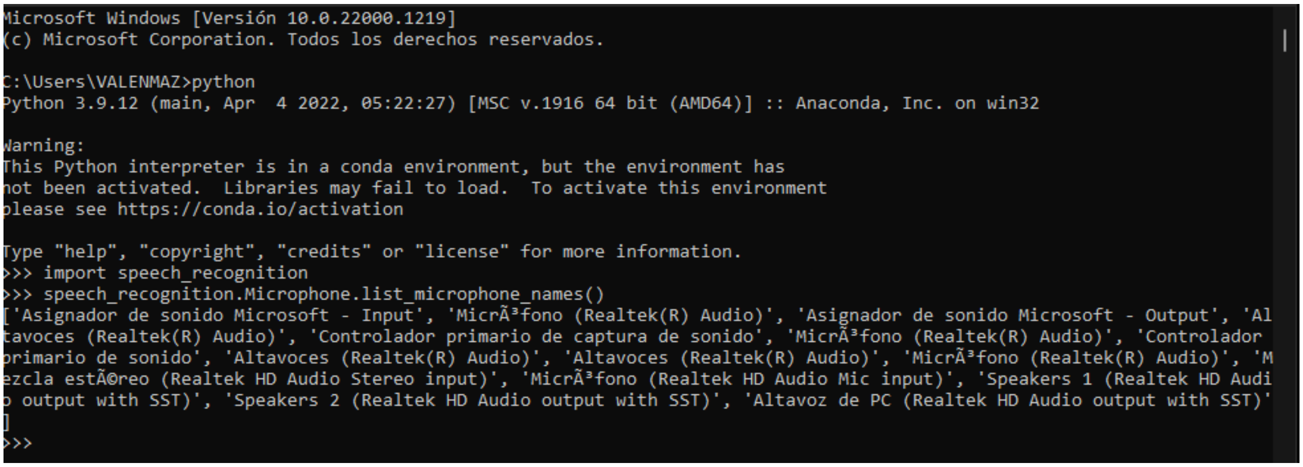
Search Assistant with Speech Recognition

Description: the search assistant recognizes speech and then queries the recognized speech and automatically opens web browser to display the results.

Dependencies:

- SpeechRecognition
- pyaudio

Default Browser: Brave/Google Chrome



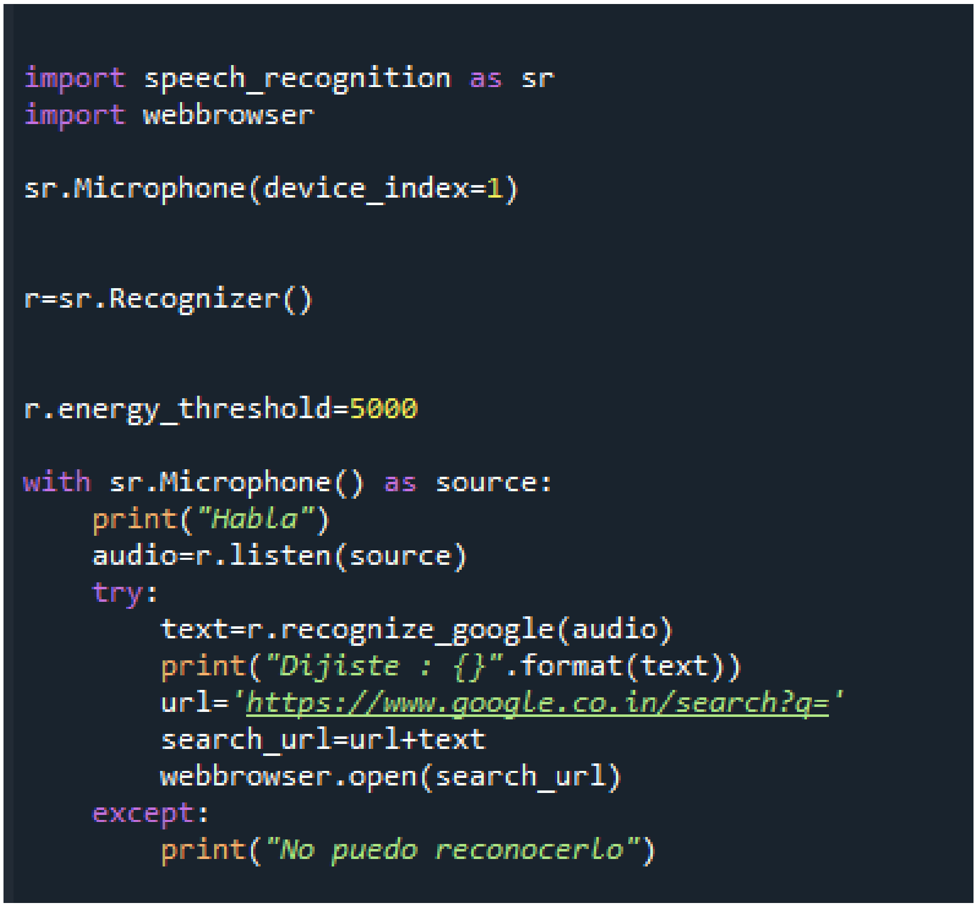
```
Microsoft Windows [Versión 10.0.22000.1219]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\VALENMAZ>python
Python 3.9.12 (main, Apr  4 2022, 05:22:27) [MSC v.1916 64 bit (AMD64)] :: Anaconda, Inc. on win32

Warning:
This Python interpreter is in a conda environment, but the environment has
not been activated.  Libraries may fail to load.  To activate this environment
please see https://conda.io/activation

Type "help", "copyright", "credits" or "license" for more information.
>>> import speech_recognition
>>> speech_recognition.Microphone.list_microphone_names()
['Asignador de sonido Microsoft - Input', 'MicrÃ³fono (Realtek(R) Audio)', 'Asignador de sonido Microsoft - Output', 'Al
tavoces (Realtek(R) Audio)', 'Controlador primario de captura de sonido', 'MicrÃ³fono (Realtek(R) Audio)', 'Controlador
primario de sonido', 'Altavoces (Realtek(R) Audio)', 'Altavoces (Realtek(R) Audio)', 'MicrÃ³fono (Realtek(R) Audio)', 'M
ezcla estÃ©reo (Realtek HD Audio Stereo input)', 'MicrÃ³fono (Realtek HD Audio Mic input)', 'Speakers 1 (Realtek HD Audi
o output with SST)', 'Speakers 2 (Realtek HD Audio output with SST)', 'Altavoz de PC (Realtek HD Audio output with SST)'
]
>>>
```

The first microphone that is listed is the Microsoft’s microphone. We used that one for our project.



```
import speech_recognition as sr
import webbrowser

sr.Microphone(device_index=1)


r=sr.Recognizer()


r.energy_threshold=5000

with sr.Microphone() as source:
    print("Habla")
    audio=r.listen(source)
    try:
        text=r.recognize_google(audio)
        print("Dijiste : {}".format(text))
        url='https://www.google.co.in/search?q='
        search_url=url+text
        webbrowser.open(search_url)
    except:
        print("No puedo reconocerlo")
```

Microphone(device_index = 1)

This is available if PyAudio is available, and is undefined otherwise.

We created a new Microphone instance, which represents a physical microphone on the computer.

If device_index is unspecified or None, the default microphone is used as the audio source. Otherwise, device_index should be the index of the device to use for audio input. In our case, we used the first microphone that was listed.

r.energy_threshold=5000

This is basically how sensitive the recognizer is to when recognition should start. Higher values mean that it will be less sensitive.

This value depends entirely on your microphone or audio data. Good values typically range from 50 to 4000.

with sr.Microphone() as source:

It opens the microphone and starts recording. The microphone will be our source to capture the command given by the user.

audio=r.listen(source)

This function listens to the phrase and extracts it into audio data.

text=r.recognize_google(audio)

we pass the audio object to the recognize_google() method of the Recognizer() class object and the audio file will be converted to text.

webbrowser.open(search_url)

It opens the url in a new window of the default browser, if possible, otherwise, open url in the only browser window.