1. Prendre le contrôle du microphone:

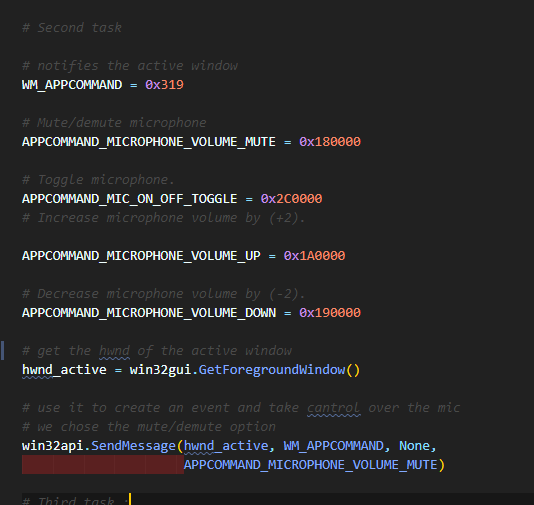
Pour cela, on aura besoin 2 bibliothèques python reliées a I’ API et la GUI Windows:

-win32api

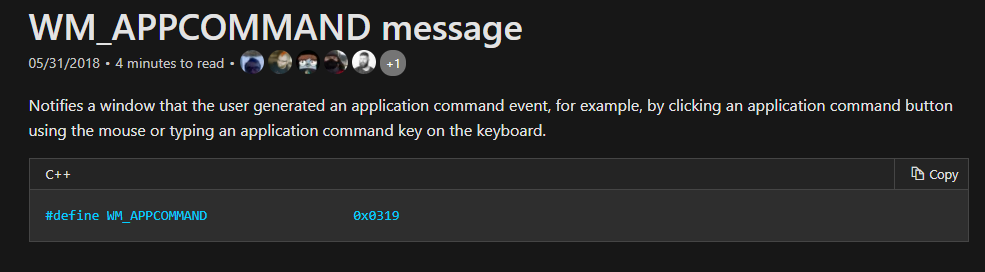
-win32gui

A l’aide de la bibliothèque win32.gui on peut obtenir le HWND( handle window) d’une fenêtre active qu’on utilisera pour faire appel à la fenêtre et créer un événement.

Lors de la création de l’événement, on aura accès au microphone via l’ API Windows ( win32.api), qu’on pourra par la suite couper son son, augmenter son volume ext ... .



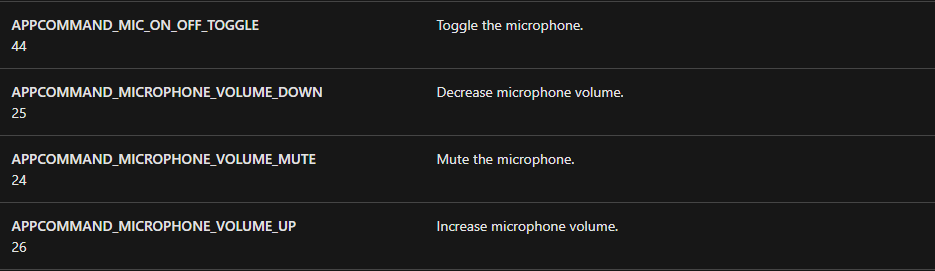
- le WM\_APPCOMMAND notifie la fenêtre Windows que l’user a créé un événement (par défaut il est initialisé a 0x319 en hexadécimal) .



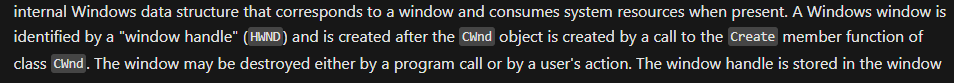
- l’ API supporte plusieurs fonctionnalités pour gérer le microphone installé sur la machine Windows. On en a choisi 4:

1. couper le son
2. Activer/désactiver le micro
3. Augmenter le volume du microphone
4. Diminuer le volume du microphone

Chaque fonctionnalité a son propre code qui sera ajouté en hexadécimal.

