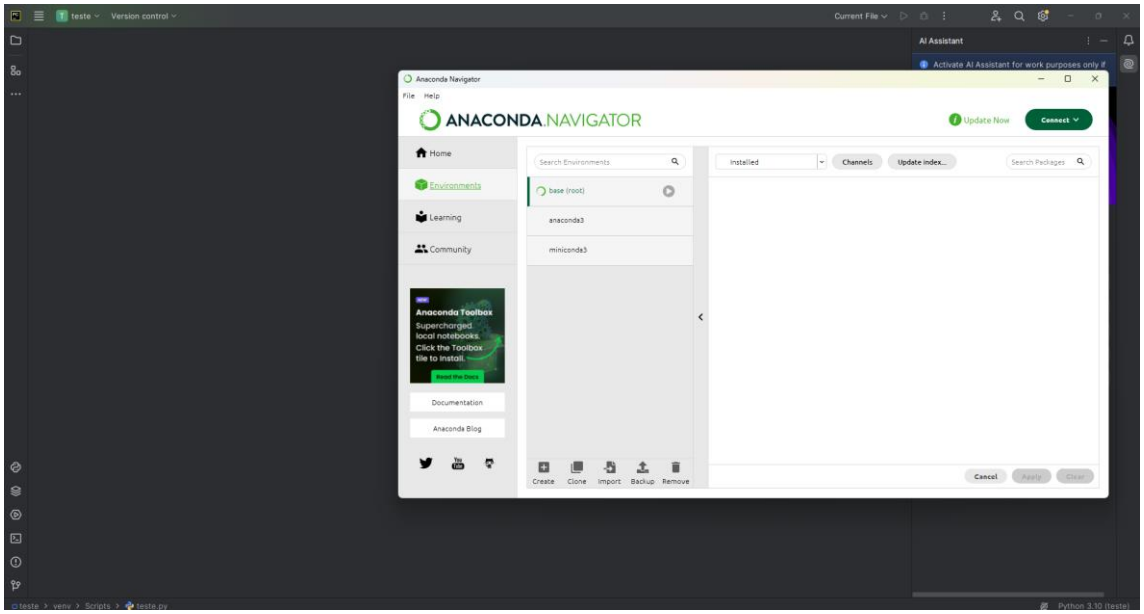
Linguagem de Processamento Natural

Introdução Assistente Virtual

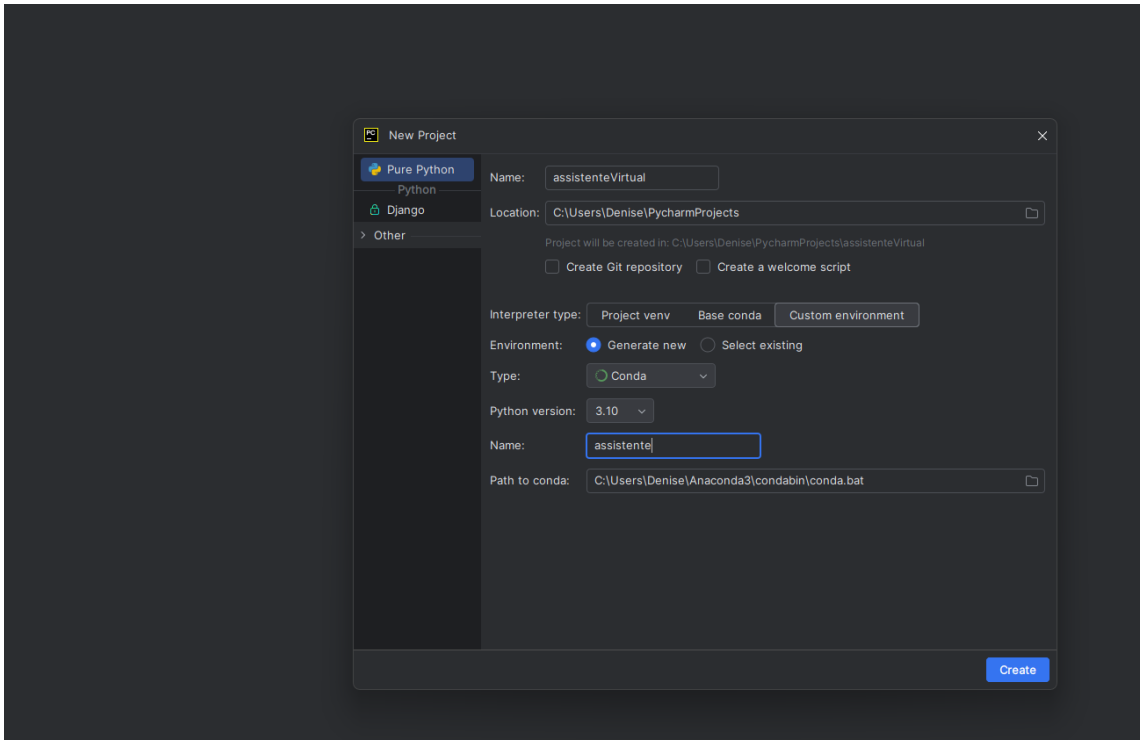
Ferramentas :

