PIBRELLA COMMANDS

Run Python in sudo: `sudo idle`

TURNING LIGHTS ON / OFF:

- Turn **all** lights **on**:   `pibrella.light.on()`
- Turn **all** lights **off**:   `pibrella.light.off()`
- Turn **red** light **on**:   `pibrella.light.red.on()`
- Turn **red** light **off**:   `pibrella.light.red.off()`
  *#Lights can be referred to by **colour** or as the group name **light***
  *#On and Off can be replaced with **High/Low***

FLASHING LIGHTS:

- **Blink** the **Green** light: `pibrella.light.green.blink(ON_TIME, OFF_TIME)`
- **Blink** 2sec. on 1sec off: `pibrella.light.green.blink(2, 1)`

PULSE LIGHTS:

- **Pulse** red light:
  `pibrella.light.red.pulse(FADE_IN_TIME, FADE_OUT_TIME, ON_TIME, OFF_TIME)`
- **Pulse** in for 2secs, on for 2secs, fade out 4secs, off for 1sec:
  `pibrella.light.red.pulse(2,4,2,1)`

FADE LIGHTS:

- **Fade** Amber light: `pibrella.light.red.fade(0, 100, 5)`
  #Fade from 0 to 100% in 5 seconds

BUZZER:

- Buzzer Fail sound: `pibrella.buzzer.fail()`
- Buzzer Win sound: `pibrella.buzzer.success()`
- Read state of Buzzer: `pibrella.button.read()`
- Read state of Buzzer: `pibrella.button.read()`
- Play a note: `pibrella.buzzer.buzz( frequency )`
- Play a note: `pibrella.buzzer.note(0 is A, 440Hz)`
- Turn buzzer off: `pibrella.buzzer.off()`

- Loop a function：   `pibrella.loop(name_of_function)`

INPUTS:

The **four inputs** are named **a, b, c, d**, when connect the **white LED** will light up
Inputs are read, 1 is high and 0 is low, these values are interchangeable.

- To read an input **b**:   `pibrella.input.b.read()`
- To Store the value of input **c**:
  `my_value = pibrella.input.c.read()`
- To check of input a is high: `if pib.input.a.read()==1:`
- Or to read all inputs into a dictionary:
  `inputs = pibrella.input.read()`
  `input_e = inputs['e']`
- The button works as an input and can also be checked:
  `pibrella.button.pressed()`
  `pibrella.button.changed()`

OUTPUTS:

The **four outputs** are named **e, f, g, h** and also have a **white LED**.
Outputs can be turned **on** or written to with a value of 1 / 0

- **Turn ON** output '**f**':
- `pibrella.output.f.on()` **or** `pibrella.output.f.write(1)`
- **Turn OFF** output '**f**':
- `pibrella.output.f.off()` **or** `pibrella.output.f.write(0)`
- **Turn ON ALL** outputs:
- `pibrella.output.on()` **or** `pibrella.output.write(1)`
- **Turn OFF ALL** outputs:
- `pibrella.output.off()` **or** `pibrella.output.write(0)`

A CHANGED STATE:

- Check to see if the **state** of the button has changed:
  ```
  def button_pressed(pin):
          print("You pressed the button!")

  pibrella.button.pressed(button_pressed)
  ```

- Turning a light on when the button is pressed:
  ```
  import pibrella
  def button_changed(pin):
          pibrella.light.red.write(pin.read())
  pibrella.button.changed(button_changed)
  ```