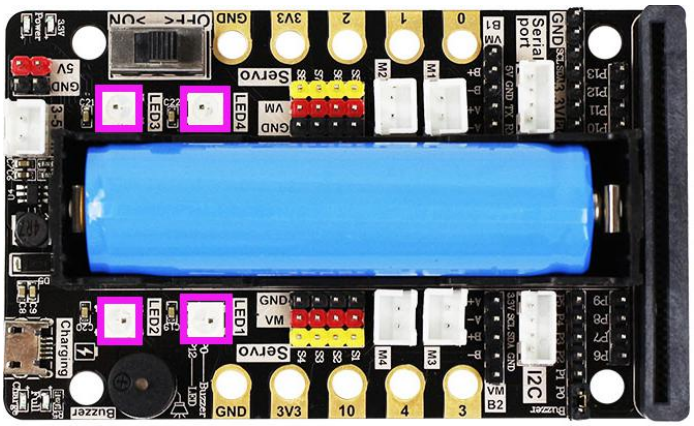
Control all RGB lights

1.Learning goals

In this course, we mainly learn how to control all RGB lights on the super:bit expansion board through Python programming. And we will realize some game play. Four RGB light is located on the expansion board as shown in the figure below.



2.Code and analysis

The program for this course, please view .py file.

```
from microbit import *
import neopixel
```

First, we need to import the library needed for this lesson from micro:bit, neopixel library is dedicated to control RGB light;

```
Red = (255, 0, 0)
Orange = (255, 165, 0)
Yellow = (255, 255, 0)
Green = (0, 255, 0)
Blue = (0, 0, 255)
Dark_Violet = (148, 0, 211)
White = (255, 255, 255)

color = (Red, Orange, Yellow, Green, Blue, Dark_Violet, White)

display.show(Image.HAPPY)

np = neopixel.NeoPixel(pin12, 4)
i = 0
```

Define different RGB light colors.

display.show(Image.HAPPY): Display the smile pattern on the micro:bit matrix;

np = neopixel.NeoPixel (pin12, 4): RGB light initialization settings, a total of 4 RGB lights, connected to the P12 pin of the micro:bit board (you can check the hardware interface manual).

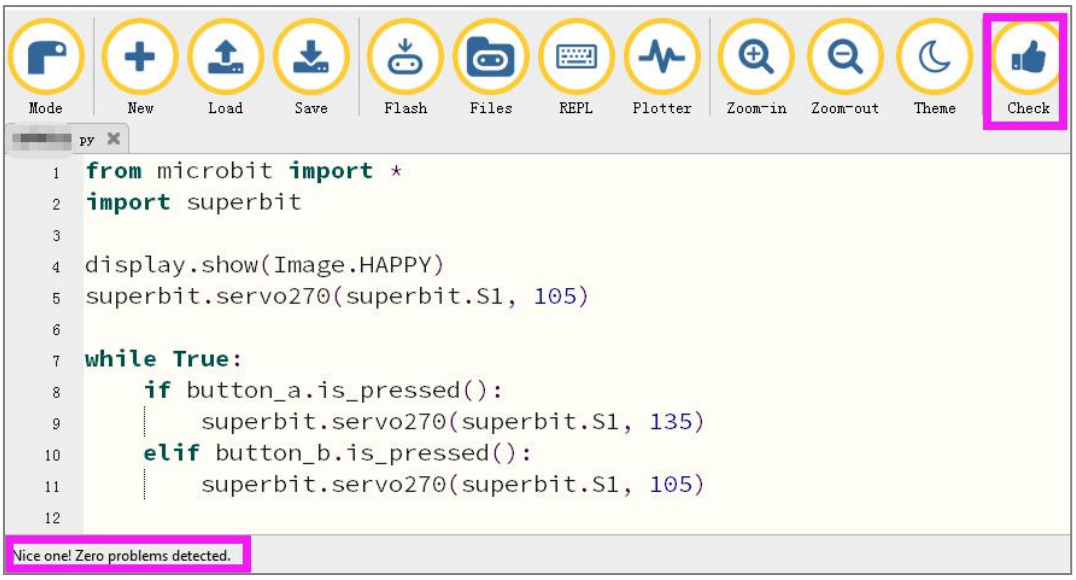
3.Writing and download code

1.You should open the Mu software, and enter the code in the edit window, , as shown below.

Note! All English and symbols should be entered in English, use the Tab key (tab key) to indent and the last line must be a space.

