

Import

The import statements load functionality into Python. They should be the first lines of your program.

```
from microbit import *
from neopixel import NeoPixel
import random
import music
import radio
```

Buttons & pins

The two buttons and 3 pins detect user input

```
if button_a.was_pressed():
if button_b.was_pressed():
if pin0.is_touched():
if pin1.is_touched():
if pin2.is_touched():
```

Pause

Pause the program so the user has time to see what is happening or to respond.

```
sleep(2000)
```

Display

Control the 5x5 LED display.

```
display.scroll("text")
display.scroll("text", delay=75) # double the speed
display.show(Image.HAPPY)
display.show(Image("00000:00000:00000:00000:00000"))
display.clear()
```

Gestures

Detect basic types of movement. Valid gestures are

- "up", "down", "left", "right", "face up", "face down", "freefall", "3g", "6g", "8g", "shake".

```
if accelerometer.current_gesture() == "shake":
    display.scroll("Shake your booty")
elif accelerometer.was_gesture() == "up":
    display.scroll("You lift me up")
```

Variables & calculations

Create a variable

```
q = 10 # create q with value 10
```

Operations (change the value of a variable)

```
q = r + s # r add s, save to q
q = r - s # r subtract s, save to q
q = r * s # r times s, save to q
q = r / s # r divided by s, save to q
```

Conditions (does a variable match a certain value?)

```
if r == q: # is r equal to q?
if r > q: # is r bigger than q?
if r < q: # is r less than q?
if r != q: # is r not equal to q?
```

Boolean variables ("True" or "False" variables)

```
keepgoing = True
while keepgoing:
    do_something
```

Random

Generate a random number between 0 and 99

```
import random
r = random.randint(0, 100)
```

Control structures

Decide which code to run.

```
if condition_is_met:
    do_this
    do_this
    do_this
elif other_condition_is_met:
    do_this_other_thing
    do_this_other_thing
    do_this_other_thing
else:
    last_resort_thing_to_do
    last_resort_thing_to_do
    last_resort_thing_to_do
always_do_this
always_do_this
always_do_this
while condition_is_met:
    do_this_thing_repeatedly
    do_this_thing_repeatedly
loop_is_over_with
```

For more Python & Microbit resources visit

<https://youtube.com/pbaumgarten>
<https://pbaumgarten.com/python>

Bluetooth radio

Communicate with another Microbit
Must be on the same channel (from 0 to 83)

```
from microbit import *
import radio

radio.config(channel=10)
radio.on()
while True:
    display.clear()
    if button_a.was_pressed():
        radio.send("iheart")
    if button_b.was_pressed():
        radio.send("upset")
    incoming = radio.receive()
    if incoming == "iheart":
        display.show(Image.HEART)
        sleep(2000)
    elif incoming == "upset":
        display.show(Image.SAD)
        sleep(2000)
    else:
        display.show(".")
        sleep(100)
        display.show("")
        sleep(100)
```

Neopixels

Programmable multi colour LEDs that attach to the microbit.

- White wire to pin 0 (or change code)
- Red wire to 3V pin
- Black wire to GND pin

Create the Neopixel object

Replace the 8 with the number of LEDs you have

```
from neopixel import NeoPixel
leds = NeoPixel(pin0, 8)
```

Create your colours (refer to common colours table)

Colour values are red, green, blue values with range from 0 (fully off) to 255 (fully on).

```
white = [255, 255, 255]
blue = [ 0, 0, 255]
red = [255, 0, 0]
green = [ 0, 255, 0]
magenta = [255, 0, 255]
black = [ 0, 0, 0]
```

Set individual LEDs to a colour.

Remember: The first LED is 0, the second LED is 1, the third LED is 2 etc.

```
leds[0] = white
leds[1] = blue
leds[2] = red
leds.show() # don't forget this
```

Full example

```
from microbit import *
from neopixel import NeoPixel
leds = NeoPixel(pin0, 8)
white = [255,255,255]
black = [0,0,0]
n = 0
while True:
    for q in range(8):
        leds[q] = black
    leds[n] = white
    leds.show()
    n = n + 1
    sleep(200)
    if n >= 8:
        n = 0
```

R	G	B
0	38	133
66	154	223
77	190	253
76	14	119
94	83	190
126	119	210
205	30	16
252	0	127
254	121	200
118	57	49
241	171	0
250	223	0
0	0	0
0	126	58
100	200	62

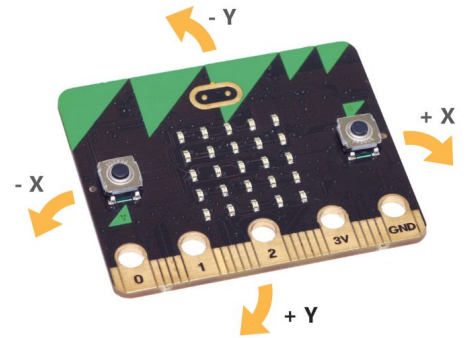
Accelerometer

Detect movement.

X axis - tilting from left to right.

Y axis - tilting forwards and backwards.

Z axis - moving up and down



```
x,y,z = accelerometer.get_values()
if x < -50:
    display.scroll("tilting left")
elif x > 50:
    display.scroll("tilting right")
```

Music

Play music. Assumes speaker connected to pin0 and GND.

Each note has three parts as follows:

note**octave**:**duration**

Note is one of a,b,c,d,e,f,g. F-sharp would be f# and A-flat would be Ab. The note "r" indicates rest (silence).

Octave starts at 4. Remembers each time you change it and applies until next change.

Duration starts at 4. Remembers each time you change it and applies until next change.

```
import music
```

```
tune = ["f#:2", "f#:4", "d", "b3",
        "r:2", "b3:4", "r:2", "e4:2",
        "r:2", "e:2", "r:2", "e:2", "r:2",
        "g#:2", "g#:4", "a", "b", "a:2",
        "a:2", "a:2", "e:4", "d", "f#:2",
        "r:2", "f#:2", "r:2", "f#:2",
        "e:4", "e", "f#", "e"]
```

```
music.set_tempo(bpm=168, ticks=4)
music.play(tune, wait=False)
```

Speech

Your Microbit will talk back to you if you let it

```
import speech
speech.say("Hello, I am your microbit", speed=92, pitch=60, throat=110, mouth=190)
```