A Python BBC Microbit Project: ‘Meriden Robot’

Introduction

https://bit.ly/36v5M4a

In this project you will program your micro:bit to control a Robot using Python Code.

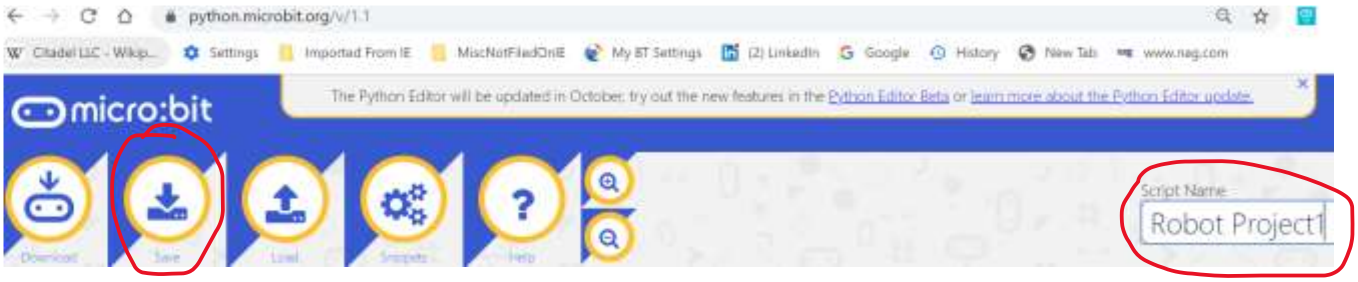
Getting Set-up to program the Meriden Robot (v0.3.2)

Using Internet Explorer *(not Chrome)*

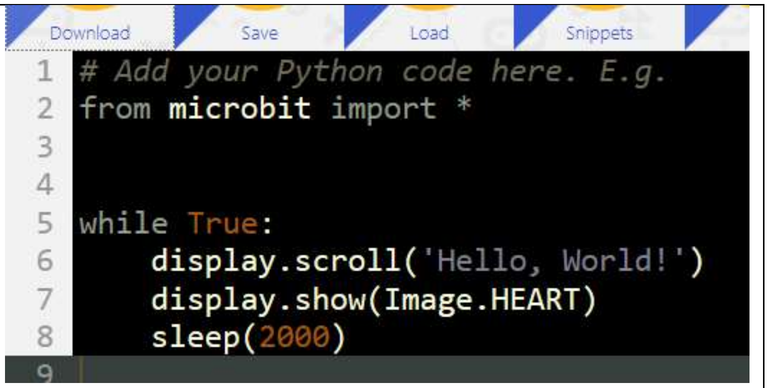
search for [Python Editor for micro:bit](#) or

go to www.microbit.org Select Create Code or Let’s create code. You need the Python Editor.

At the top of the screen, in the right, change the name of your script to Robot Project1. We will save the work later.


python.microbit.org/v/1.1
micro:bit
The Python Editor will be updated in October: try out the new features in the Python Editor Beta or learn more about the Python Editor update.
Download
Save
Load
Snippets
Help

