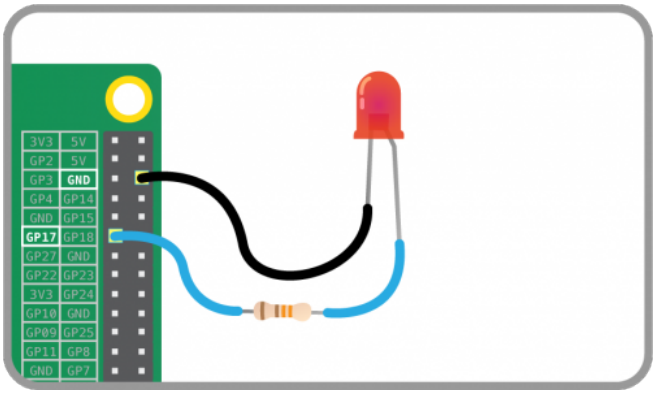
Control an LED with Python on the Raspberry Pi

Physical Computing uses a computer's power to control things in the outside world. The Raspberry Pi is a full-blown computer on a single chip. It has IO (Input/Output) pins which can power and control your devices programmatically. Powering an LED light bulb is a very basic example. In our case Pin 17 is attached to the light bulb, but it could just as easily be attached to a sensor, button, or camera.



The little wormy-looking beige thing connected to the blue wire is a capacitor. It limits the flow of electricity so the Pi doesn't burn out.

Python is a great language to get started with - it's fast and easy to use. We are also accessing the 'gpiozero' Python library to control the pins. Let's get started.

```
from gpiozero import LED

led = LED(17)
led.on()
led.off()
```

These first two lines import the methods we will need.

Tell the Pi that when we say 'led' we mean pin #17.

'Let there be light!'

What about a blinking light?

```
from gpiozero import LED
from time import sleep

led = LED(17)
while True:
     led.on()
     sleep(1)
     led.off()
     sleep(1)
```

This time we will also import the 'sleep' method from time.

Tell the Pi that when we say 'led' we mean pin #17.

The tabs are important.

CTRL+ c to stop.

(Adapted from www.raspberrypi.org)