

# Procedura

- Apri un terminale e inserisci il seguente comando:

**git clone <https://github.com/ArcoBit-PC/Carla.git>** (Script per la versione italiana "Carla.py")

Inserisci il seguente comando per aprire il file bashrc:

**sudo nano ~/.bashrc**

Scorri fino alla fine del file utilizzando la tastiera e aggiungi le seguenti righe alla fine [assicurati di includere #(Descrizione)]:

```
# sets a location where the Raspberry Pi OS and Python can look for
# executable/configuration files
export PATH="$HOME/.local/bin:$PATH"
```

Premi contemporaneamente i tasti CTRL e X sulla tastiera, quindi premi Y e poi premere Invio per salvare le modifiche del file.  
Dopo inserisci il seguente comando:

**sudo reboot**

Una volta completato il riavvio, apri un terminale e inserisci i seguenti comandi nel seguente ordine:

```
sudo apt update
sudo apt full-upgrade
pip3 install --upgrade pip
sudo apt-get install portaudio19-dev (Quando ti viene chiesto se vuoi continuare, premi Si e poi invio)
pip3 install pyaudio
pip3 install pvrecorder
pip3 install pvporcupine

pip3 install pvcobra
pip3 install pvleopard
pip3 install --upgrade openai
pip3 install boto3
pip3 install awscli
```

**sudo reboot** (Riavvio Raspberry pi)

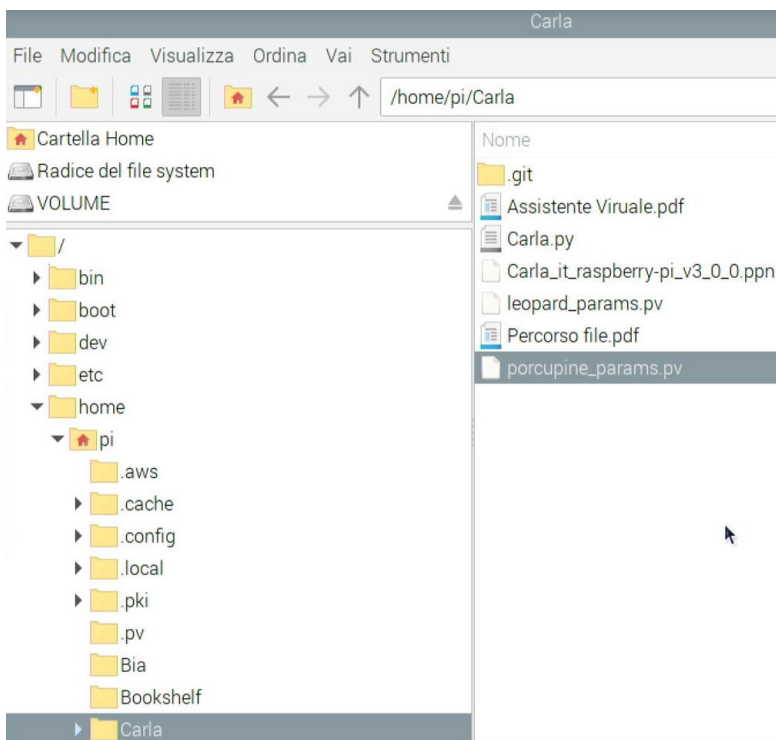
- Ora configura AWS con il seguente comando:

**aws configure**

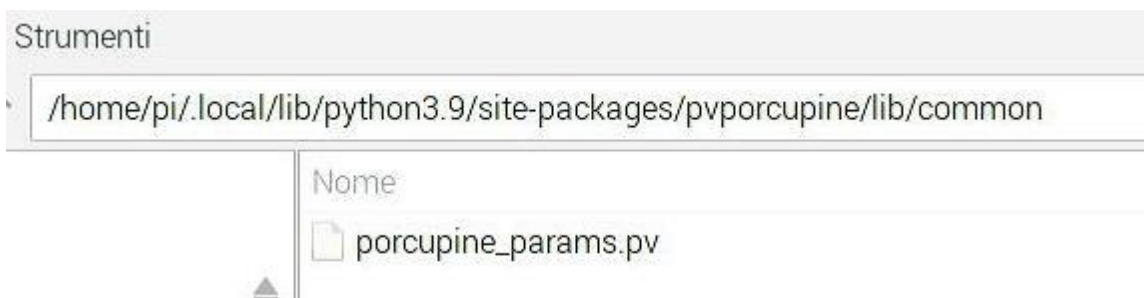
**AWS Access Key ID [None]:** [digita l'ID della tua chiave di accesso e premi invio]  
**AWS Secret Access Key [None]:** [digita la tua chiave di accesso segreta e premi invio]  
**Default region name [None]:** (cerca qui il nome della regione più vicina a te):  
[https://docs.aws.amazon.com/general/latest/gr/rande.html#pol\\_region](https://docs.aws.amazon.com/general/latest/gr/rande.html#pol_region)