Script Name: Robot Project1

Delete lines 5, 6, 7 & 8	

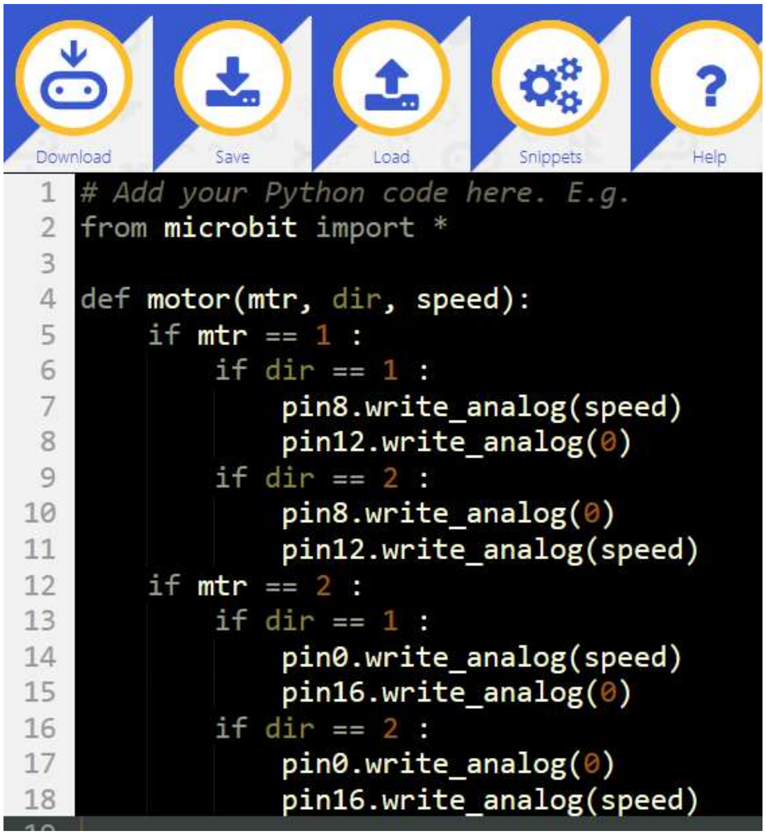
```
1  # Add your Python code here. E.g.
2  from microbit import *
3
4
5  while True:
6      display.scroll('Hello, World!')
7      display.show(Image.HEART)
8      sleep(2000)
9
```

We will now create the code that controls the two motors, controlling the two wheels of the robot.

Open another TAB in your web browser https://bit.ly/36v5M4a you should now get an electronic copy of these instruction sheet

Now highlight the text below and then copy (CTRL C) the following lines of code from this document and paste (CTRL V) into your Python Project. Note if the idents do not transpose you will need to fix those yourself.

```
def motor(mtr, dir, speed):
    if mtr == 1 :
        if dir == 1 :
            pin8.write_analog(speed)
            pin12.write_analog(0)
        if dir == 2 :
            pin8.write_analog(0)
            pin12.write_analog(speed)
    if mtr == 2 :
        if dir == 1 :
            pin0.write_analog(speed)
            pin16.write_analog(0)
        if dir == 2 :
            pin0.write_analog(0)
            pin16.write_analog(speed)
```



```
# Add your Python code here. E.g.
from microbit import *

def motor(mtr, dir, speed):
    if mtr == 1 :
        if dir == 1 :
            pin8.write_analog(speed)
            pin12.write_analog(0)
        if dir == 2 :
            pin8.write_analog(0)
            pin12.write_analog(speed)
    if mtr == 2 :
        if dir == 1 :
            pin0.write_analog(speed)
            pin16.write_analog(0)
        if dir == 2 :
            pin0.write_analog(0)
            pin16.write_analog(speed)
```

- There are two motors: show above as 'mtr' – 1 or 2
- Wheel directions forward as 1, backwards as 2
- Speed can be in the range of 0 to 1000

(1) Meriden Robot – First Movements

OK. Now we can begin properly. Add the following code…

You do not have to add the commented code (anything prefixed with a #)

```
# these first three lines make sure when we 'plug in' the motors are stopped for 6 seconds
motor(1,1,0)            # (1 = motor 1, 1 direction, 0 speed)
motor(2,1,0)            # (1 = motor 2, 1 …….
sleep (6000)            #6000 milliseconds = 6 seconds
# now we tell both motors to move for 15 seconds,
motor(1,1,400)          # (1 = motor 1, 1  direction, 400 speed)
motor(2,1,400)
sleep (15000)           #15 seconds
# now we stop the motors
motor(1,1,0)
motor(2,1,0)
```

<u>Save your work…</u>

giving it a useful file name e.g. Robot Project1

Load/Save,

Download Python Script

Save As (Save As can be found from the Save button – click on the upside down triangle)

Save your file to your preferred location, ideally to your network drive where you can recall this when working from a different PC/laptop]

Meriden Code Clubbers:

Search for Network and Save to your I: Drive (User/Personal Drive)

