## Lesson 1 – Activity Sheet

## Task 1 - Getting Started

Navigate to [https://makecode.microbit.org](https://makecode.microbit.org/) and use MakeCode to make equivalent programmes to the Python code below. You may recognise some of these programmes!

## Success Criteria

## In the table below, look at the Python examples and work out the blocks equivalent

## Pro-tip

## Look for the key words like ‘while’ and look for them in MakeCode. Most of these are fairly easy to find in MakeCode.

## 

|  |  |
| --- | --- |
| **Python code** | **Blocks equivalent** |
| from microbit import \*  while True:  if button\_a.is\_pressed():  display.show(Image.HAPPY)  elif button\_b.is\_pressed():  display.show(Image.SAD)  else: display.show(Image.YES) |  |
| from microbit import \*  while True:  display.scroll('I love micro:bit')  display.show(Image.HEART)  sleep(2000) |  |
| from microbit import \*  pattern1 = Image('09090:'  '90909:'  '09090:'  '90909:'  '09090')  display.show(pattern1) |  |
| from microbit import \*  my\_str = ('Hello')  my\_int = 35  display.show(my\_int) |  |

## Stretch Tasks

## What does the following python make?

|  |  |
| --- | --- |
| from microbit import \*  import random  while True:  if button\_a.was\_pressed():  int1 = random.randint(0,10)  int2 = random.randint(0,10)  display.scroll(int1)  display.scroll('x')  display.scroll(int2)  display.scroll('= ?')  elif button\_b.was\_pressed():  display.scroll(int1\*int2)  else:  display.show('<-Q A->') |  |

## Task 2 - Getting Started

Use the micro:bit to create a message display system for a topic of your choice. For example, a weather report, a sports scoreboard, a friendship message, online gaming results.

Log into your computer and go to

You will be presented with the text and image code.

Graphical user interface, text, application, chat or text message

Description automatically generated

Change the ‘Hello, World!’ to a word, name or phrase of your choice.

Now let’s try the program on your micro:bit

Click the **send to microbit** button

Graphical user interface, text, application, chat or text message

Description automatically generated

This will save your file onto the micro:bit.

Alternatively, if you wish to save your program onto your computer, click on the save option. This will allow you to edit the program in future.

Graphical user interface, text, application, chat or text message

Description automatically generated

The LED on the back of the micro:bit will begin to flash as your program is written. Once completed the program will run.

## Success Criteria

1. What text appears on the micro:bit?
2. What image appears?
3. Change the text so that it is related to your ‘message display’, this could be an announcement, a weather report, or you name.
4. Change the image to one that is related to your new text.
5. You can read more about images here: <https://microbit-micropython.readthedocs.io/en/latest/tutorials/images.html>

## Pro-tip

Check that you are using the correct indentation, this is the space at the beginning of the lines of code.

from microbit import \*

while True:

display.scroll('Hello, World!')

display.show(Image.HEART)

sleep(2000)

They must be aligned in order for the program to work correctly.

## Test Time

This activity has introduced the lines of code to, scroll text and display images. To test and run your programs, download the program onto your micro:bit.

## Stretch Tasks

* What happens if you change line 8 to sleep(200)?
* What happens if you change line 8 to sleep(8000)?
* Include several different lines of text

while True:

display.scroll('Hello, World!')

display.scroll('Morning, World!')

display.scroll('Afternoon, World!')

display.show(Image.HEART)

sleep(2000)

* Include several images
* Alternate the display so that it scrolls text and then displays an image and then scrolls text
* Use these lines of code to build your message display machine
* Create a name badge that scrolls your name, then displays an image and then an interesting fact about you

## Final Thoughts

In this lesson you have started to take the first steps into using Python to programme a micro:bit. Whilst these examples may seem complicated, you have done all of these things before using blocks and so now you just ned to get confident with doing the same thing but using Python. Python can do much more than MakeCode and whilst it may seem daunting at first you will soon appreciate the power it gives you to achieve your coding goals.

Using Python, you have just learned how to:

* Download a program to your micro:bit
* Scroll text
* Display images
* Scroll and display multiply text and images