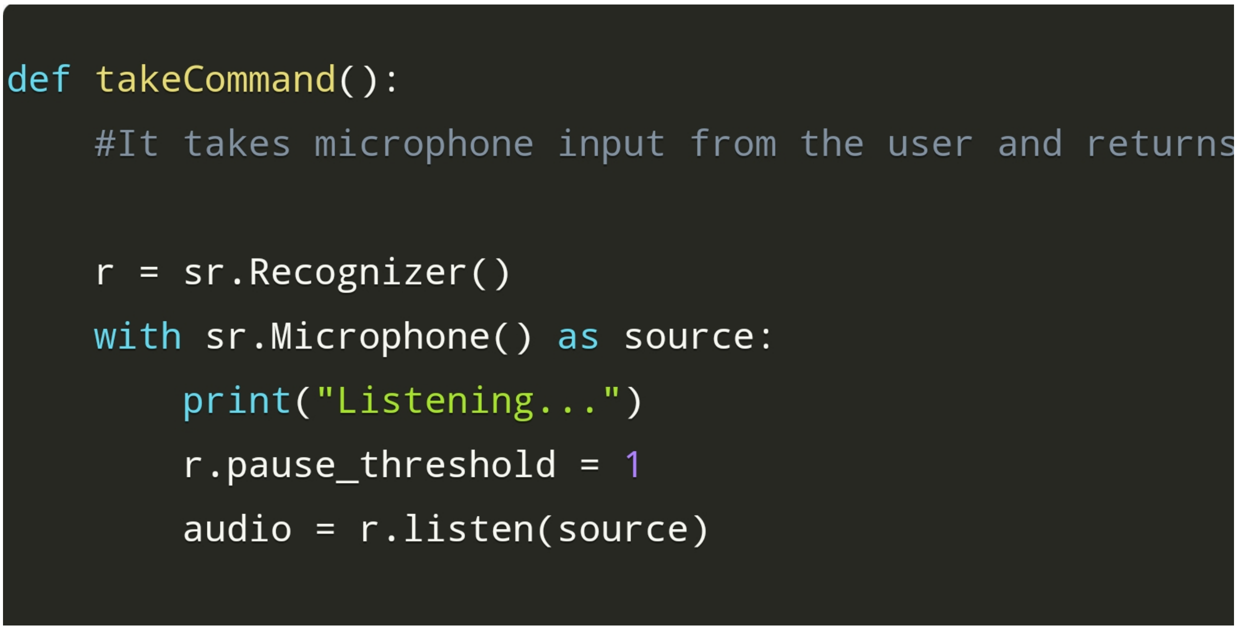
After successfully installing this module, import this module into the program by writing an import statement.



```
import speechRecognition as sr
```

FIGURE3. 6 SPEECH RECOGNITION SR IMPORTED INTO THE PROGRAM

The takeCommand() function is coded as:



```
def takeCommand():
    #It takes microphone input from the user and returns

    r = sr.Recognizer()
    with sr.Microphone() as source:
        print("Listening...")
        r.pause_threshold = 1
        audio = r.listen(source)
```

FIGURE3. 7TAKE COMMAND FUNCTION

We have successfully created our takeCommand() function. Now we are going to add a try and except block to our program to handle errors effectively.

We have added a try and exception block so that in case program runs into an error it doesn't abruptly gets stopped and the user can again give the command.

```
try:
        print("Recognizing...")
        query = r.recognize_google(audio, language='en-in') #Using google for voice recognition.
        print(f"User said: {query}\n")  #User query will be printed.

    except Exception as e:
        # print(e)
        print("Say that again please...")   #Say that again will be printed in case of improper voice
        return "None" #None string will be returned
    return query
```

#Coding logic of Assistant

Now, we will develop logics for different commands such as Wikipedia searches, playing music, etc.

Defining Task 1: To search something on Wikipedia

To do Wikipedia searches, we need to install and import the Wikipedia module into our program. Type the below command to install the Wikipedia module :

```
                Pip install wikipedia
```

After successfully installing the Wikipedia module, import it into the program by writing an import statement.

```
if __name__ == "__main__":
    wishMe()
    while True:
    # if 1:
        query = takeCommand().lower() #Converting user query into lower case

        # Logic for executing tasks based on query
        if 'wikipedia' in query:  #if wikipedia found in the query then this block will be executed
        speak('Searching Wikipedia...')
        query = query.replace("wikipedia", "")
        results = wikipedia.summary(query, sentences=2)
        speak("According to Wikipedia")
        print(results)
        speak(results)
```

fig 3.8 :Wikipedia query

In the above code, we have used an if statement to check whether Wikipedia is in the search query of the user or not. If Wikipedia is found in the user's search query, then two sentences from the summary of the Wikipedia page will be converted to speech with the help of speak function.

Defining Task 2: To open YouTube site in a web-browser

To open any website, we need to import a module called webbrowser. It is an in-built module, and we do not need to install it with pip statement, we can directly import it into our program by writing an import statement.

Code:

```
    elif 'open youtube' in query:
        webbrowser.open("youtube.com")
```

Here, we are using the elif loop to check whether the Youtube is in the query of the user or not. Let' suppose, the user gives command as " open youtube." So, open youtube will be in the user's query, and the elif condition will be true.