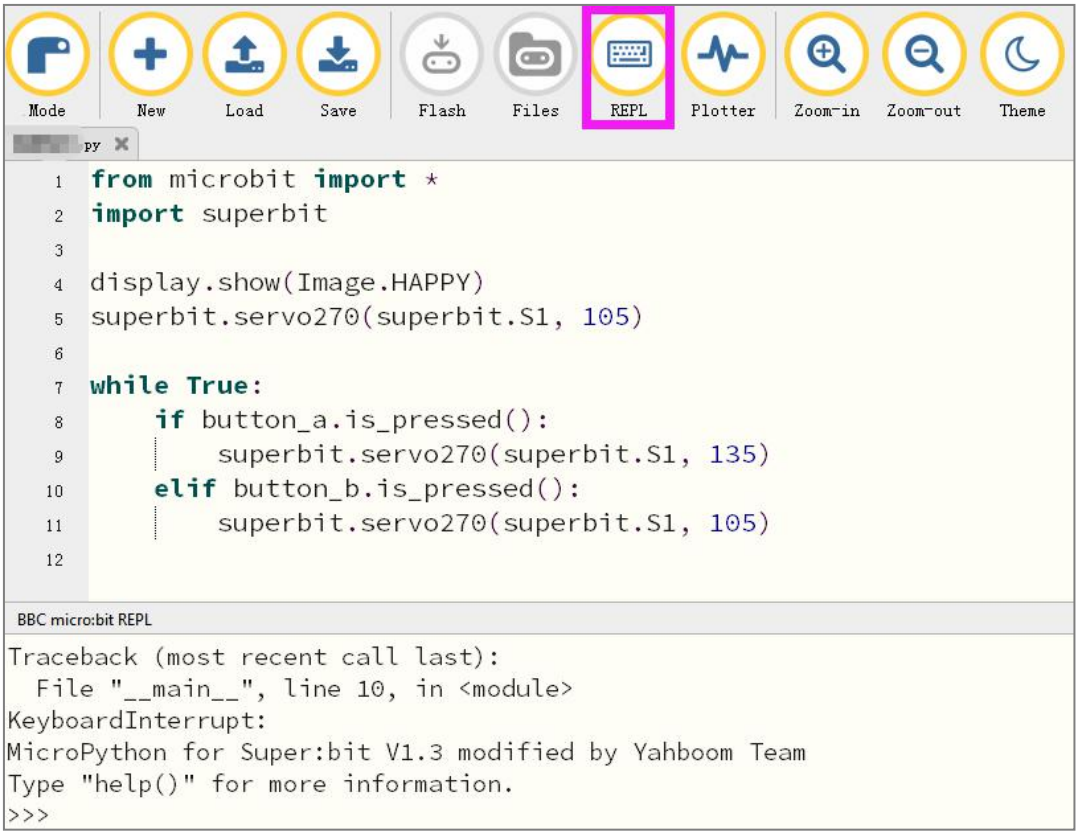
2.You can click the “Check” button to check if our code has an error.

If a cursor or underline appears on a line, it indicates a syntax error, please check and modify. If there is no error in the program, the bottom left of the interface will prompt that there is no problem in detection.

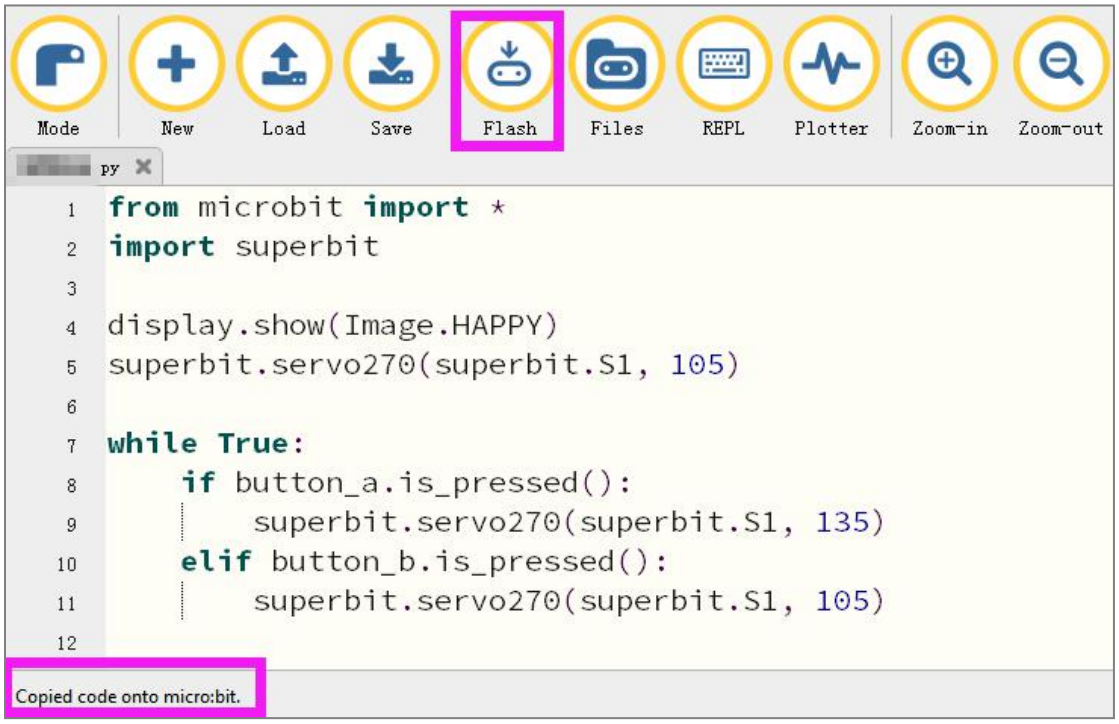


3. Click the ‘REPL’ button to check whether the Superbit library has been downloaded.

If not, please refer to [Preparation before class] --> [2.4 Python Programming Guide] .



4. After the program is written, use a micro USB cable to connect the computer and the micro:bit board. Please click the 'Flash' button to download the program to the micro:bit motherboard (You need to click the 'REPL' button again to close the function of importing library files before you download the program).



5. If the download failed, please confirm whether the micro:bit is connected to the computer through the micro USB data cable, and confirm whether the Super:bit

Python library has been imported.

4.Experimental phenomena

After the program is successfully downloaded, the micro:bit dot matrix will display a smiley pattern.

Game play 1: We can see that all RGB lights will switch a color every 1 second, red --> green --> blue --> white --> off, and keep cycling in this state.

Game play 2: We can see that the four RGB lights are become green in turn, time interval is 200ms, and keep cycling in this state.

Game play 3: We can see that the four RGB lights will turn on different colors in turn, time interval is 200ms, and keep cycling in this state.

Game play 4: We can see that all the RGB lights gradually go from off to on, then from on to off, and keep cycling in this state.

If you need to restart, press the reset button on the back of the micro:bit board.