

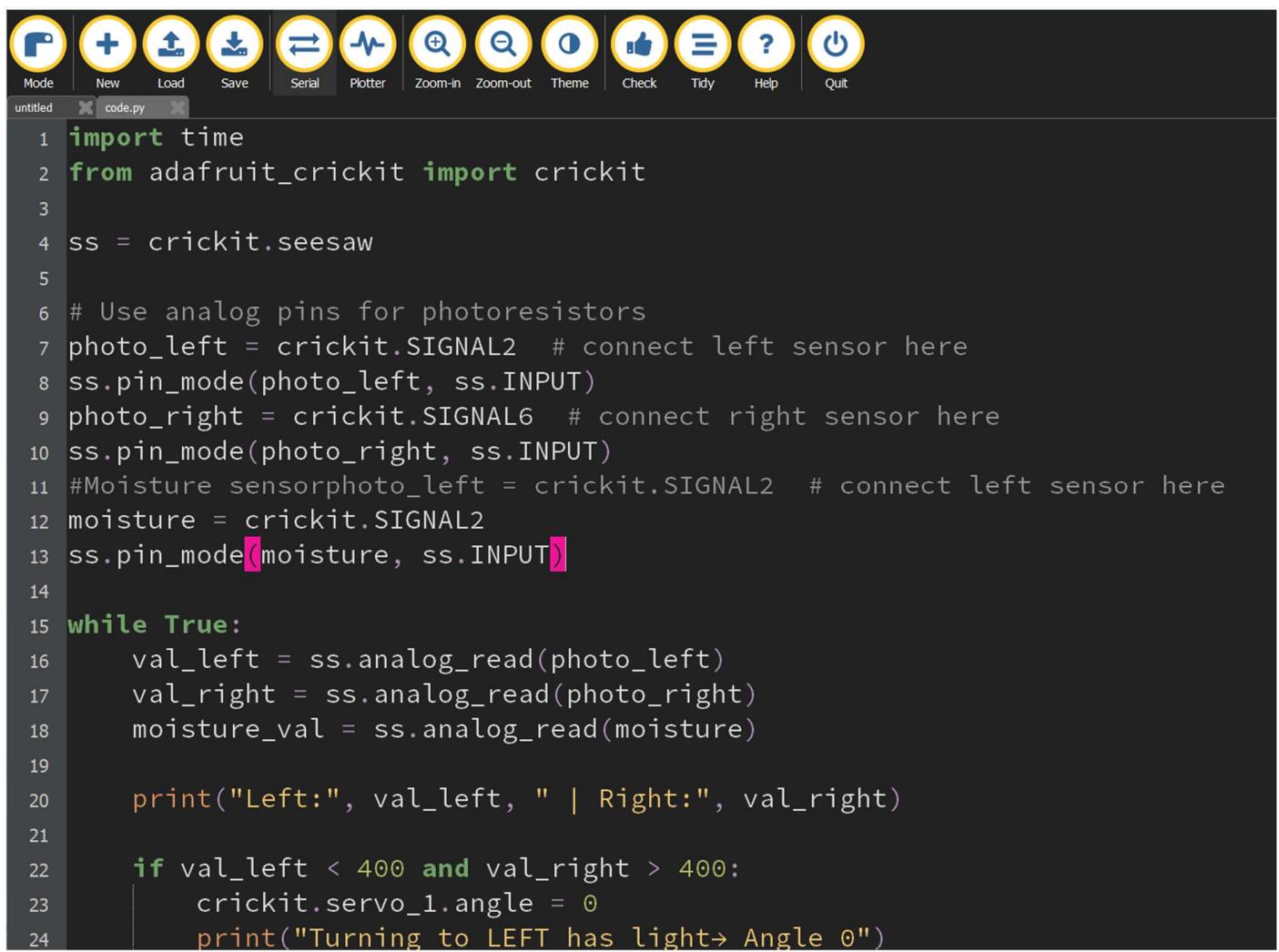
## Coding part – Using servo and light sensor to optimise light conditions for a plant

### Sensor Explanation

The servo sensor is an addition to the previous lesson. The servo is used for turning towards the sunlight. If both sides have light, the servo should be set to 90 degrees. If the left side has light, it should turn left, and if the right side has light, it should turn right.