Defining Task 3: To open Google site in a web-browser

```
elif 'open google' in query:
        webbrowser.open("google.com")
```

We are opening Google in a web-browser by applying the same logic that we used to open youtube.

Task 4 and Task 5: To play Music and read latest news we have used the same logic as for opening youtube and google just by specifying their path.

Defining Task 6: To know the current time

```
  elif 'the time' in query:
        strTime = datetime.datetime.now().strftime("%H:%M:%S")
        speak(f"Sir, the time is {strTime}")
```

In the above, code with are using datetime() function and storing the current or live of the system into a variable called strTime. After storing the time in strTime, we are passing this variable as an argument in speak function. Now, the time string will be converted into the speech.

Defining Task 7: To send Email

To send an email, we need to import a module called smtplib.

What is smtplib?

- Simple Mail Transfer Protocol (SMTP) is a protocol that allows us to send emails and to route emails between mail servers. An instance method called **sendmail** is present in the SMTP module. This instance method allows us to send an email. It takes 3 parameters:
- **The sender:** Email address of the sender.
- **The receiver:**T Email of the receiver.
- **The *message:*** A string message which needs to be sent to one or more than one recipient.

#Defining Send email function :

Now, we will create a **sendEmail()** function, which will help us to send emails to one or more than one recipients.

```
def sendEmail(to, content):
    server = smtplib.SMTP('smtp.gmail.com', 587)
    server.ehlo()
    server.starttls()
    server.login('youremail@gmail.com', 'your-password')
    server.sendmail('youremail@gmail.com', to, content)
    server.close()
```

In the above code, we are using the SMTP module, which we have already discussed above.

Note: Do not forget to ***'enable the less secure apps'*** feature in your Gmail account. Otherwise, the sendEmail function will not work properly.

Calling sendEmail() function inside the main() function:

```
elif 'email to Aman' in query:
        try:
            speak("What should I say?")
            content = takeCommand()
            to = "amanyourEmail@gmail.com"
            sendEmail(to, content)
            speak("Email has been sent!")
        except Exception as e:
            print(e)
            speak("Sorry sir. I am not able to send this email")
```

We are using the try and except block to handle any possible error that can occur while sending emails.

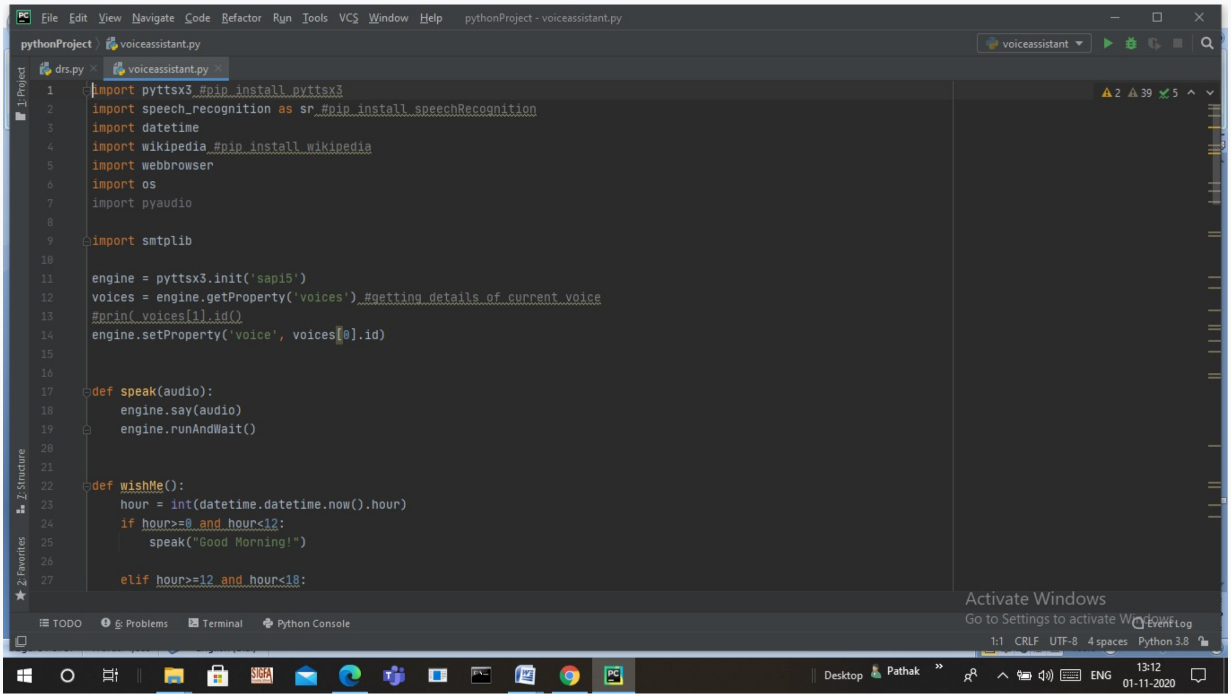
3.3.2 *Is it an A.I.?*

A lot of people will argue that the virtual assistant that we have created is not an A.I, but it is the output of the bunch of the statement. But, if we look at the very basic level, the sole purpose of A.I is to develop machines that can perform human tasks with the same effectiveness or even more effectively than humans.

