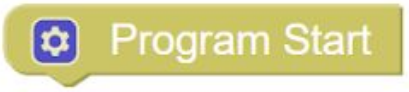
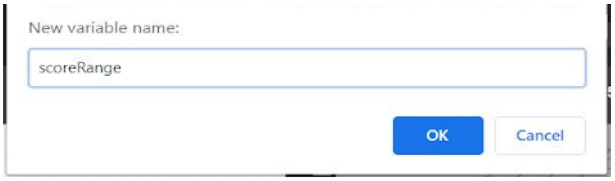
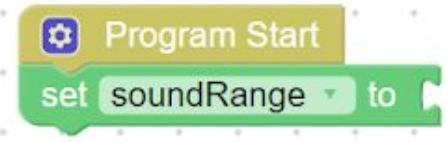
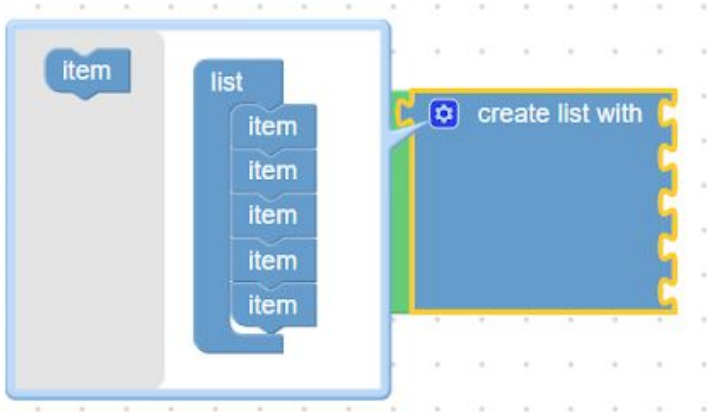
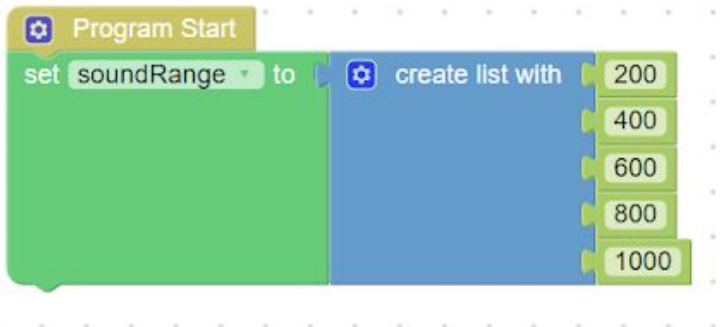
Step-by-Step

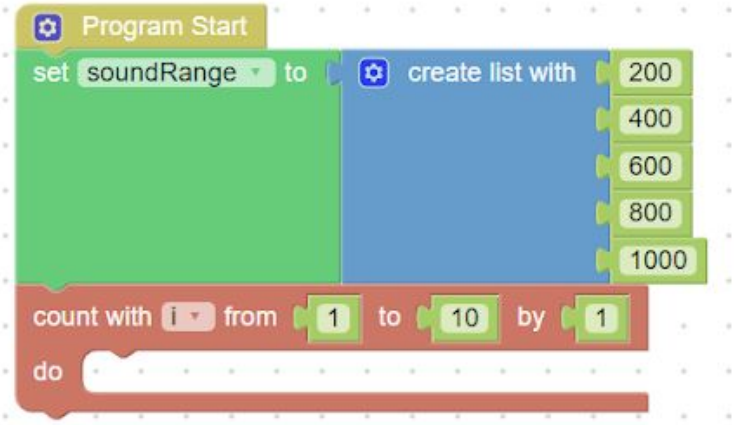
Program MiRo to play a range of sounds in the MiRo simulator

part (a)