**Default output format [None]:** [lascia vuoto questo campo e premi invio]

- **Copia il file porcupine\_params.pv che si trova nella cartella Carla**



e incollalo sulla cartella: `/home/pi/.local/lib/python3.9/site-packages/pvporcupine/lib/common` ([sovrascrivi il file](#))



- **Copia il file leopard\_params.pv che si trova nella cartella Carla**

**e incollalo sulla cartella:**

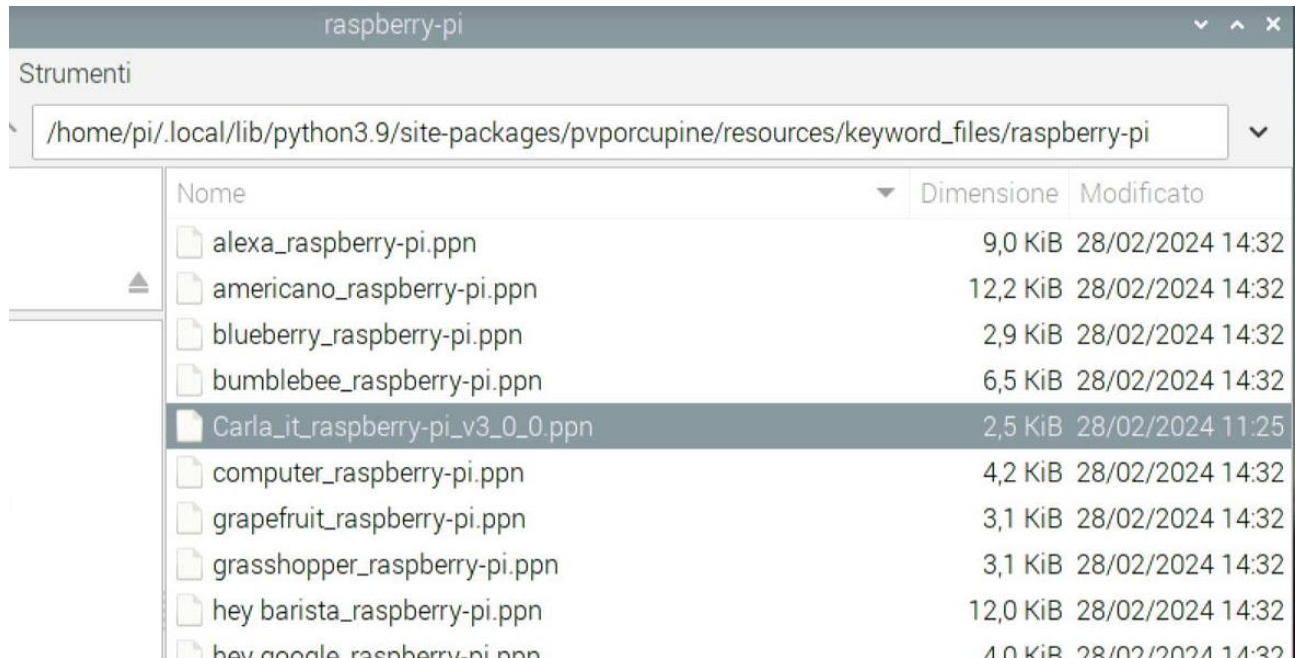
`/home/pi/.local/lib/python3.9/site-packages/pvleopard/lib/common` ([sovrascrivi il file](#))



- **Copia il file `Carla_it_raspberry-pi_v3_0_0.ppn` che si trova nella cartella **Carla** e incollalo sulla cartella:**

`/home/pi/.local/lib/python3.9/site-packages/pvporcupine/resources/keyword_files/raspberry-pi`

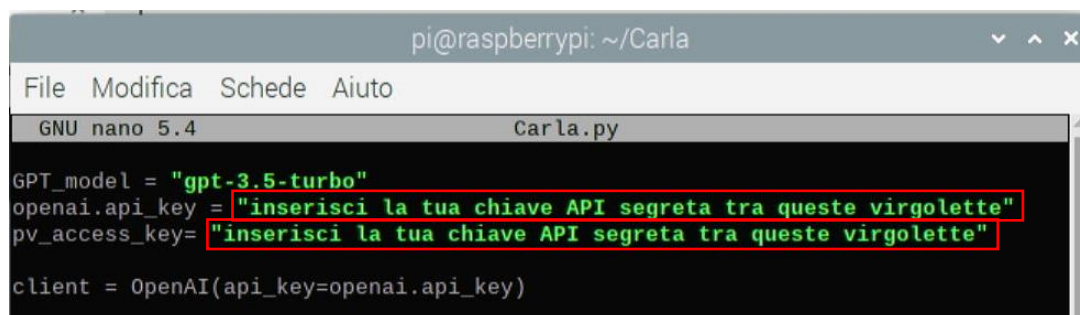
**N.B. Questo file attiva l'assistente vocale con la parola "Carla".**



- **Digita su linea di comando:**

`pi@raspberrypi:~ $cd Carla`

`pi@raspberrypi:~/Carla $sudo nano Carla.py`



**Inserire la chiave API di OpenAI e le API Picovoice (AccessKey) nello script python**

- Attivazione vocale <https://picovoice.ai> (modificare “Carla” con la parola che avete creato su picovoice).