It is a fact that our virtual assistant is not a very good example of A.I., but it is an A.I. !

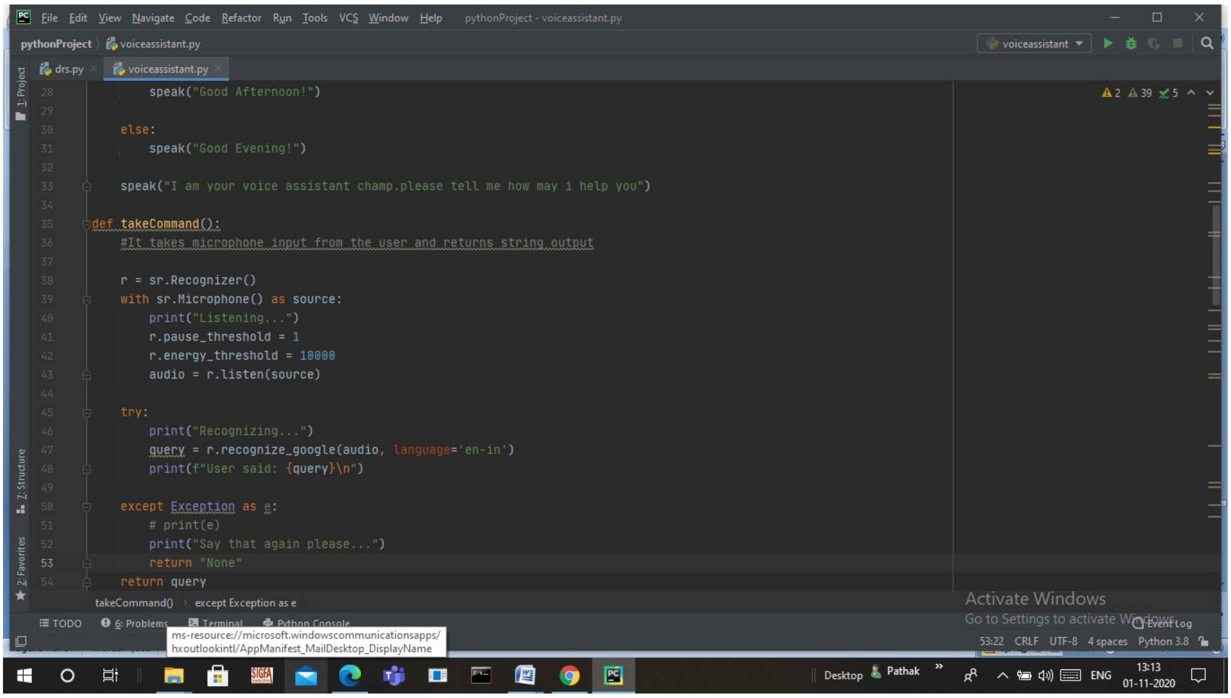
THUS,using the above modules and command we are able to create our Voice Assistant which is capable of providing the speech to text output for human beings specially designed for visually impaired.

3.3.3 COMPLETE CODE FOR THE VOICE ASSISTANT


File Edit View Navigate Code Refactor Run Tools VCS Window Help
pythonProject - voiceassistant.py
pythonProject voiceassistant.py
drs.py
voiceassistant.py
TODO
6: Problems
Terminal
Python Console
Activate Windows
Go to Settings to activate Windows.
Event Log
1:1 CRLF UTF-8 4 spaces Python 3.8
Desktop
Pathak
ENG
13:12
01-11-2020

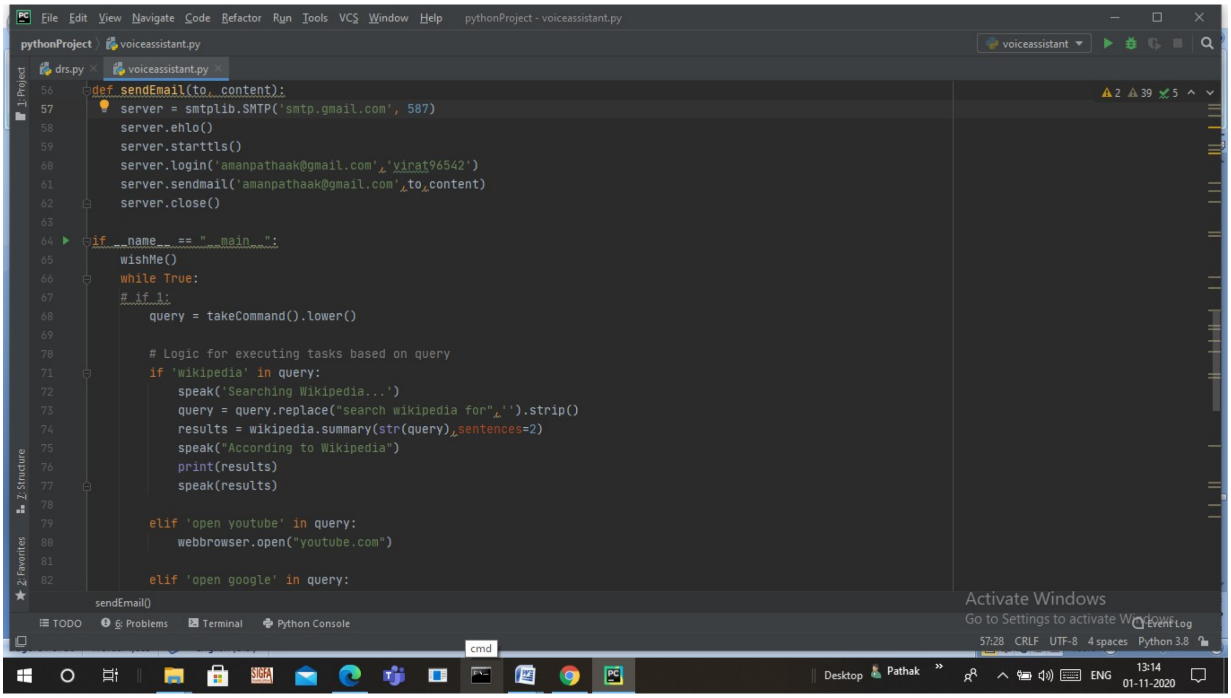
```python
import pyttsx3 #pip install pyttsx3
import speech_recognition as sr #pip install speechRecognition
import datetime
import wikipedia #pip install wikipedia
import webbrowser
import os
import pyaudio

import smtplib

engine = pyttsx3.init('sapi5')
voices = engine.getProperty('voices') #getting details of current voice
#prin( voices[1].id()
engine.setProperty('voice', voices[0].id)


def speak(audio):
    engine.say(audio)
    engine.runAndWait()


def wishMe():
    hour = int(datetime.datetime.now().hour)
    if hour>=0 and hour<12:
        speak("Good Morning!")

    elif hour>=12 and hour<18:
```

