7. Dice game

1. Learning goals

In this lesson, we will learn to use micro:bit to realize dice game.

2.Code and analysis

```python
from microbit import *
import random

one = Image("00000:"
            "00000:"
            "00900:"
            "00000:"
            "00000")

two = Image("00000:"
            "00090:"
            "00000:"
            "09000:"
            "00000")

three = Image("00000:"
              "09000:"
              "00900:"
              "00090:"
              "00000")

four = Image("00000:"
             "09090:"
             "00000:"
             "09090:"
             "00000")
```

```
28  five = Image("00000:"
29              "09090:"
30              "00900:"
31              "09090:"
32              "00000")
33
34  six = Image("00000:"
35              "09990:"
36              "00000:"
37              "09990:"
38              "00000")
39
40  while True:
41      x, y, z = accelerometer.get_values()
42      if x+y+z > 900:
43          stochastic = random.randint(0, 5)
44          if stochastic == 0:
45              display.show(one)
46          elif stochastic == 1:
47              display.show(two)
48          elif stochastic == 2:
49              display.show(three)
50          elif stochastic == 3:
51              display.show(four)
52          elif stochastic == 4:
53              display.show(five)
54          elif stochastic == 5:
55              display.show(six)
56
```

from microbit import *
This code is to import everything from the microbit library, and any program need to uses import this library.

import random
Import random library, we need to use randint() in the library to get random integer.

acceleromete.get_values()

Simultaneously measure the acceleration of all axes to generate a ternary ordered (X, Y, Z) integer group.

The micro:bit possess a dot matrix of 5*5 LEDs, and the brightness of each LED on the dot matrix can be set to a value from 0 to 9.
If the brightness of an LED is set to 0, then it goes out.
If its brightness is set to 9, then it is at the brightest level.
Using this feature, we can display custom images on the micro:bit lattice.

Test whether the microbit is shaking by detecting the acceleration values of the three axes of X, Y and Z, and use random numbers to display random number patterns.
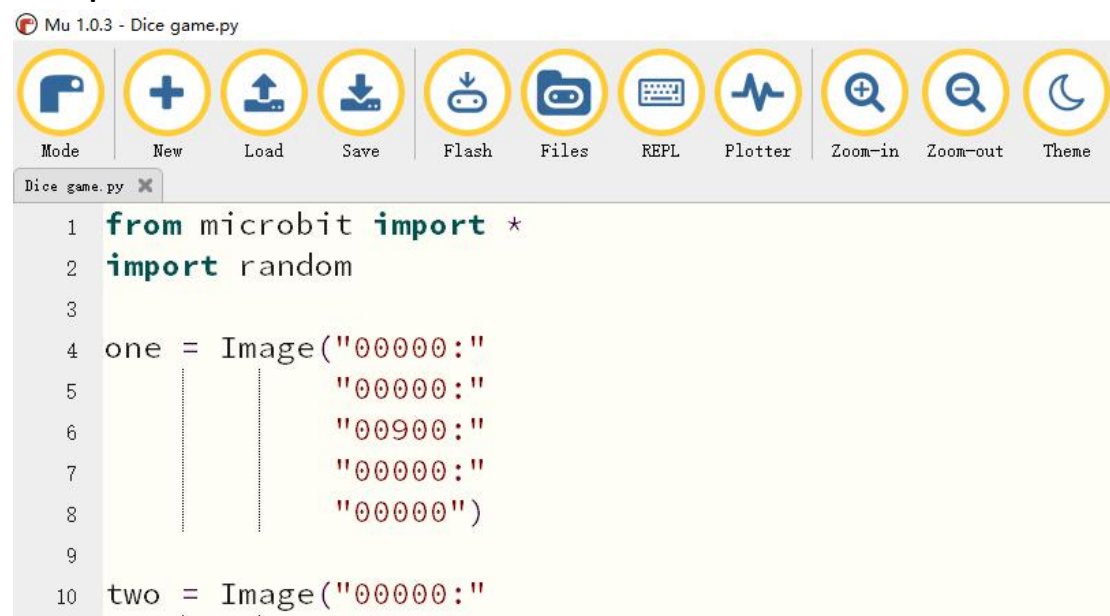
Note:
1 - The capital letter/lowercase letters must be distinguished！
2 - Correct spelling!
3 - Keywords such as # need a space between the content.
4 - You can only use the Tab key (tabulation key) for indentation.
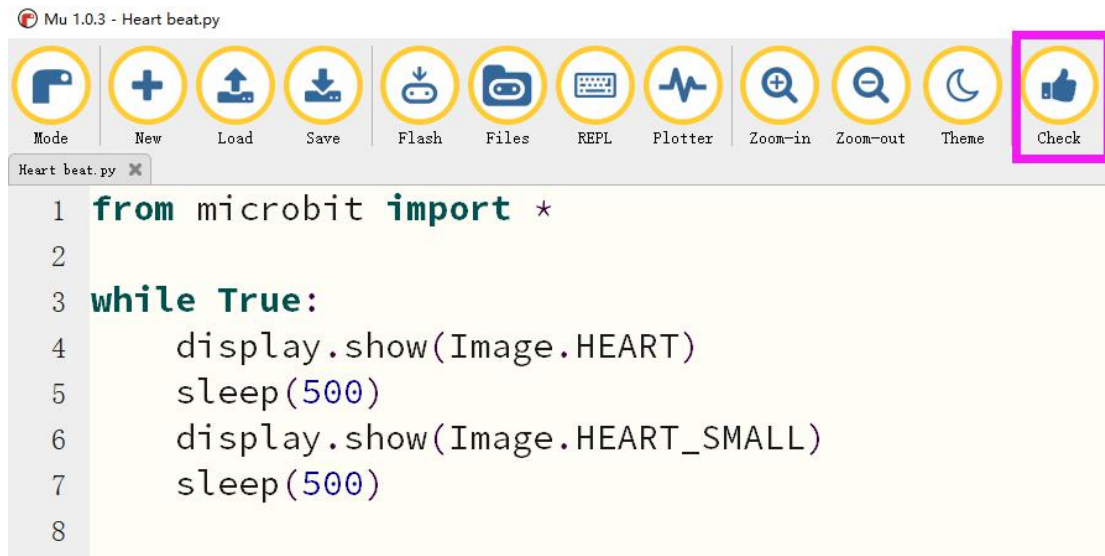
3. Programming and downloading