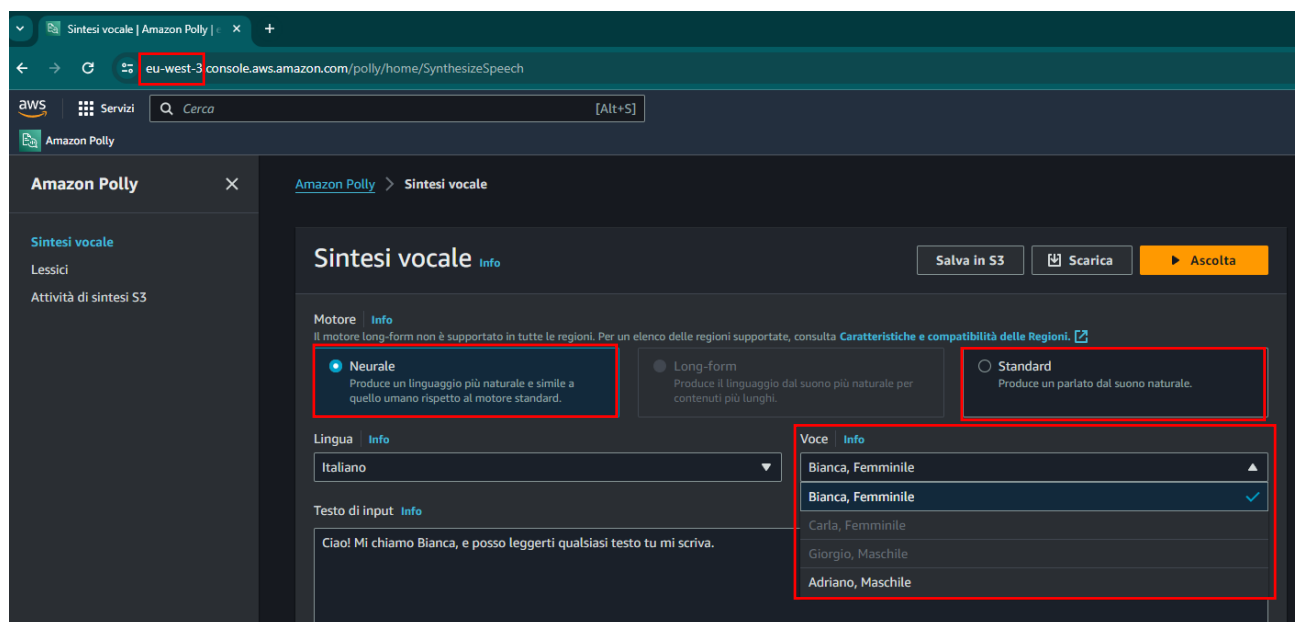
N.B. se hai generato un file con una parola di tua scelta (Esempio `pippo.ppn` ricordati di copiare il file sul percorso

```
/home/pi/.local/lib/python3.9/site-packages/pvporcupine/resources/keyword_files/raspberry-pi
```

Modifica anche sullo script python la parola “Carla” con quella che hai generato su picovoice

```
def wake_word():
    keywords = ["Carla"]
    porcupine = pvporcupine.create(keywords=keywords,
                                   access_key=pv_access_key,
                                   sensitivities=[0.1], #da 0 a 1.0
```

- Su sintesi vocale puoi scegliere tra voci Neurale e standard, nel menu a tendina visualizzi le voci disponibili. Nell'esempio Zona: **eu-west-3** voci disponibili Bianca e Adriano.



Modifica la voce di tua scelta sullo script Python

```
def voice(chat):
    voiceResponse = polly.synthesize_speech(Text=chat, OutputFormat="mp3",
                                             VoiceId="Bianca") #su AWS altre voci disponibili oltre
    if "AudioStream" in voiceResponse:
        with voiceResponse["AudioStream"] as stream:
            output_file = "speech.mp3"
```

- Esegui lo Script: `Carla.py`

`pi@raspberrypi:~ $cd Carla`

`pi@raspberrypi:~/Carla $python3 Carla.py`