FIGURE3. 8 CODE PYTHON 1


File Edit View Navigate Code Refactor Run Tools VCS Window Help
pythonProject - voiceassistant.py
pythonProject voiceassistant.py
drs.py
voiceassistant.py
takeCommand() except Exception as e
TODO
6: Problems
Terminal
Python Console
ms-resource://microsoft.windowscommunicationsapps/hxoutlookintl/AppManifest_MailDesktop_DisplayName
Activate Windows
Go to Settings to activate Windows.
Event Log
53:22 CRLF UTF-8 4 spaces Python 3.8
Desktop
Pathak
ENG
13:13
01-11-2020

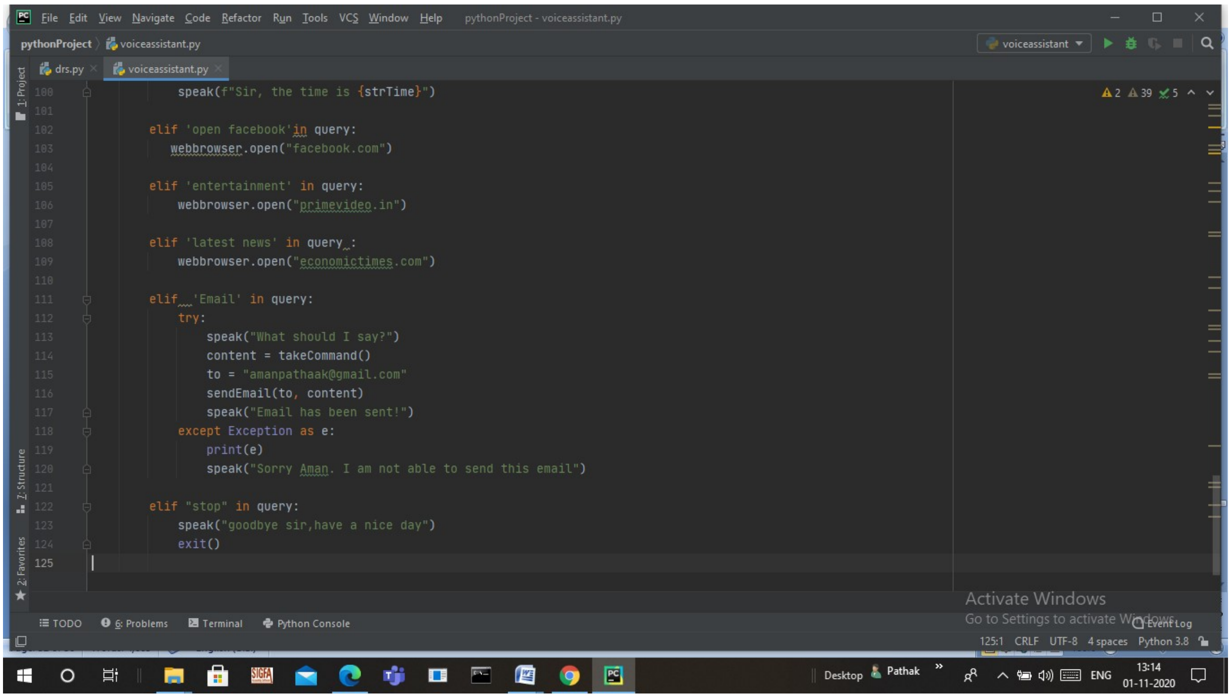
```python
        speak("Good Afternoon!")

    else:
        speak("Good Evening!")

    speak("I am your voice assistant champ.please tell me how may i help you")

def takeCommand():
    #It takes microphone input from the user and returns string output

    r = sr.Recognizer()
    with sr.Microphone() as source:
        print("Listening...")
        r.pause_threshold = 1
        r.energy_threshold = 10000
        audio = r.listen(source)

    try:
        print("Recognizing...")
        query = r.recognize_google(audio, language='en-in')
        print(f"User said: {query}\n")

    except Exception as e:
        # print(e)
        print("Say that again please...")
        return "None"
    return query
```