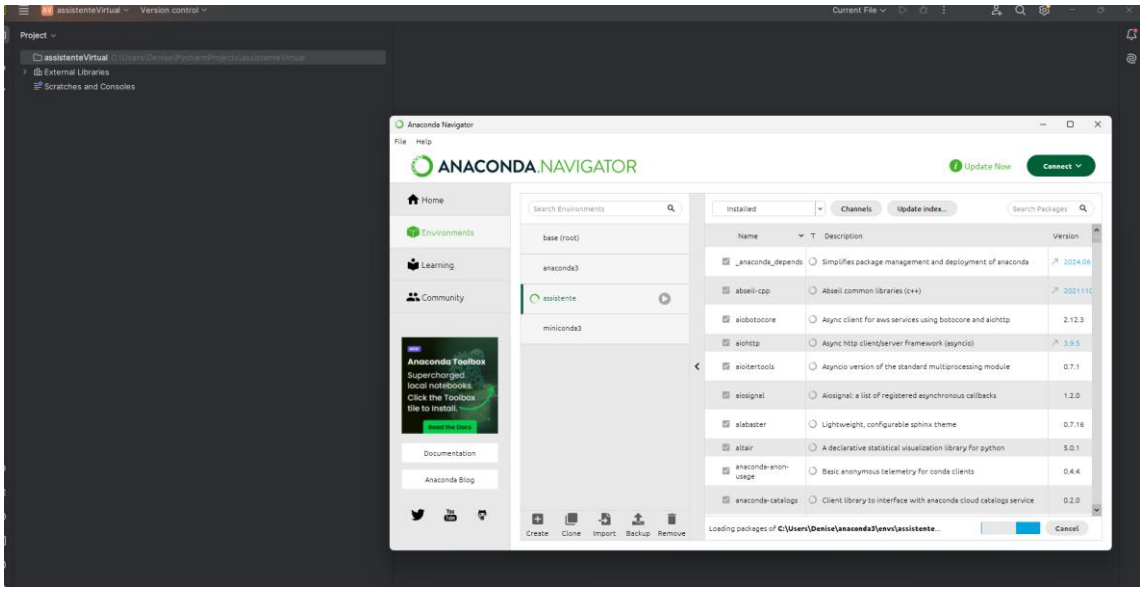
Anaconda (pacotes adicionais já instalados), pode criar ambientes diferentes.

PyCharm : IDE para trabalharmos com o Python – versão Comunity



Gerar um novo projeto no ambiente novo do Anaconda





Instalando bibliotecas

1- Playsound

Para tocar som no ambiente~, usaremos a biblioteca para que o assistente execute os arquivos de áudio.

[playsound 1.3.0 on PyPI - Libraries.io - security & maintenance data for open source software]

[gTTS — gTTS documentation (Para sintetizar a fala)]

[pip install gTTS]

pip install playsound == 1.2.2

```
(assistente) PS C:\Users\Denise\PycharmProjects\assistenteVirtual> pip install playsound==1.3.0
Collecting playsound==1.3.0
  Downloading playsound-1.3.0.tar.gz (7.7 kB)
  Preparing metadata (setup.py) ... done
Building wheels for collected packages: playsound
  Building wheel for playsound (setup.py) ... done
  Created wheel for playsound: filename=playsound-1.3.0-py3-none-any.whl size=7044 sha256=a0cdfeae76d713e33064fe048ac613bfac13bc53968d5c07dd507c3ec6a549cb
  Stored in directory: c:\users\denise\appdata\local\pip\cache\wheels\90\89\ed\2d643f4226fc8c7c9156fc28abd8051e2d2c0de37ae51ac45c
Successfully built playsound
Installing collected packages: playsound
Successfully installed playsound-1.3.0
(assistente) PS C:\Users\Denise\PycharmProjects\assistenteVirtual>
```

2- Biblioteca SpeechRecognition

Biblioteca para reconhecimento de fala , gerando um texto de acordo com o que vc falou.