Step	Image
1 – Set Up • Add **'Program Start'** from **'Setup'.**	
2 – Create a Variable • Create the variable **soundRange** from the **Variables** icon.	
3 – Add Variable • Add **set [variable] to** from **Variables.** • Connect to the **Program Start** block. • Set variable to **soundRange**	
4 – Add a list • Add **create list with** from **Lists.** • Connect to **set soundRange to** block. • Click the settings icon and add two more items to the block. *Drag a new item from the left to under the others on the right.*	
5 – Create list items • Add **[0]** blocks x 5 • Connect them to the end of the **create list with** block. • Set, in order to: **200**, **400**, **600**, **800**, **1000**	

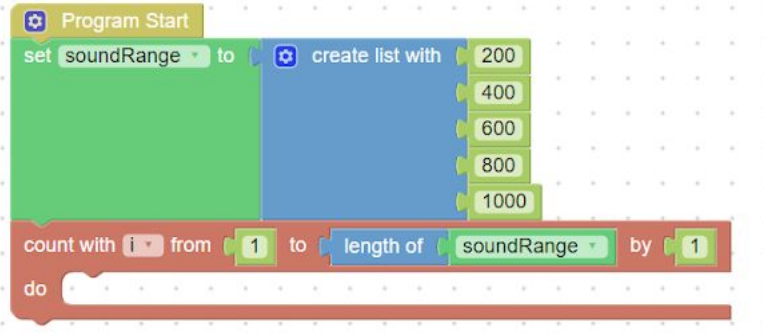
6 – Add loop

- Add **count with [i]** block from **Loops.**
- Connect to the bottom of the list.



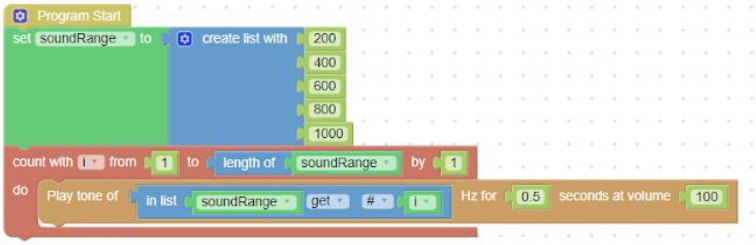
7 – Set the loop to iterate through the sounds

- Add **length of** block from **Lists.**
- Connect it on top of where the number 10 is on the loop block.
- Add **soundRange** from **Variables.**
- Connect it at the end of the **length of** block.



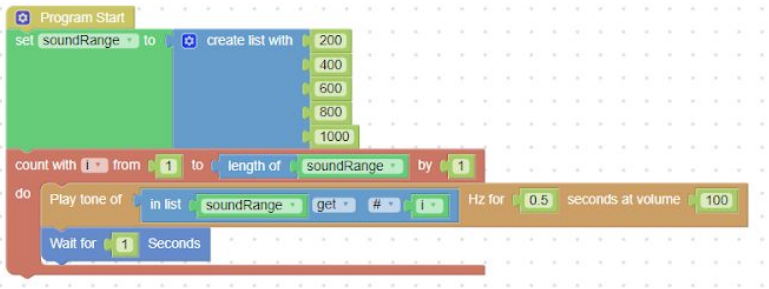
8 – add sound block to play list item

- Add **play tone** block from **Voice.**
- Connect it inside the loop.
- Add **in list [ ] get #** block from **Lists.**
- Connect it on top of the number 500.
- Set the list name to **soundRange**