3.1 You should open the Mu software, and enter the code in the edit window, , as shown in Figure .
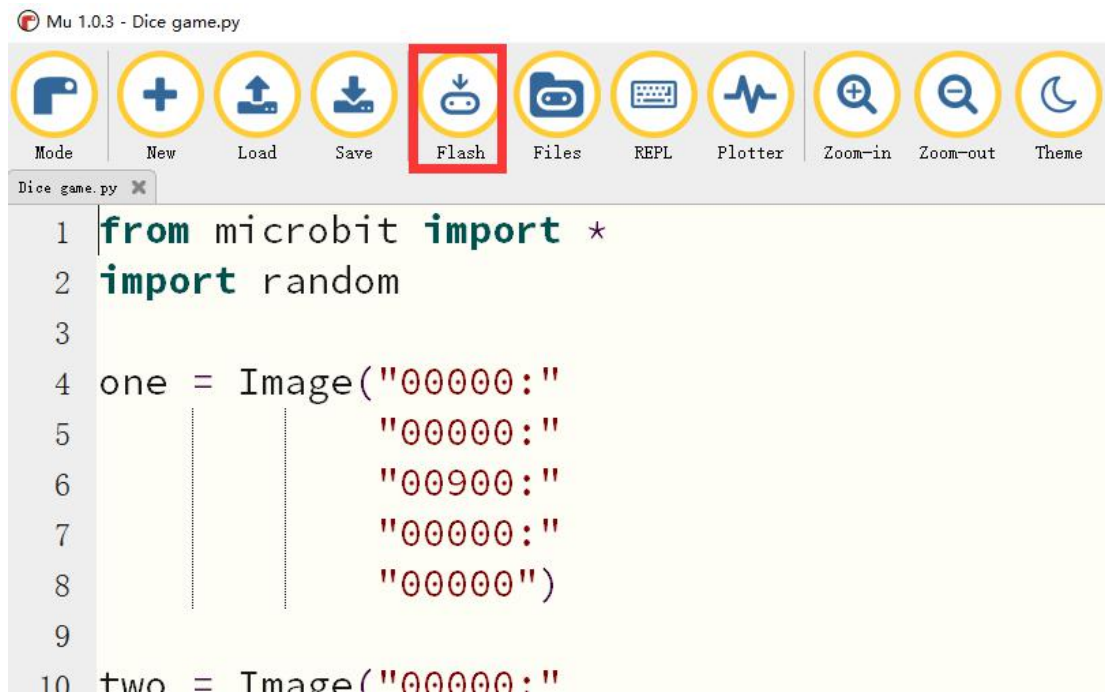**Note! All English and symbols should be entered in English, and the last line must be a space.**


```python
from microbit import *
import random

one = Image("00000:"
            "00000:"
            "00900:"
            "00000:"
            "00000")

two = Image("00000:"
```

3.2 As shown in Figure, you need to click the Check button to check if our code has an error. If a line appears with a cursor or an underscore, the program indicating this line is wrong.

```
Mu 1.0.3 - Heart beat.py

Heart beat.py ✕

1  from microbit import *
2
3  while True:
4      display.show(Image.HEART)
5      sleep(500)
6      display.show(Image.HEART_SMALL)
7      sleep(500)
8
```

3.3 You need to connect the micro data cable to micro:bit and the computer, then click the Flash button to download the program to micro:bit as shown in Figure.


```
Mu 1.0.3 - Dice game.py

Dice game.py ✕

1  from microbit import *
2  import random
3
4  one = Image("00000:"
5               "00000:"
6               "00900:"
7               "00000:"
8               "00000")
9
10 two = Image("00000:"
```

4. Experimental phenomena

After the program is successfully downloaded, when you shake your micro:bit, the program selects a random number between 1 and 6 and shows it on the LED display.