Linguagem de Processamento Natural

pip install SpeechRecognition==3.8.1

```
(assistente) PS C:\Users\Denise\PycharmProjects\assistenteVirtual> pip install SpeechRecognition==3.8.1
Collecting SpeechRecognition==3.8.1
  Downloading SpeechRecognition-3.8.1-py2.py3-none-any.whl.metadata (28 kB)
Downloading SpeechRecognition-3.8.1-py2.py3-none-any.whl (32.8 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 32.8/32.8 MB 18.1 MB/s eta 0:00:00
Installing collected packages: SpeechRecognition
Successfully installed SpeechRecognition-3.8.1
(assistente) PS C:\Users\Denise\PycharmProjects\assistenteVirtual>
```

3- Biblioteca Pyttsx3

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

Bilbioteca de TEXTO para gerar a fala. O assistente irá conversar

pip install pyttsx3==2.90

```
  Downloading comtypes-1.4.8-py3-none-any.whl.metadata (7.1 kB)
Collecting pypiwin32 (from pyttsx3==2.90)
  Downloading pypiwin32-223-py3-none-any.whl.metadata (236 bytes)
Collecting pywin32 (from pyttsx3==2.90)
  Downloading pywin32-308-cp310-cp310-win_amd64.whl.metadata (8.3 kB)
Downloading pyttsx3-2.90-py3-none-any.whl (39 kB)
Downloading comtypes-1.4.8-py3-none-any.whl (229 kB)
Downloading pypiwin32-223-py3-none-any.whl (1.7 kB)
Downloading pywin32-308-cp310-cp310-win_amd64.whl (6.6 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 6.6/6.6 MB 9.0 MB/s eta 0:00:00
Installing collected packages: pywin32, pypiwin32, comtypes, pyttsx3
Successfully installed comtypes-1.4.8 pypiwin32-223 pyttsx3-2.90 pywin32-308
(assistente) PS C:\Users\Denise\PycharmProjects\assistenteVirtual>
```

4-Biblioteca Tensorflow

Cenários de Deep Learning e Redes Neurais , para usar o classificador de emoções.

Pip install tensorflow

```
Using cached werkzeug-3.0.4-py3-none-any.whl (227 kB)
Using cached namex-0.0.8-py3-none-any.whl (5.8 kB)
Downloading optree-0.13.0-cp39-cp39-win_amd64.whl (270 kB)
Using cached rich-13.9.3-py3-none-any.whl (242 kB)
Downloading importlib_metadata-8.5.0-py3-none-any.whl (26 kB)
Using cached markdown_it_py-3.0.0-py3-none-any.whl (87 kB)
Downloading MarkupSafe-3.0.2-cp39-cp39-win_amd64.whl (15 kB)
Using cached pygments-2.18.0-py3-none-any.whl (1.2 MB)
Using cached mdurl-0.1.2-py3-none-any.whl (10.0 kB)
Downloading zipp-3.20.2-py3-none-any.whl (9.2 kB)
Installing collected packages: namex, libclang, flatbuffers, zipp, wrapt, typing-extensions, termcolor, tensorflow-io-gcs-filesystem, tensorboard-data-server, si
x, pygments, protobuf, opt-einsum, numpy, mdurl, MarkupSafe, grpcio, gast, absl-py, werkzeug, optree, ml-dtypes, markdown-it-py, importlib-metadata, h5py, google
-pasta, astunparse, rich, markdown, tensorboard, keras, tensorflow-intel, tensorflow
Successfully installed MarkupSafe-3.0.2 absl-py-2.1.0 astunparse-1.6.3 flatbuffers-24.3.25 gast-0.6.0 google-pasta-0.2.0 grpcio-1.67.0 h5py-3.12.1 importlib-meta
data-8.5.0 keras-3.6.0 libclang-18.1.1 markdown-3.7 markdown-it-py-3.0.0 mdurl-0.1.2 ml-dtypes-0.4.1 namex-0.0.8 numpy-1.26.4 opt-einsum-3.4.0 optree-0.13.0 prot
obuf-4.25.5 pygments-2.18.0 rich-13.9.3 six-1.16.0 tensorboard-2.17.1 tensorboard-data-server-0.7.2 tensorflow-2.17.0 tensorflow-intel-2.17.0 tensorflow-io-gcs-f
ilesystem-0.31.0 termcolor-2.5.0 typing-extensions-4.12.2 werkzeug-3.0.4 wrapt-1.16.0 zipp-3.20.2
(base) PS C:\Users\Denise\PycharmProjects\assistenteVirtual>
```