FIGURE 3.9CODE PYTHON 2



```python
def sendEmail(to, content):
    server = smtplib.SMTP('smtp.gmail.com', 587)
    server.ehlo()
    server.starttls()
    server.login('amanpathaak@gmail.com','virat96542')
    server.sendmail('amanpathaak@gmail.com',to,content)
    server.close()

if __name__ == "__main__":
    wishMe()
    while True:
    # if 1:
        query = takeCommand().lower()

        # Logic for executing tasks based on query
        if 'wikipedia' in query:
            speak('Searching Wikipedia...')
            query = query.replace("search wikipedia for",'').strip()
            results = wikipedia.summary(str(query),sentences=2)
            speak("According to Wikipedia")
            print(results)
            speak(results)

        elif 'open youtube' in query:
            webbrowser.open("youtube.com")

        elif 'open google' in query:
```

FIGURE3. 10PYTHON CODE 3



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

        elif 'open facebook'in query:
           webbrowser.open("facebook.com")

        elif 'entertainment' in query:
            webbrowser.open("primevideo.in")

        elif 'latest news' in query :
            webbrowser.open("economictimes.com")

        elif  'Email' in query:
            try:
                speak("What should I say?")
                content = takeCommand()
                to = "amanpathaak@gmail.com"
                sendEmail(to, content)
                speak("Email has been sent!")
            except Exception as e:
                print(e)
                speak("Sorry Aman. I am not able to send this email")

