

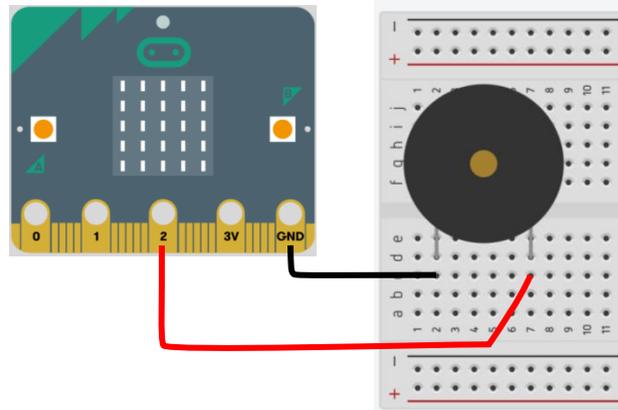
## Exemple 4 : Générer un son.

**Objectifs :** Emettre un son sans choisir de fréquence.

Répéter toutes les secondes l'émission d'un son de 440 Hz pendant d'une durée de 0,5 s.

**Document 1 :** Schéma du montage.

Utiliser les ports 2 et GND pour connecter le buzzer.



**Document 2 :** Fonctions utiles.

Attention à la ponctuation et à la casse.

Fonction	Rôle
<code>from microbit import *</code>	Permet d'utiliser les fonctions de base liées à la carte microbit
<code>import utime</code>	Permet d'utiliser les fonctions liées au temps en $\mu$ s
<code>import music</code>	Permet d'utiliser les fonctions pour faire des sons avec un haut-parleur
<code>while True:</code>	Fait une boucle infinie
<code>pin①.write_digital(②)</code>	Permet de fixer l'état du port de connexion. ① : Ecrire le numéro du port concerné ② : Ecrire 1 pour un état haut (1 logique soit 3,3V électrique) Ecrire 0 pour un état bas (0 logique ou 0V électrique) <code>pin2.write_digital(1)</code> → fixe le niveau haut sur la borne 2
<code>utime.sleep_us(①)</code>	Permet de programmer une temporisation en microsecondes ① : Ecrire la durée en microsecondes. <code>utime.sleep_us(500)</code> → attend pendant 500 $\mu$ s
<code>music.pitch(①, ②, pin③)</code>	Permet de jouer un son avec le haut-parleur ① : Ecrire la fréquence du son souhaité en Hz ② : Ecrire la durée du son en ms ③ : Ecrire le numéro du port concerné <code>music.pitch(440, 3000, pin1)</code> → joue un son à 440 Hz pendant 3 secondes avec le haut-parleur connecté à la borne 1

**Document 3 :** Structure du programme

La capture d'écran montre l'interface de programmation Mu 1.0.2. Une barre d'outils en haut propose des actions comme Mode, Nouveau, Charger, Enregistrer, Flasher, Fichiers, REPL, Graphique, Zoomer, Dé-zoomer, Thème, Vérifier, Aide et Quitter. Le code Python est écrit dans une fenêtre intitulée 'Atelier3a.py' :

```
1 from microbit import *
2 import utime
3 import music
4
5 while True:
6     # Ecrire ici le code qui se répète à l'infini
7
```