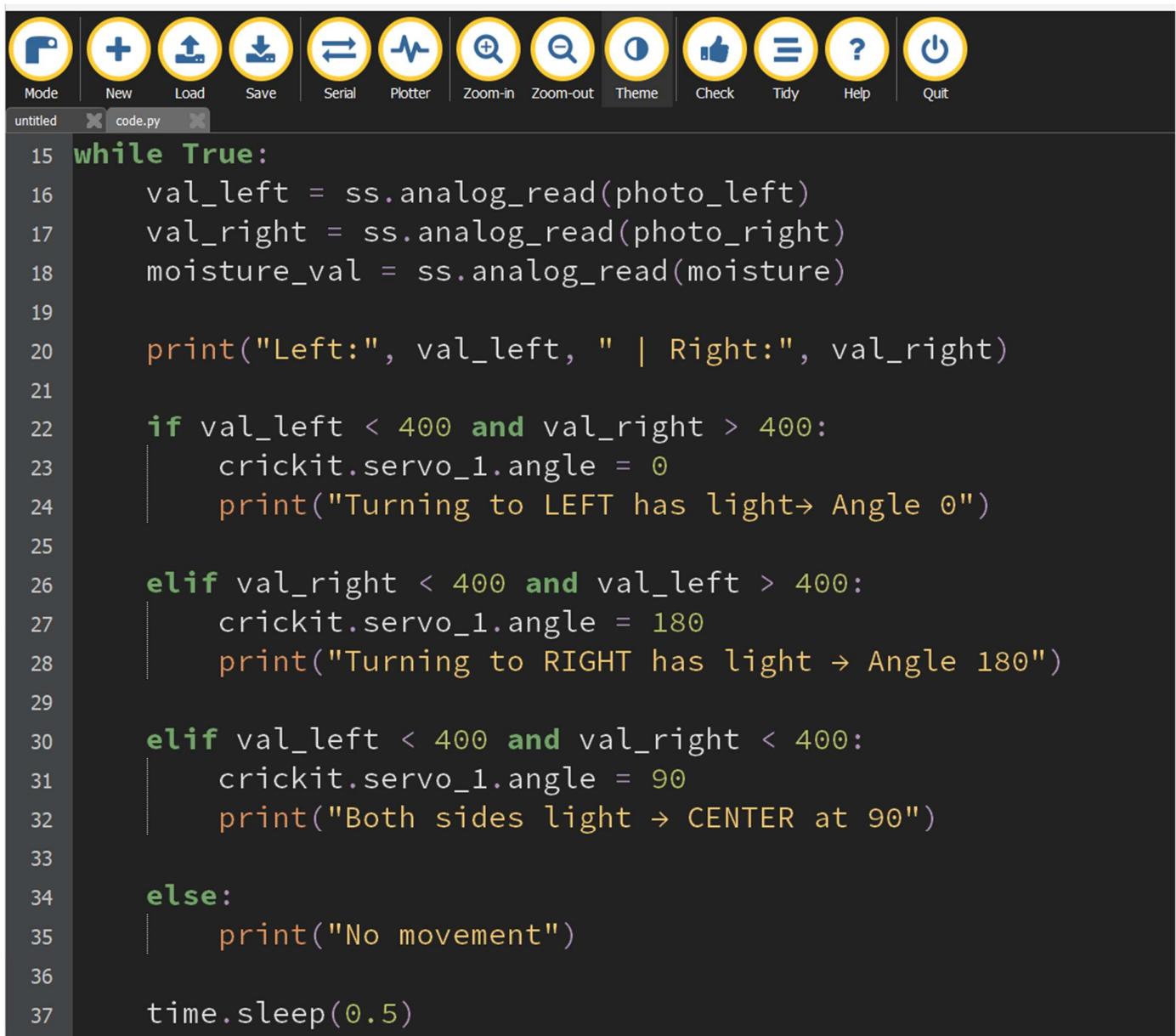
### Coding part

Here is the code you can see if you face any doubt while doing this task.



The image shows a screenshot of the micro:bit code editor. The interface includes a toolbar with icons for Mode, New, Load, Save, Serial, Plotter, Zoom-in, Zoom-out, Theme, Check, Tidy, Help, and Quit. Below the toolbar, there are two tabs: 'untitled' and 'code.py'. The code editor displays the following Python code:

```
1 import time
2 from adafruit_crickit import crickit
3
4 ss = crickit.seesaw
5
6 # Use analog pins for photoresistors
7 photo_left = crickit.SIGNAL2 # connect left sensor here
8 ss.pin_mode(photo_left, ss.INPUT)
9 photo_right = crickit.SIGNAL6 # connect right sensor here
10 ss.pin_mode(photo_right, ss.INPUT)
11 #Moisture sensor
12 moisture = crickit.SIGNAL2
13 ss.pin_mode(moisture, ss.INPUT)
14
15 while True:
16     val_left = ss.analog_read(photo_left)
17     val_right = ss.analog_read(photo_right)
18     moisture_val = ss.analog_read(moisture)
19
20     print("Left:", val_left, " | Right:", val_right)
21
22     if val_left < 400 and val_right > 400:
23         crickit.servo_1.angle = 0
24         print("Turning to LEFT has light> Angle 0")
```



The image shows a screenshot of a MicroPython IDE. The top bar features a toolbar with icons for Mode, New, Load, Save, Serial, Plotter, Zoom-in, Zoom-out, Theme, Check, Tidy, Help, and Quit. Below the toolbar, the title bar displays 'untitled' and 'code.py'. The main area is a code editor with the following Python script:

```
15 while True:
16     val_left = ss.analog_read(photo_left)
17     val_right = ss.analog_read(photo_right)
18     moisture_val = ss.analog_read(moisture)
19
20     print("Left:", val_left, " | Right:", val_right)
21
22     if val_left < 400 and val_right > 400:
23         crickit.servo_1.angle = 0
24         print("Turning to LEFT has light→ Angle 0")
25
26     elif val_right < 400 and val_left > 400:
27         crickit.servo_1.angle = 180
28         print("Turning to RIGHT has light → Angle 180")
29
30     elif val_left < 400 and val_right < 400:
31         crickit.servo_1.angle = 90
32         print("Both sides light → CENTER at 90")
33
34 else:
35     print("No movement")
36
37 time.sleep(0.5)
```

You can adjust the value of checking condition and modify and see the output.