        elif "stop" in query:
            speak("goodbye sir,have a nice day")
            exit()
```

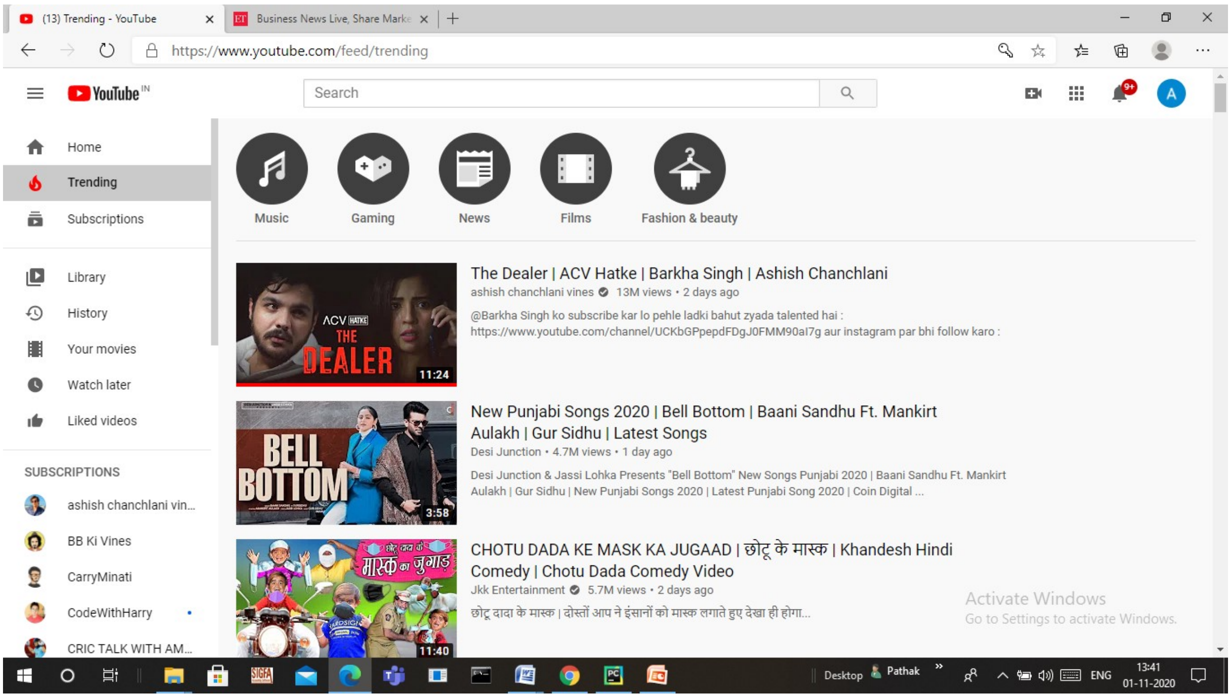
FIGURE 3.11CODE PYTHON 4

CHAPTER 4: RESULT ANALYSIS

After the writing of code our ultimate goal is to get the output from the above written code and the output obtained from the code is as follow.

NOTE:All the results are obtained by giving audio input to the system which it recognizes and then show us the desired output.

1.YOUTUBE: elif "youtube" in query:



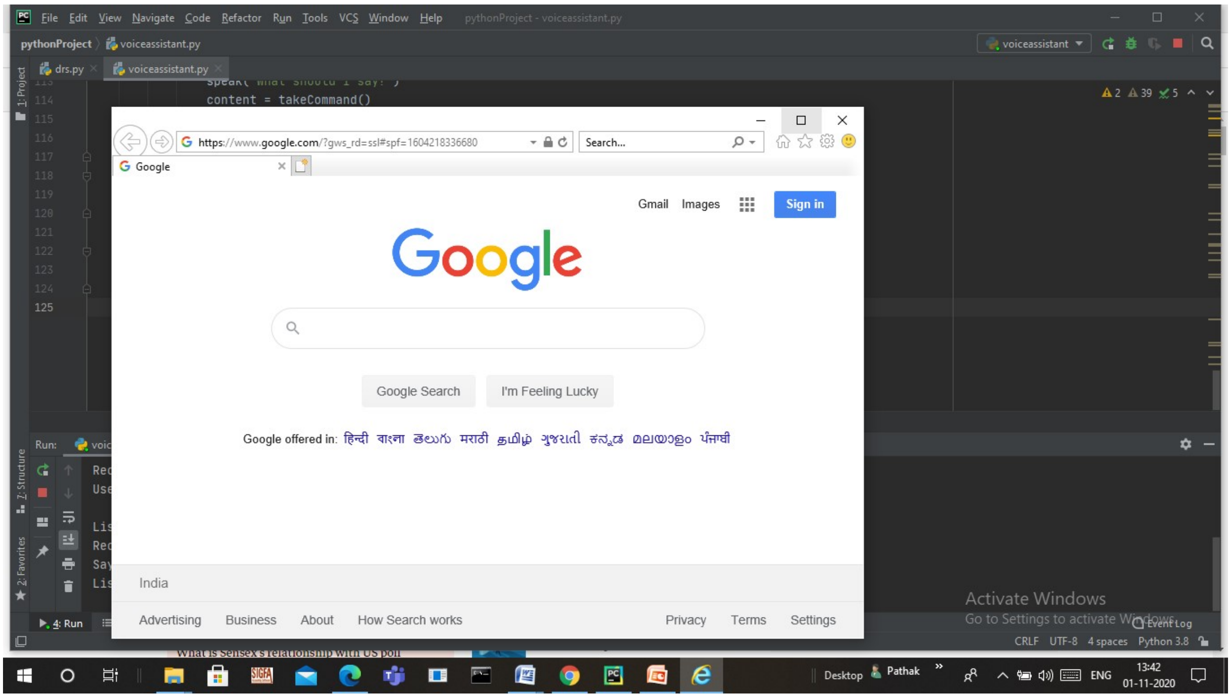
The Dealer | ACV Hatke | Barkha Singh | Ashish Chanchlani
ashish chanchlani vines · 13M views · 2 days ago
@Barkha Singh ko subscribe kar lo pehle ladki bahut zyada talented hai : https://www.youtube.com/channel/UCKbGPpepdFDgJ0FMM90aI7g aur instagram par bhi follow karo :

New Punjabi Songs 2020 | Bell Bottom | Baani Sandhu Ft. Mankirt Aulakh | Gur Sidhu | Latest Songs
Desi Junction · 4.7M views · 1 day ago
Desi Junction & Jassi Lohka Presents "Bell Bottom" New Songs Punjabi 2020 | Baani Sandhu Ft. Mankirt Aulakh | Gur Sidhu | New Punjabi Songs 2020 | Latest Punjabi Song 2020 | Coin Digital ...

CHOTU DADA KE MASK KA JUGAAD | छोटू के मास्क | Khandesh Hindi Comedy | Chotu Dada Comedy Video
Jkk Entertainment · 5.7M views · 2 days ago
छोटू दादा के मास्क | दोस्तों आप ने इंसानों को मास्क लगाते हुए देखा ही होगा...

Figure 4.1output fig 1

2.GOOGLE- elif “google” in query:



```
speak("What should I say?")
content = takeCommand()
```

```
Rec
Use

Lis
Rec
Say
Lis
```

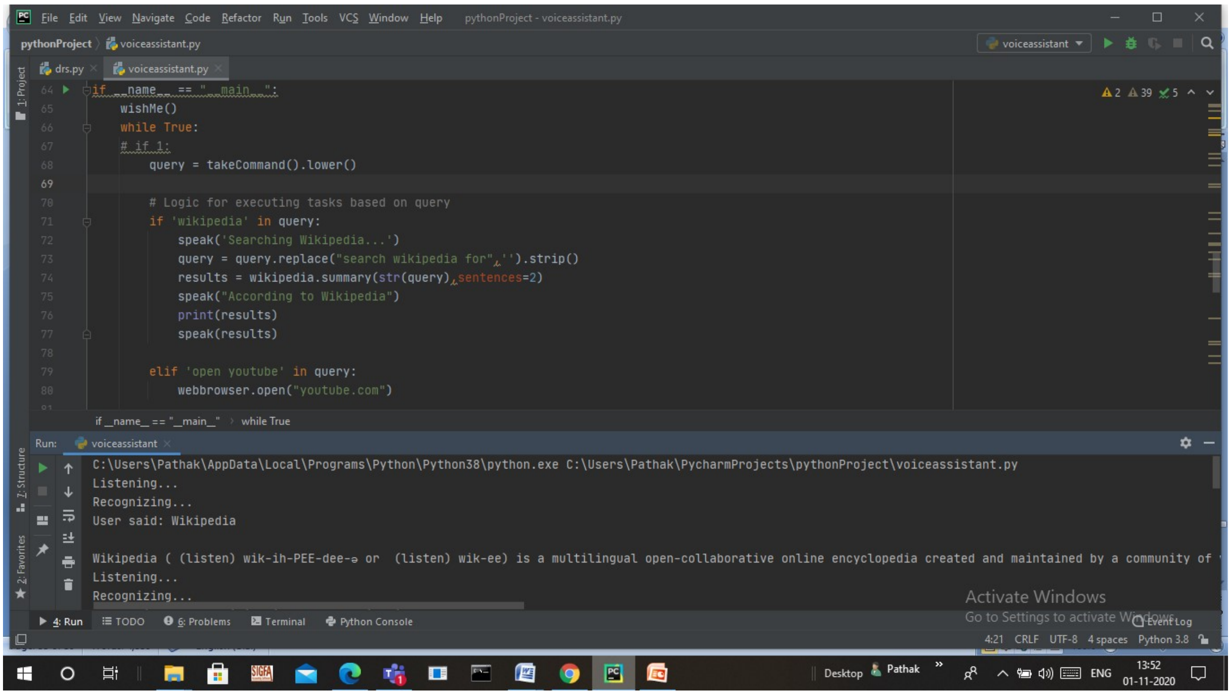
https://www.google.com/?gws_rd=ssl#spf=1604218336680

Google offered in: हिन्दी বাংলা తెలుగు मराठी தமிழ் ગુજરાતી ಕನ್ನಡ മലയാളം ਪੰਜਾਬੀ

India

FIGURE 12OUTPUT FIG 2

3.WIKIPEDIA: for Wikipedia the program logic says such that when we give “Wikipedia” in query and then we search as “search for anything in Wikipedia” it returns the first two lines the first output on Wikipedia which can be seen in the run section of below
program.



File Edit View Navigate Code Refactor Run Tools VCS Window Help pythonProject - voiceassistant.py
pythonProject voiceassistant.py
voiceassistant
drs.py
voiceassistant.py
1: Project
7: Structure
2: Favorites
Run:
voiceassistant
4: Run
TODO
6: Problems
Terminal
Python Console
Event Log
Activate Windows
Go to Settings to activate Windows.
4:21 CRLF UTF-8 4 spaces Python 3.8
Desktop
Pathak
ENG
13:52
01-11-2020

```python
if __name__ == "__main__":
    wishMe()
    while True:
    # if 1:
        query = takeCommand().lower()

        # Logic for executing tasks based on query
        if 'wikipedia' in query:
            speak('Searching Wikipedia...')
            query = query.replace("search wikipedia for",'').strip()
            results = wikipedia.summary(str(query),sentences=2)
            speak("According to Wikipedia")
            print(results)
            speak(results)

        elif 'open youtube' in query:
            webbrowser.open("youtube.com")
```

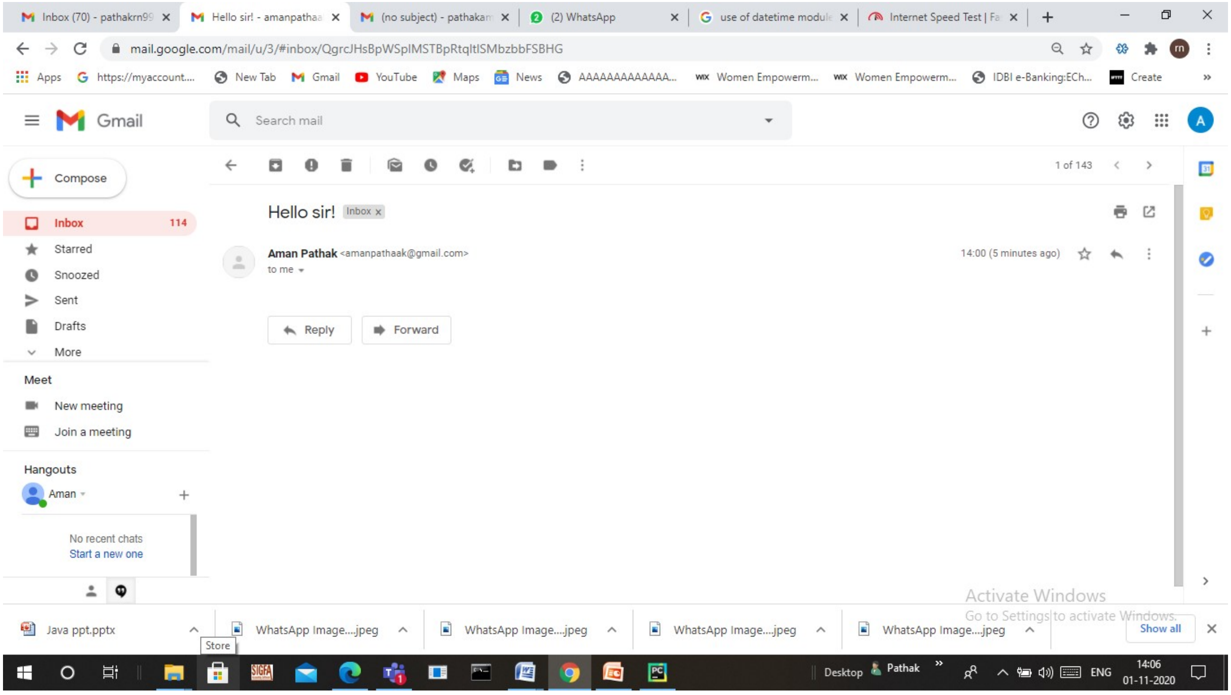
if _name_ == "_main_" › while True