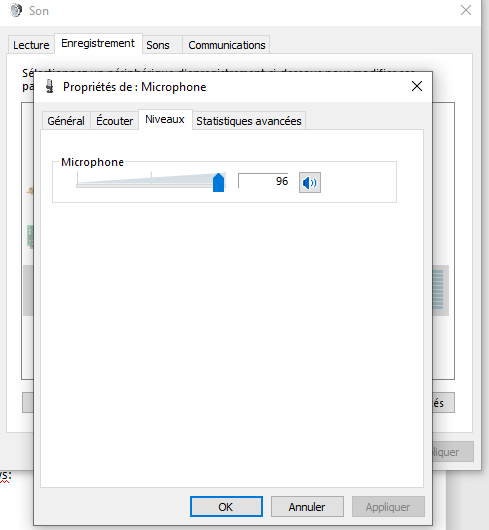


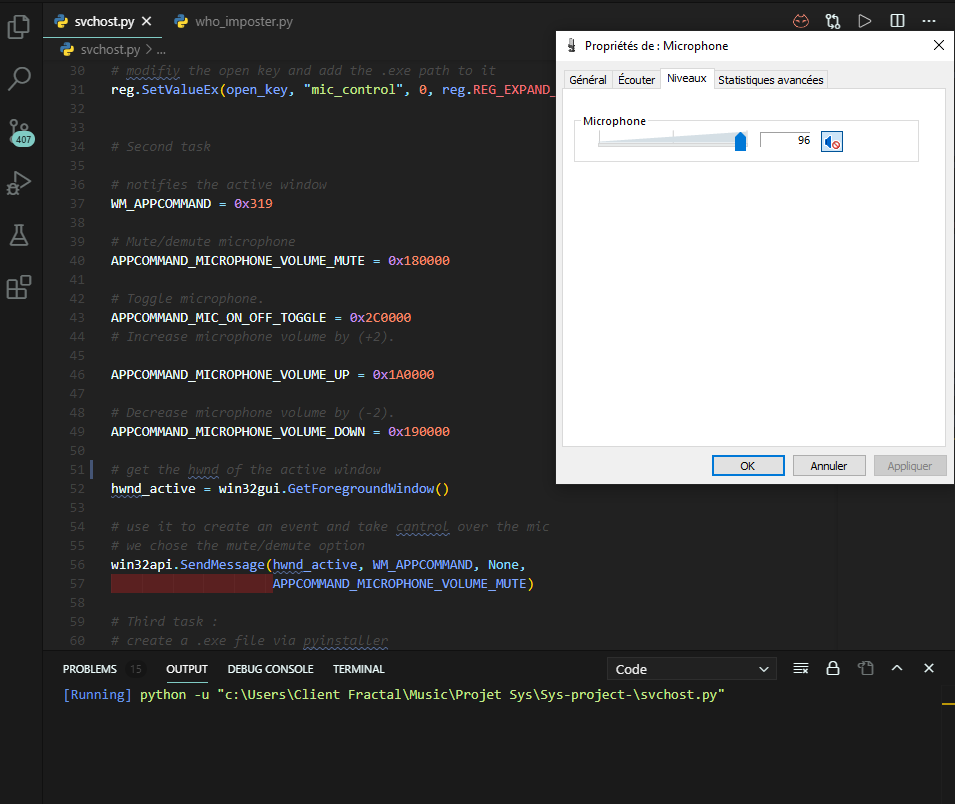
-Le nombre HWND ( window handle) est l’identifiant d’une fenêtre Windows.



-en utilisant le WM\_APPCOMMAND on peut maintenant notifier la fenêtre active grâce a son HWND et créer un événement qui exécutera une des fonctionnalités mentionnées ci dessus (dans ce cas: couper le son).

Démonstration sur une machine windows:

1. Vérifier l’état du microphone(Panneau de configuration/son/Enregistrement/microphone/niveau) 
2. Exécuter le bout de code précédent et revérifier l’état du microphone:



On peut remarquer que l’état du microphone a changé ( pas de son) donc on a pris le contrôle du microphone de ce pc.