Micro:bit Musical Instruments

# Materials:

* BBC Microbit
* Piezo Speaker? (Might be optional as the microbit has an inbuilt speaker apparently), or headphones
* Crocodile clips to connect the external speaker
* Micro-USB to connect to the web editor
* Optional buttons and lights to add on

# Step 1: Research & Design (25th Nov)

Look at examples of both electronic musical instruments and real-world ones. What properties do you want your musical instrument to have? Are there any particular instruments out there that you want to take inspiration from?

Examples of Digital Instruments: Synths, Theremin, Synthetic Instruments

Examples of Real-World Instruments: Strings, Brass, Percussion, Woodwind

# Step 2: Learn how to use Micro bits (2nd Dec)

Not sure how much understanding the students have with using micro bits, so we could have a tutorial to the basics of programming the micro bits, and beginning to cover some relevant aspects for their project (e.g. adding sound, involving inputs such as buttons, tilt controls etc.) Could use the PRIMM approach here with existing programs.

## How to set-up a micro bit:

### Connecting a speaker or headphones

<https://support.microbit.org/support/solutions/articles/19000101901-connecting-headphones-or-a-speaker-to-the-micro-bit>

### Connect your micro:bit to your computer or mobile device.

* If you're using a computer, you need a micro USB cable to connect to your micro:bit to your computer's USB socket
* If you're using a phone or tablet, use Bluetooth to connect your micro:bit wirelessly

### Transfer from a computer

* Transferring your program to your micro:bit is called flashing because it copies your program into the micro:bit’s flash memory.
* Your micro:bit will pause and the yellow LED on the back will blink while your program is being transferred. Once it’s copied across, your program starts running on your micro:bit.
* There are two ways to transfer your program from a computer:
* Drag and drop is like copying a downloaded file from your computer to a USB memory stick. It works on any computer.
* Direct flashing sends your program from the code editor direct to your micro:bit. It works on any computer in two popular web browsers.
* When you plug the micro:bit into your computer's USB socket, it will appear on your computer like a USB memory stick called MICROBIT.
* Download your program as a .hex file from the code editor to your computer, usually to your downloads folder. Then drag and drop the .hex file on to the MICROBIT drive.
* After you transfer your .hex file, the MICROBIT drive will disconnect and reconnect as the micro:bit resets. The .hex file will not be listed on the MICROBIT drive after this. This is expected. Your micro:bit is not a flash storage device, but your computer shows it as one to make it easy to transfer .hex files.

### LED’s and Buttons

<https://microbit.org/get-started/first-steps/leds-and-buttons/>

## Microbit Coding Editor

<https://python.microbit.org/v/2>

# Step 3: Make (9th and 16th Dec)

Set of the students (in teams?) to work on their own instrument creation. Motivate them with awards for Most Creative, Best Instrument etc. Offer assistance when they get stuck with the more complicated aspects.

# Awards (After Christmas)

Could host a performance/demo of each instrument to the other students/groups, and present the awards. Dan to buy prizes.