```
C:\Users\Pathak\AppData\Local\Programs\Python\Python38\python.exe C:\Users\Pathak\PycharmProjects\pythonProject\voiceassistant.py
Listening...
Recognizing...
User said: Wikipedia

Wikipedia ( (listen) wik-ih-PEE-dee-ə or  (listen) wik-ee) is a multilingual open-collaborative online encyclopedia created and maintained by a community of
Listening...
Recognizing...
```

FIGURE 13OUTPUT FIG 3

4.SENDING EMAIL : elif “email” in query the assistant asks for the message to be send and when we provide that message it sends the email to the located email address.

Here,a email of hii sir! Is sent to my email address.



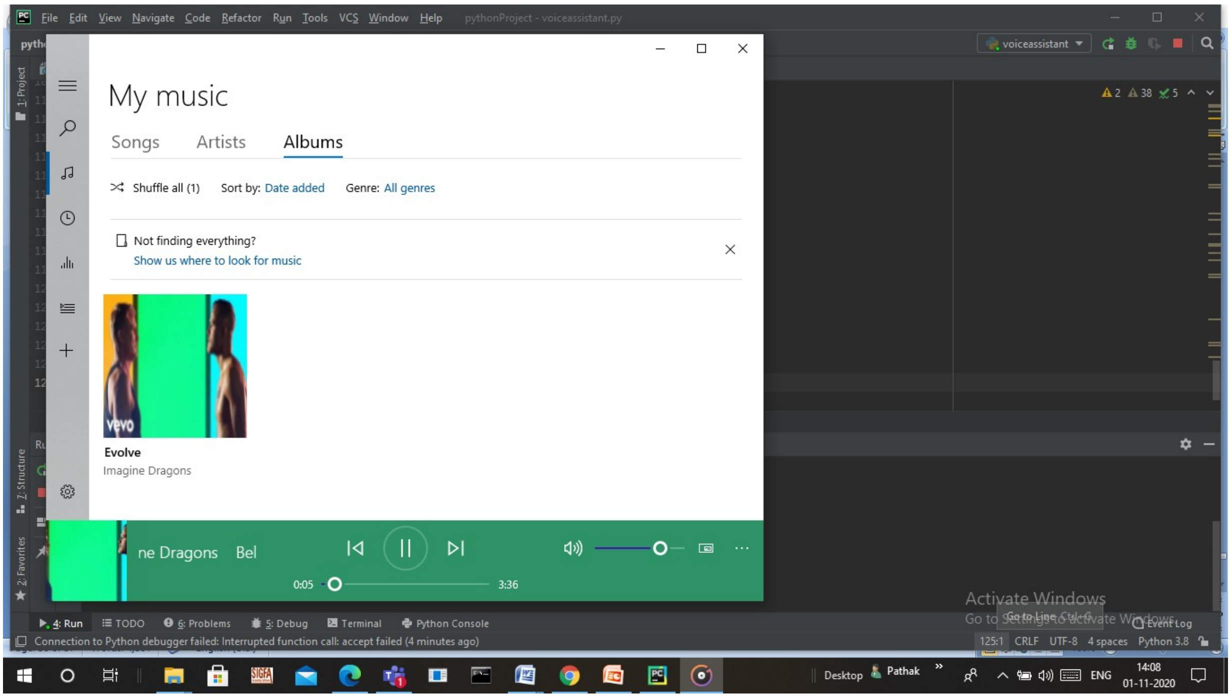
Hello sir!

Aman Pathak <amanpathaak@gmail.com>
to me

14:00 (5 minutes ago)

FIGURE 14OUTPUT FIG 4

5.PLAY MUSIC: elif “play music” in query the assistant plays the song specified in the program and the song starts.



Connection to Python debugger failed: Interrupted function call: accept failed (4 minutes ago)

FIGURE 15OUTPUT FIG 5

These are outputs of top 5 queries searched however queries such as time,latest news,entertainment is also available in the assistant.

CHAPTER 5: CONCLUSIONS

5.1 Problem and Issues in Current Application

The application is working well with speech recognition but a little problem can be caused due to the external environment noises and for that the disturbance can be removed by increasing the recognition voice frequency.A little speed of network can cause problem if the net is slow.Apart from this assistant is working properly and is ready to go.

5.2 Suggestion for New Application

Some exciting features can be added such as adding whatsapp texting and other interactive features to encourage the use of voice assistant for visually impaired people.

5.3 REFERENCES

Python introduction- https://www.geeksforgeeks.org/how-to-run-a-python-script

pyttsx3 - https://pypi.org/project/pyttsx3/

time - https://www.programiz.com/python-programming/time

smtplib - https://docs.python.org/3/library/smtplib.html

images – https://images.google.com

course- https://udemy.com