

...cont'd

- 8 Turn on your Raspberry Pi then begin a new Python script by making attributes and methods available
`import RPi.GPIO as GPIO`
`from time import sleep`
- 9 Add statements to use board pin numbers in the script, and set up pins 7,11 and 13 to supply power as output
`GPIO.setmode(GPIO.BOARD) ; GPIO.setup(7,GPIO.OUT)`
`GPIO.setup(11,GPIO.OUT) ; GPIO.setup(13,GPIO.OUT)`
- 10 Now, add a loop to sequentially light each LED for one second on each of three iterations, then reset the channels
`i = 1`
`while i < 4 :`
 `print('Cycle: ' + str(i))`
 `GPIO.output(7 , True) ;`
 `print('\t7 Output True - RED ON') ; sleep(1)`
 `GPIO.output(7 , False)`
 `GPIO.output(11 , True) ;`
 `print('\t11 Output True - YELLOW ON') ; sleep(1)`
 `GPIO.output(11 , False)`
 `GPIO.output(13 , True) ;`
 `print('\t13 Output True - GREEN ON') ; sleep(1)`
 `GPIO.output(13 , False)`
 `i += 1`
`GPIO.cleanup()`
- 11 Save the file, then enter this command to run the script with superuser privileges and see the LED sequence
`sudo python gpio_sequence.py`



If you miss out the loop incrementer `i+=1` it will continue to run – press **Ctrl + C** to exit the script.

```
pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi:~$ sudo python gpio_sequence.py
Cycle: 1
    7 Output True - RED ON
    11 Output True - YELLOW ON
    13 Output True - GREEN ON
Cycle: 2
    7 Output True - RED ON
    11 Output True - YELLOW ON
    13 Output True - GREEN ON
Cycle: 3
    7 Output True - RED ON
    11 Output True - YELLOW ON
    13 Output True - GREEN ON
pi@raspberrypi:~$ █
```