Get the Hex File onto the Microbit

Download Project Hex*

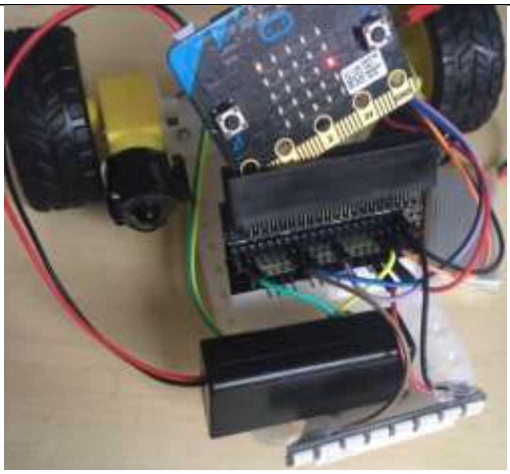
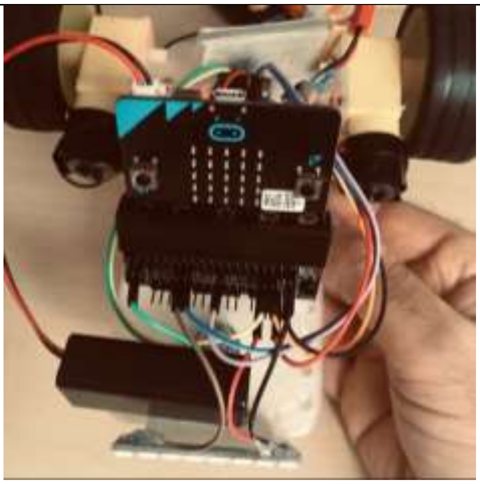
Save As

Save the Hex directly onto the Microbit

Meriden Code Clubbers:

The Microbit is usually your D: drive

Now fit the Microbit to the Robot as shown below

Put robot on the floor, next fit motor battery (you do not need the Microbit battery)

……….. if you've done everything correct your Robot will move and you have done your first project 'First Movements'

CONGRATULATIONS

(2) Meriden Robot – Take Control

We would like you use the code you have written in *(1) First Movements* to use buttons to

- Test the hex file has downloaded successfully and has no Syntax errors, by displaying images
- Start the motor and move the robot

In the real world when programming you are not given worksheets, so we are going to encourage you to work in a similar manner and search the internet or read the online manual.

Search for BBC micro:bit MicroPython documentation (or click on the hyperlink)

Look for the Buttons tutorial, read the Example

Need a hint.... The code you will is found in the Handling an Event section of the Buttons tutorial

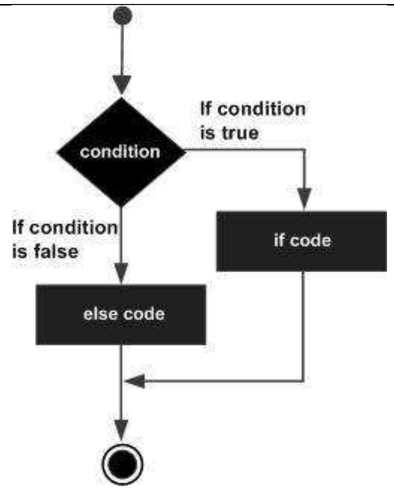
YOUR TASKS:

- Make your robot display* a smiley face if button a is pressed
  - Reuse bits of your code from (1) to ensure the wheels do NOT move
- Make your robot display* a different image and move and if button b is pressed
  - Simply reuse the code you made in *(1)* to move the Robot and then stop
- Else..... make your robot display* a sad face
  - Reuse bits of your code from (1) to ensure the wheels do NOT move

*image is displayed on the microbit

You will use the two lines of code from 'First Movements' within a "While true:....if............elif..... else:"

```
if something is True:
   # do one thing
elif some other thing is True:
   # do another thing
else:
   # do yet another thing.
```



You will need to read the BBC micro:bit MicroPython documentation again to find a list of images and the command. Pick an image you have not used before!