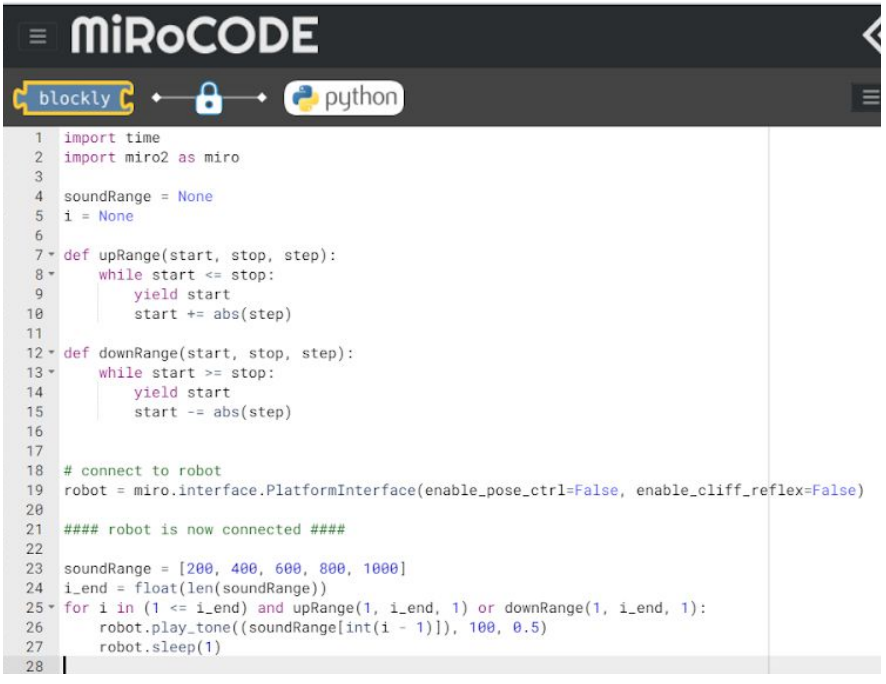
9 – add a wait

- Add **wait for 1 second** block from **Time.**
- Connect under the **play tone** block.



10 – Simulate Click ‘Simulate and you will hear MiRo play the range of tones set within the list. Click Cancel to stop the simulation. To start the simulation again you can reset the world.	
Extend	1. The range of the sound is 200-2000 can you create another musical range 2. Can you have the musical range activate when MiRo hears a clap or is touched?

part (b)

Step 1 - Flip to Python code by clicking on the **python** button. **Can you find the list?** **What code defines a function?**	 ``` import time import miro2 as miro soundRange = None i = None def upRange(start, stop, step): while start <= stop: yield start start += abs(step) def downRange(start, stop, step): while start >= stop: yield start start -= abs(step) # connect to robot robot = miro.interface.PlatformInterface(enable_pose_ctrl=False, enable_cliff_reflex=False) #### robot is now connected #### soundRange = [200, 400, 600, 800, 1000] i_end = float(len(soundRange)) for i in (1 <= i_end) and upRange(1, i_end, 1) or downRange(1, i_end, 1): robot.play_tone((soundRange[int(i - 1)]), 100, 0.5) robot.sleep(1) ```
Step 2 - Locate the list items. Click the lock icon between the blockly and python buttons. The screen will go black for editing.	soundRange = [200, 400, 600, 800, 1000]

Step 3

Add to the list items using a comma and increasing the number by 200 each time.

```python
import time
import miro2 as miro

soundRange = None
i = None

def upRange(start, stop, step):
    while start <= stop:
        yield start
        start += abs(step)

def downRange(start, stop, step):
    while start >= stop:
        yield start
        start -= abs(step)


# connect to robot
robot = miro.interface.PlatformInterface(enable_pose_ctrl=False, enable_cliff_reflex=False)

#### robot is now connected ####

soundRange = [200, 400, 600, 800, 1000, 1200, 1400, 1600, 1800]
i_end = float(len(soundRange))
for i in (1 <= i_end) and upRange(1, i_end, 1) or downRange(1, i_end, 1):
    robot.play_tone((soundRange[int(i - 1)]), 100, 0.5)
    robot.sleep(1)

```

Step 5

Run your program

Experiment with the settings and investigate:

- Change the list items to find a suitable sound range
- Change the time set within robot.sleep(1) to experiment with the time delay between sounds.