Using a micro:bit - Python Recap:

## Set Up

1. Make sure that your micro:bit is plugged in
2. To start writing programs for your micro:bit, go to: <https://python.microbit.org/v/2>

Now you should be all set to write programs!

Try doing each of the tasks in order to build up your program and make it more interesting!

## Displaying Text

**display.scroll(‘Put your message here’)**

Text

Description automatically generated

## Displaying Images

**display.show(Image.Happy)**

(You can find other preset images at <https://microbit-micropython.readthedocs.io/en/v1.0.1/image.html#attributes> )

## Text Description automatically generated

## Making Your Own Images

Copy this template and put it before the while True line.

**image = Image("00000:"**

**"00000:"**

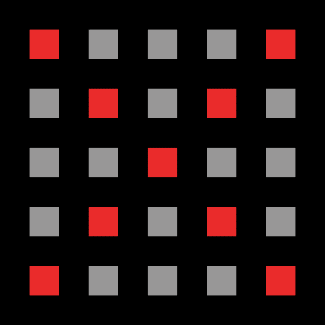
**"00000:"**

**"00000:"**

**"00000")**

Then if you want a light in the grid to turn on, change the 0 to a 9

Example:

**image = Image("90009:"**

**"09090:"**

**"00900:"**

**"09090:"**

**"90009")**

To display this image, put image inside **display.show()**

Text

Description automatically generated

## Button Presses

You can make things happen when you press buttons by using **if statements**.

This is where the A and B buttons are on your micro:bit:

## Calendar Description automatically generated

Text

Description automatically generated

Put the code that you want to happen after pressing the A or B button underneath the if button\_a.is\_pressed(): line

Make sure you look closely to see how many TAB spaces you need before each line!

When you make a new line under a while or if line, you need to add an extra TAB for the new line and all of the lines in that block

## Playing Sound – Connecting the Speaker

Before you can start playing sound, you’ll need to connect your speaker to the micro:bit

A picture containing indoor

Description automatically generated

Make sure that the wires are plugged in on the same row (don’t make it the top one that says GND – do any of the rows below it)

(You might have to use crocodile clips to attach them instead if you can’t get it working)

## Playing Sound – Writing the Code

You need to add import music so you can use the extra features.

Text

Description automatically generated

Then you can use music.play to play music! Like with the images, you can either choose a preset tune or write your own! Try testing that the speaker works with a prewritten one first before you test out your own melody!

## Prewritten Music

Use the code below as an example – you can find other tunes at <https://microbit-micropython.readthedocs.io/en/latest/music.html#built-in-melodies>

Text

Description automatically generated

## Custom Music

Your music tune is a **list** made of individual **notes**.

These notes tell the program what note to play, and for how long.

The name of the note (A-G)

Could have a # after it (like C#) if its one of the black keys on the piano

The octave number. A3 is a lower note than A4. ‘Middle C’ is C4

The duration of the note, this example is 4 beats.

You can find the write note name and octave number using an online piano such as <https://www.apronus.com/music/flashpiano.htm>

You can make a tune from a list of notes by putting these notes in a sequence in square brackets [ ]

Make sure you call the list something, like notes like below:

notes = [a3:4, b3:4, c3:4, d3:4]

Then you can play the music using your list name:

Text

Description automatically generated

## Troubleshooting

1. Have you imported the microbit package into python? Check that your code starts with **from microbit import \***
2. Do you have a **while True:** loop? This needs to be at the start of its line, with no spaces before it (it can have Enter spaces, just not space bar spaces or tabs).
3. Are all of your statements indented properly? Each line inside a block needs to start with a TAB before you write the code. You add another TAB for each block you are in (see below)

A screenshot of a computer

Description automatically generated with medium confidence