- Save your work,   [Save As to your location]
- Download the Hex to the Microbit
- Plug the Microbit into your Robot and test out your code

(3) Meriden Robot – 'More movements'

In your second project 'Take Control' you made it easier to get the Robot started and check your hex files were downloading OK. You also learnt to refer back to BBC micro:bit MicroPython documentation

3.1 More Robot Movements

You are now going to experiment with the two robot motors and their commands

One motor controls the left wheel and the other the right. You now know making the speed 0 stops the wheel(s)!

motor(1,1,500) - this means Motor 1, Motor Direction 1, Motor Speed 500

motor(2,2,500) - this means Motor 2, Motor Direction 2, Motor Speed 500

Motor Direction can be 1 or 2

3.2 Discuss with a friend what you think will happen before testing it

What happens if you

- change motor direction on one motor? (i.e. the two motor directions are different)
- make motor speed zero on one motor?
- change motor speed on one motor? (i.e. so the motor speeds are different)

3.3 Challenges:

- Can you make the robot make a bigger circle / smaller circle?
    - Experiment with the sleep function so it makes exactly one circle and not more or less
- Can you make the robot turn at 90 degrees (right angle)?
    - If so can you make it go forward, turn 90 degrees and go forward again for the same distance (and stop)
    - Take it a bit further… can you make it complete a
        - a square, ?
        - a triangle?

Note this may be a little bit fiddly so we don't expect it to be exact triangle or square, but you should be able to get close.

(4)Super Challenge

Using what you have learnt above can you reduce the amount of code and use a For loop

- Can you use a For Loop to
    - Make your Robot complete 1 circle, 3 circles,….
- Can you use a For Loop to
    - Make your Robot complete 2 circles
    - and then complete 3 triangles in the opposite direction (i.e. clockwise -> anticlockwise or vice versa)

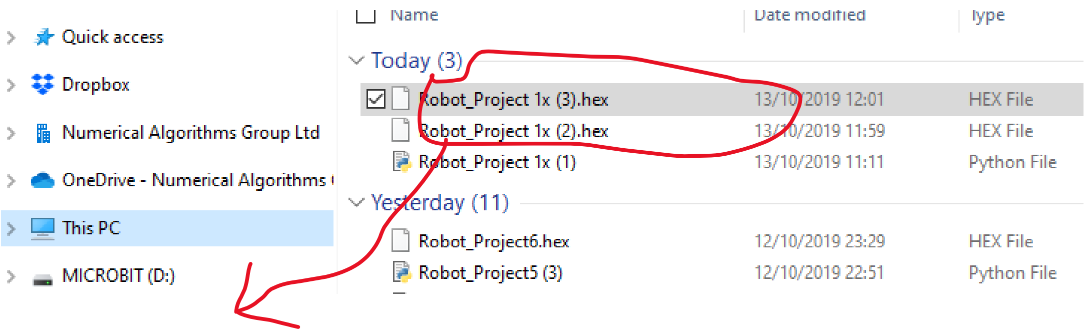
Written by

John Holden – @JohnCHolden , Jon Rogers, Worksheet: Python_Microbit_Robot_Vn0.3.2

APPENDIX – code or training notes not needed – delete in final version!

*ALTERNATIVELY

- Download the Hex file to your preferred location
- Open folder
- Drag and Drop the hex file you have just saved and downloaded to your PC onto the Microbit (D:)


Quick access
Dropbox
Numerical Algorithms Group Ltd
OneDrive - Numerical Algorithms
This PC
MICROBIT (D:)

☐ Name	Date modified	Type
Today (3)		
☑ Robot_Project 1x (3).hex	13/10/2019 12:01	HEX File
Robot_Project 1x (2).hex	13/10/2019 11:59	HEX File
Robot_Project 1x (1)	13/10/2019 11:11	Python File
Yesterday (11)		
Robot_Project6.hex	12/10/2019 23:29	HEX File
Robot_Project5 (3)	12/10/2019 22:51	Python File

My program that does the loops is Robot_Project3b_RightAngle

Model solution to up to stage 2 ->

Set the wheels moving by pressing B, then press A it does not stop – A will only work if pressed after the task assigned to B complete…. IS THAT RIGHT?

So task B runs, finishes with the else command and then B can be pressed – can't interrupt it

Need a reminder on For Loops

Check out:

https://snakify.org/en/lessons/for_loop_range/

or turtle race…