## Lesson 4 -Activity Sheet

## Task 1 - Getting Started

Look at the code below what is the difference, is there one? Discuss your findings with another student.

from microbit import \*

total = 3

while True:

if button\_a.is\_pressed():

display.scroll(total)

from microbit import \*

total = "3"

while True:

if button\_a.is\_pressed():

display.scroll(total)

In the second example program the 3 is enclosed in double quotation marks “3”. What does this do then? This means that the 3 is treated as a string: although it is the number 3 is does not have a value of 3, it is simply the symbol used to represent the number 3. Try out the code below:

from microbit import \*

total = "3" + "1"

while True:

if button\_a.is\_pressed():

display.scroll(total)

What happens? The program does not add the two numbers together, it simply places them together as 31. This is because they are strings. String use the “” or ‘’ symbols, double quotes or single quotes. Remove the quotes and change the numbers to 2 + 5, does the program work now? These are no longer strings, so they have value, the total displayed in your micro:bit will be 7.

## **Keeping a Running Total**

This program creates a **variable** called *total* and stores an initial value of 0 in the variable. Every time you press Button A it adds one to the total – **increments** it. For example, if you press Button A four times, then it will display the numbers 1, 2, 3, then 4 on the micro:bit. Each time you press Button B it subtracts 2 from the total. If the current total is 4 then the program subtracts 2 and displays the new total, 2.

It is possible to create negative numbers which are preceded by the – sign. Copy up the code below and experiment with the program.

from microbit import \*

total = 0

while True:

if button\_a.is\_pressed():

total = total + 1

sleep(100)

display.scroll(total)

elif button\_b.is\_pressed():

total = total - 2

sleep(100)

display.show(total)

elif button\_a.is\_pressed() and button\_b.is\_pressed():

total = 0

sleep(100)

display.show(total)

else:

display.show(Image.SMILE)

This program is a useful method for keeping a running total or a score in a game or a quiz. Think about how else you could use it.

## Success Criteria

1. Add two numbers stored as strings together
2. Convert the strings to numbers and add the numbers together
3. Create a program that keeps a running score
4. Adapt the program

## Pro-tip

If the program requires non-numerical responses such as ‘yes’ or ‘no’ then you can use breakto break out of the while True loop and move onto the next question or section of the program. Create a program that uses this method.

A question

while True:

if button\_a.is\_pressed():

display.scroll("yes")

sleep(2000)

break

elif button\_b.is\_pressed():

display.scroll("no")

sleep(2000)

break

Another question

while True:

if button\_a.is\_pressed():

display.scroll("yes")

sleep(2000)

break

elif button\_b.is\_pressed():

display.scroll("no")

sleep(2000)

break

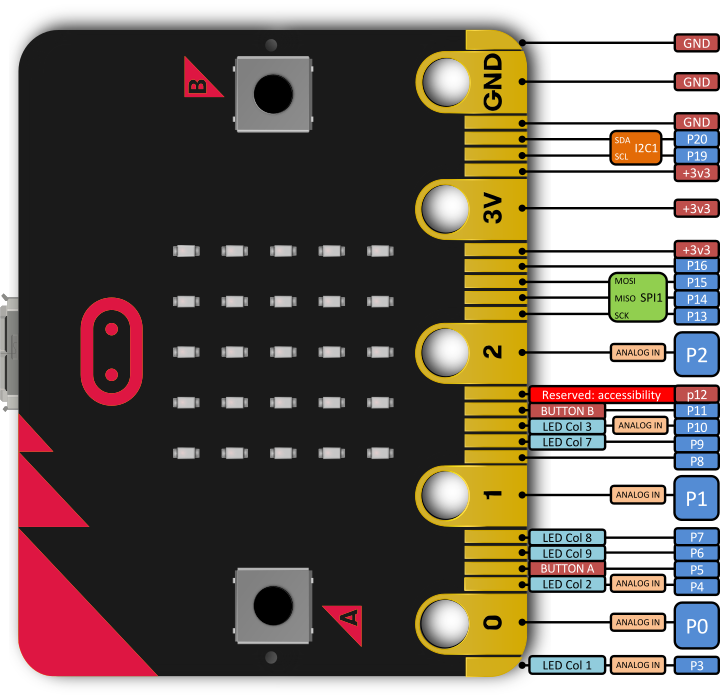
## Test Time

When writing the program code check the indentation levels as these can cause errors and stop the program functioning correctly.

