

## Lab 4

### Introduction to CircuitPython

#### Part 1 - the CP REPL

With your Feather attached to the computer and Mu in CP mode, click the “Serial” button on the top menu panel. This should open a new panel in Mu. Hit any key to enter the CP REPL.

1. Try a print command.
2. Try some basic math.
3. Type `help("modules")`

This shows you a list of all the available modules you can import.

As you did with Python, encourage the students to play in the CP REPL a bit. They should see that CP is “Python”. CP happens to be running on a much smaller, limited electronic device.

Now let's try to use the REPL to actually “control” the Feather with CP.

1. Type `import board` into the REPL
2. Type `dir(board)` - this shows a list of all the predefined board pins.
3. Type `import digitalio`
4. Type `led = digitalio.DigitalInOut(board.RED_LED)`  
What have you just done? Check step 5 for help?

Created an instance of `digitalio.DigitalInOut` class. Created a `digitalio` object called `led`.

5. Type `help(led)`
6. One of the led's attributes is a direction.  
Type `led.direction = digitalio.Direction.OUTPUT`
7. Another led attribute is the value. Type `led.value = True`  
Make sure to look at the Feather Sense board as you hit enter!  
The red led on the Feather Sense turns on.
8. Type `led.value = False`  
What happens? Led turns off
9. Type `led.value = 1` and then `led.value = 0`  
What happens? Led turns on and off. 1 / 0 can act similar to boolean True / False

From the CP REPL, you can access and test individual hardware accessories!

Now let's try to control something else.

1. Type `import neopixel` into the REPL
2. Now create a NeoPixel object called `pixel`.  
Type `pixel = neopixel.NeoPixel(board.NEOPIXEL, 1, brightness = .25)`
3. Type `pixel[0] = (0, 0, 255)`  
What happens? The neopixel turns blue
4. Type `pixel[0] = (255, 0, 0)`  
What happens? The neopixel turns red
5. Type `pixel[0] = (0, 0, 0)`

What happens? **The neopixel turns off**

6. Type `pixel[0] = (0,255,0)`

What happens? **The neopixel turns green**

7. Type `pixel[0] = (255,255,255)`

What happens? **The neopixel turns white**

What type of data structure is the `(255,0,255)` and can you guess what each of the numbers represents?

**This is a tuple representing RGB color values. The first number represents RED. The second number represents GREEN and the third number represents BLUE.**

**`(0,0,0)` = absence of color turns the neopixel off**

**`(255,255,255)` = all color turns the neopixel white.**

**Each color value is represented by an 8 bit number (0-255) so we can mix them to make any color.**

8. Type `pixel[0] = (255,0,255)`

What happens? **The neopixel turns purple**

9. Type `pixel[0] = (255,255,0)`

What happens? **The neopixel turns yellow**

10. Type `pixel.brightness = .1`

What happens? **The neopixel gets dimmer**

11. Type `pixel.brightness = .65`

What happens? **The neopixel gets brighter**

12. Type `pixel.brightness = .25`

**In the CP REPL write a for loop that will blink the led and pixel 50 times.** You should `import time` and use `time.sleep(.5)` to create delays in your loop. **Show your code working on the Feather to your instructor.**

☐

```
import time
for i in range(100):
    led.value = 1
    time.sleep(.5)
    led.value = 0
    time.sleep(.5)
```

## Part 2 - Create and run a CP program to blink the led

In most cases, we will want our Feather Sense to start running a program as soon as it powers up. CP knows to look for specific file names on the CP device, in our case the Feather, that it will try to run. CP looks for **code.txt**, **code.py**, **main.txt** and **main.py**, in that specific order. We will use **code.py** as the name for our code we want to run on the CP board.

1. Click the New button to create a new file.
2. Double click on "untitled" in the editor tab and save the file in your mu\_code folder on the computer with the filename **"lastname\_mu4\_part2.py"**

3. Copy the code listed below into the Mu editor and save your code to the computer.

```
#
import board
import digitalio
import time
#
led = digitalio.DigitalInOut(board.RED_LED)
#
led.direction = digitalio.Direction.OUTPUT
#
while True:
    print("Hello from CircuitPython land!")
#
    led.value = True
#
    time.sleep(0.5)
#
    led.value = False
#
    time.sleep(0.5)
```

4. Double click the filename on the editor tab and save the file to the CIRCUITPY device with the name **"code.py"**  
Does it work?
5. Change the first time.sleep argument from .5 to .1  
Change both .sleep arguments to 1  
What changes? **On/off delay times will change**
6. Refer to Part1 and add some code to the file to blink the neopixel as well.
7. **Add inline comments to your code**
8. Double click on **"code.py"** in the editor tab and save the file to your mu\_code folder on the computer AGAIN with the filename **"lastname\_lab4\_part2.py"**
9. **Show your code working on the Feather to your instructor.**



Possible solution in file lab4\_part2.py