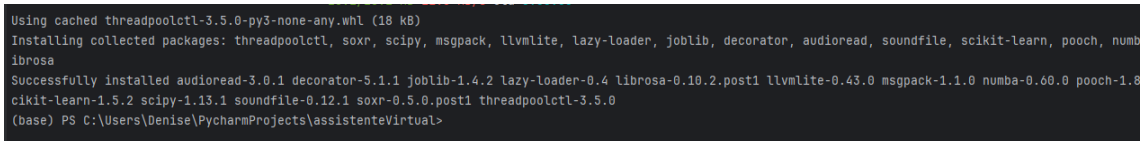
Linguagem de Processamento Natural

5-Librosa

Biblioteca para análise de música e áudio.

librosa — librosa 0.10.2 documentation

pip install librosa

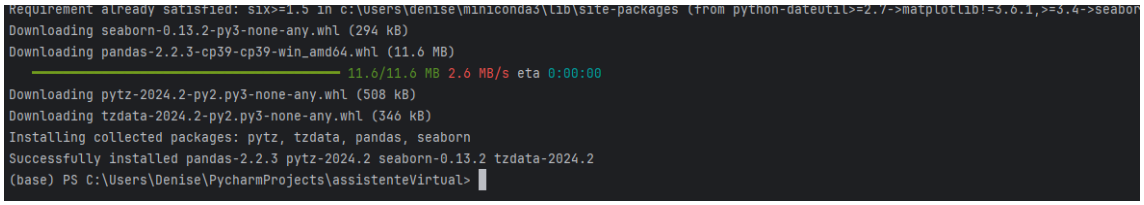


6-Biblioteca Matplotlib e Seaborn

Usarmos gráficos no Python para visualização de alguns resultados da aplicação

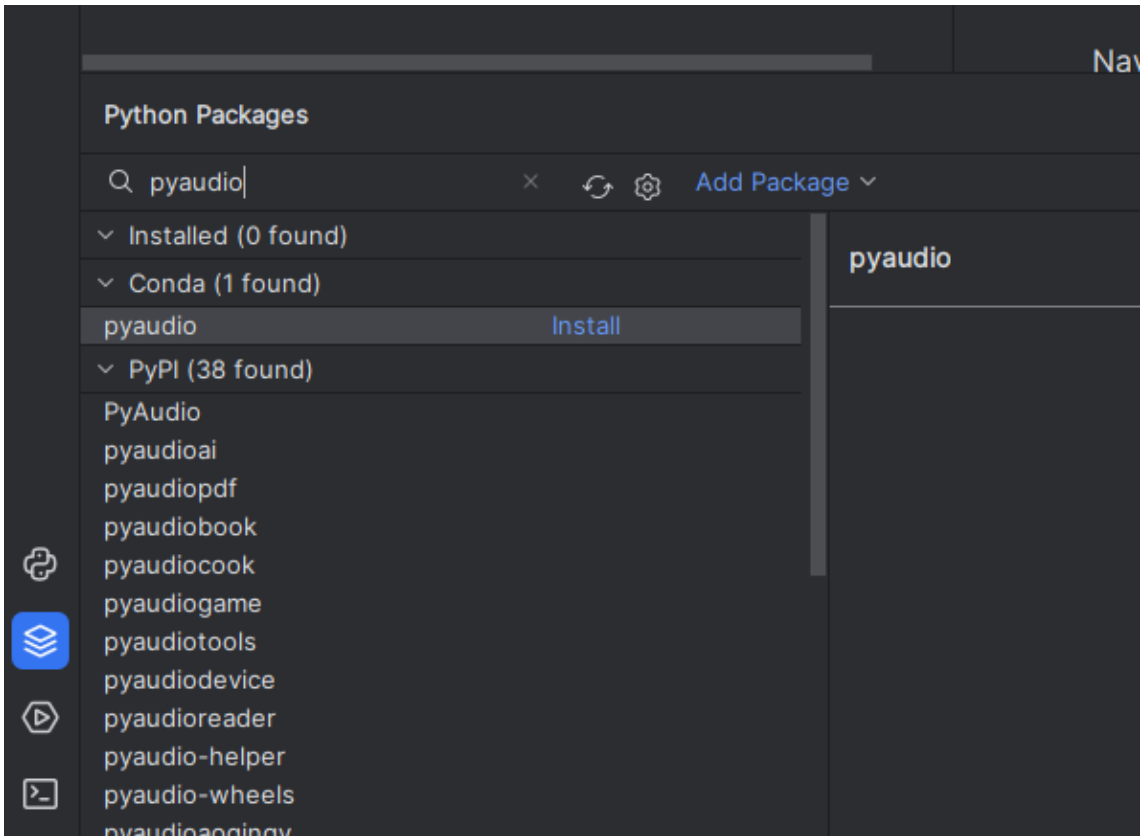
pip install matplotlib

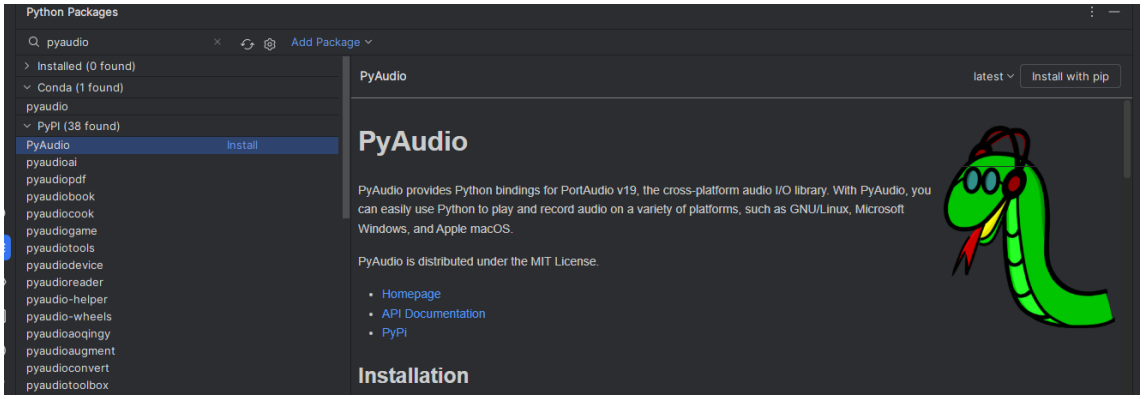
pip install seaborn



7-Bilbioteca Pyaudio

A biblioteca de E/S de áudio multiplataforma. Com o PyAudio, você pode usar facilmente o Python para reproduzir e gravar áudio em várias plataformas

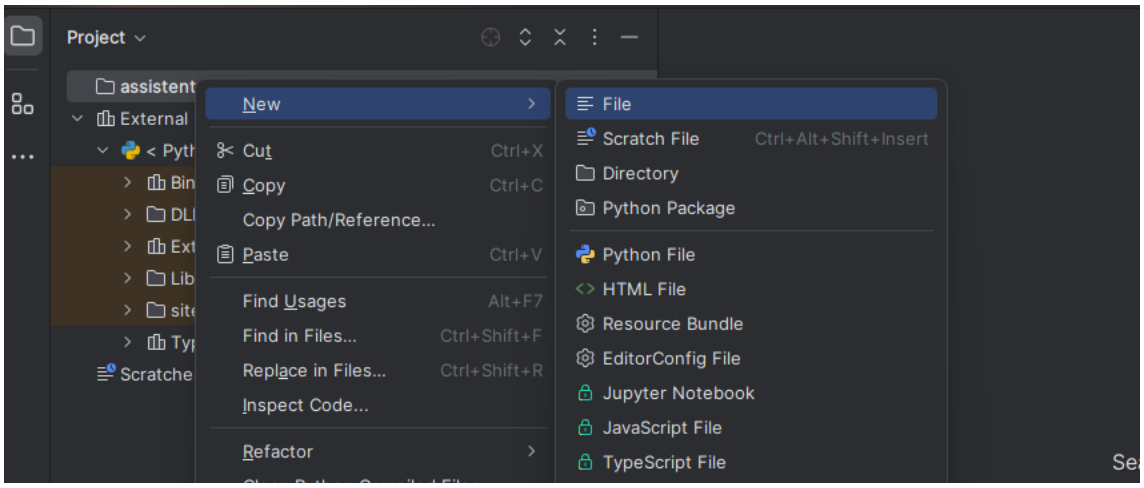




Testando a instalação

Para testar a instalação das bibliotecas , vamos gerar um arquivo em Python :

Linguagem de Processamento Natural




```
from gtts import gTTS
from playsound import playsound
playsound('n2.mp3')
def criar_audio(audio):
   tts = gTTS(audio, lang='pt-br')
   tts.save('bem_vindo.mp3')
   playsound('bem_vindo.mp3') # WINDONS

criar_audio('Oi, eu sou a Denise.')

import speech_recognition
print('Speech Recognition: ', speech_recognition.__version__)

import pyttsx3
pyttsx3.speak('Testando a biblioteca')

import tensorflow
print('TensorFlow')

import librosa
print('Librosa:', librosa.version)

import matplotlib
print('Matplotlib: ', matplotlib._get_version())
import seaborn
print('Seaborn: ')

import pyaudio
print('Pyaudio ok!')
```