## Stretch Tasks

* Add images and text to your program
* Share your program with your teacher of other learners

## Task 2 - Getting Started

The micro:bit has a number of **pins** built in which enable you to interact with the micro:bit through touch. You can add sensors such as a light sensor or a temperature sensor and write programs to respond to the readings. The micro:bit can also power LEDs and small motors.

**Using the Pins: The Shrinking Square**

In this first program we are going to write a code that responds to you touching your micro:bit. We can think of this as a kind of a touch sensor. The micro:bit displays a large square in the LED matrix and uses the code if pin0.is\_touched(): to check if you are touching Pin 0. Notice that selection is used which was introduced in Lesson 4, the if statement is used to check for the pin being touched. Copy out the code below, *you can add your own responses if you want to.*

# Add your Python code here. E.g.

from microbit import \*

while True:

if pin0.is\_touched():

display.show(Image.SQUARE\_SMALL)

else:

display.show(Image.SQUARE)

Download the program to your micro:bit and hold the GND pin in your right hand between your finger and thumb. The use the finger and thumb of your left hand the touch Pin 0. The square will shrink! You may find it easier to attach one crocodile clip to the GND pin and Pin 0 and then touch these two together. This works because we are making a **circuit** between the two pins. You can try and see how many other Learners you can connect in a chain. All hold hands and the first Learner holds the GND pin and the last the Pin 0.

**Looping**

The program uses a while True loop to keep **looping** round and checking for Pin 0 being pressed, which means that the program responds when you touch the pin and also when you break the circuit. We can think of a loop as a repeat, the program keeps repeating until we stop it

**Building an Alarm**

Alarms work by breaking a circuit which then triggers a response, usually a load siren or noise. Some alarms send notifications, trigger a camera to capture an image or even control a motor to lock a door. We can use the pins on the micro:bit to build a simple alarm system. The physical alarm works better with crocodile clips as you add a thin strip of tin foil between the two clips and attach the foil to a door, a drawer or across a book. When someone opens the door, drawer or book then the foil is torn which breaks the circuit and triggers the alarm.

while True:

if pin0.is\_touched():

**ADD THE CODE FOR THE ALARM AT REST**

else:

**ADD THE CODE FRO THE ALARM WHEN TRIGGERED**

## Success Criteria

1. The micro:bit display an image when the circuit is complete and the GND pin and Pin 0 are connected
2. A different image or text is displayed when circuit is broken
3. The alarm is set up to protect a door or a window

## Pro-tip

We can adapt the program and use the loop to add more responses. In the program below, we display both the large and small square images when the pin is touched, creating the effect that the micro:bit is flashing. We can then add a different image to the elsestatement – think of this as the image that is displayed when the micro:bit is at rest, the circuit is broken

# Add your Python code here. E.g.

from microbit import \*

while True:

if pin0.is\_touched():

display.show(Image.SQUARE\_SMALL)

sleep(500)

display.show(Image.SQUARE)

sleep(500)

else:

display.show(Image.CHESSBOARD)

## Test Time

When running the program ensure that the crocodile clips are securely connected to the Pin. This will ensure that you make a good contact and create a circuit.

Check that you have the pin0.is\_touched():responding to the correct action.This line of code is used for the action for when the alarm is on, it has not been triggered.

## Stretch Tasks

* Add images and text to your alarm responses
* Use the loop to build up an animation or a combination of responses when the alarm is triggered
* Remember to use the sleep(500)in between the events
* Add an additional wire onto Pin 1, this could stop the alarm if it has been triggered
* Add an elif statement to the program so that it responds if Pin 1 is used

from microbit import \*

while True:

if pin0.is\_touched():

**ADD THE CODE FOR THE ALARM AT REST**

elif pin1.is\_touched():

**ADD THE CODE FOR THE ALARM AT REST**

else:

**ADD THE CODE FOR THE ALARM AT REST**

* Combine the program code with button presses to add further responses.

## Final Thoughts

Variables are essential to programs so that data can be stored and used later on. This lesson has introduced how to combine selection, variables and combine the data stored in the variable. Either by joining the data together or adding it. You can use these skills in the Healthy Eating Quiz project.

You have just learned how to:

* Use the micro:bits pins
* Trigger events when the pins are apart and connected
* Use a loop to respond to the pin
* Build a simple alarm

Think about how you could use this program outside of the classroom. Remember there are many materials other than foil that conduct and could be used to make a circuit. Perhaps you could make a bath alarm that triggers when the bath is full? How would that work? How about adding an alarm sound? Research how to play music and sounds through your micro:bit.