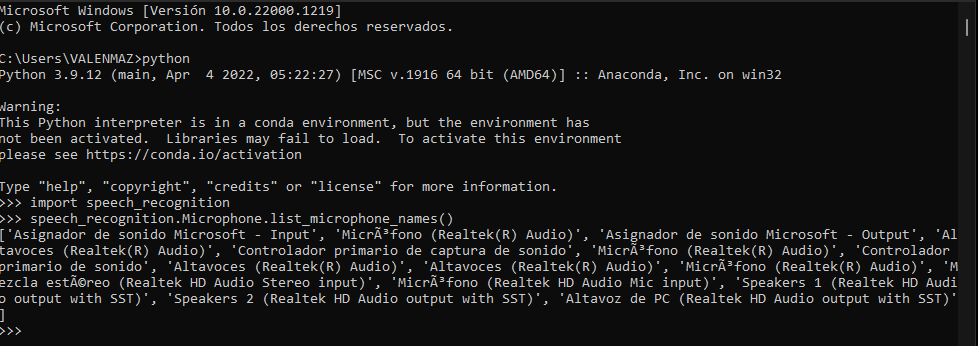
**Search Assistant with Speech Recognition**

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

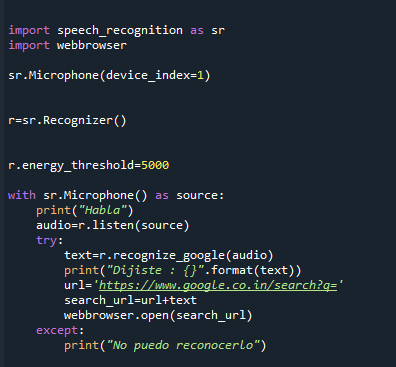
**Dependencies:**

* SpeechRecognition
* pyaudio

**Default Browser:** Brave/Google Chrome



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



**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.