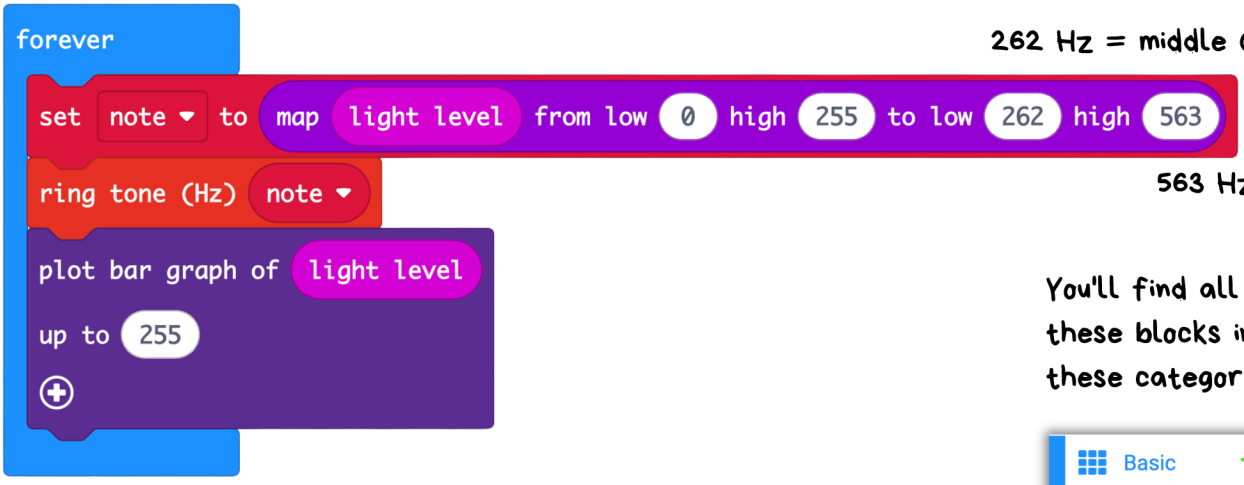
Light theremin

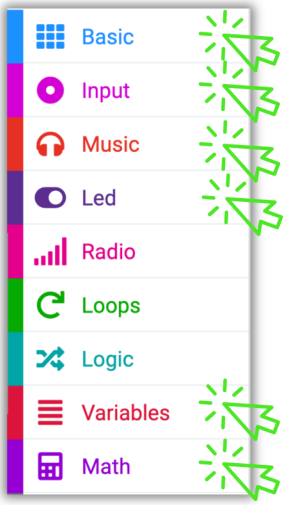
The micro:bit does not have a light sensor but we use the LEDs (Light Emitting Diodes) in the matrix to measure the ambient light.

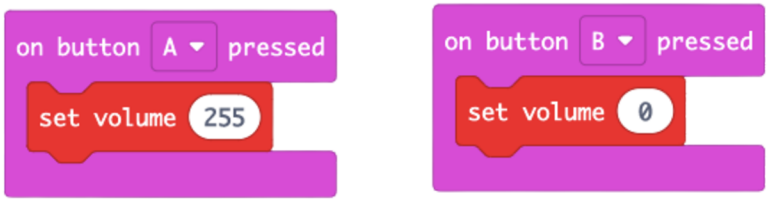
To make a light theremin...

Make a variable called 'note'.
Use a math block to map low light level (0) to 262 Hz (middle C) and high light level (255) to 563 Hz (C#)



You'll find all these blocks in these categories:

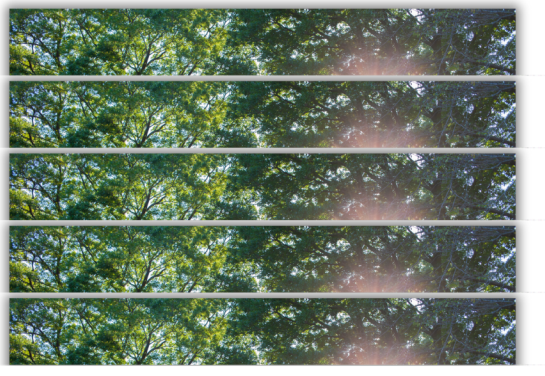




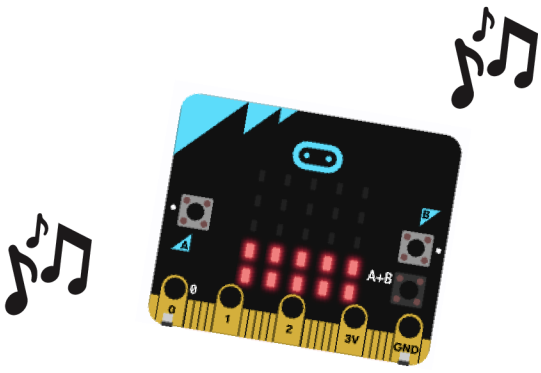
Sound is measured in Hertz and is written Hz. 262 Hz = middle C.

Use the A button to turn the theremin 'on' and the B button to mute your theremin.

Take your micro:bit outside and see what happens when you move around between shade and light areas.



Can you make this into a game?



Light theremin 2

This theremin will set a different note for each 'light level' and display corresponding LEDs.

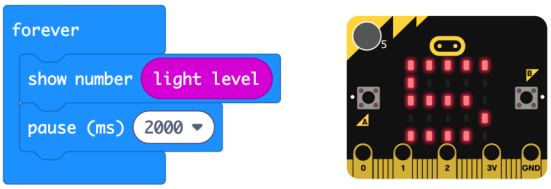
How does the micro:bit sense light?

A sensor is something that detects or measures something and responds to it.

The micro:bit doesn't have a separate light sensor - it uses the LEDs on the front to measure light! Here's how...

LEDs are made from a special material (called a semiconductor) that can also generate a tiny electrical signal when light hits them - a bit like a mini solar panel. The micro:bit reads this signal to find out how bright or dark it is.

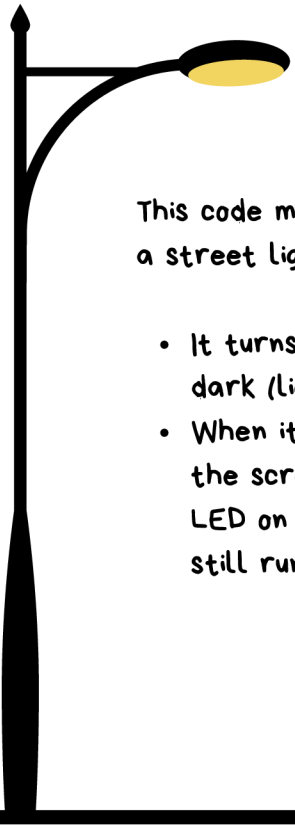
In the Microsoft MakeCode editor, you can use this code to give you a light level reading:



This block returns a number between 0 (dark) and 255 (bright).

You can show changing light levels using the 'plot bar graph' block:

Try covering your micro:bit with your hand or shine a torch on it. Watch how the bar graph changes.



This code makes your micro:bit act like a street light.

- It turns the LEDs on when it's dark (light level below 128).
- When it's bright again, it clears the screen - but keeps one dim LED on to show that the code is still running.