

MIT AI2 204

IoT with MIT App Inventor

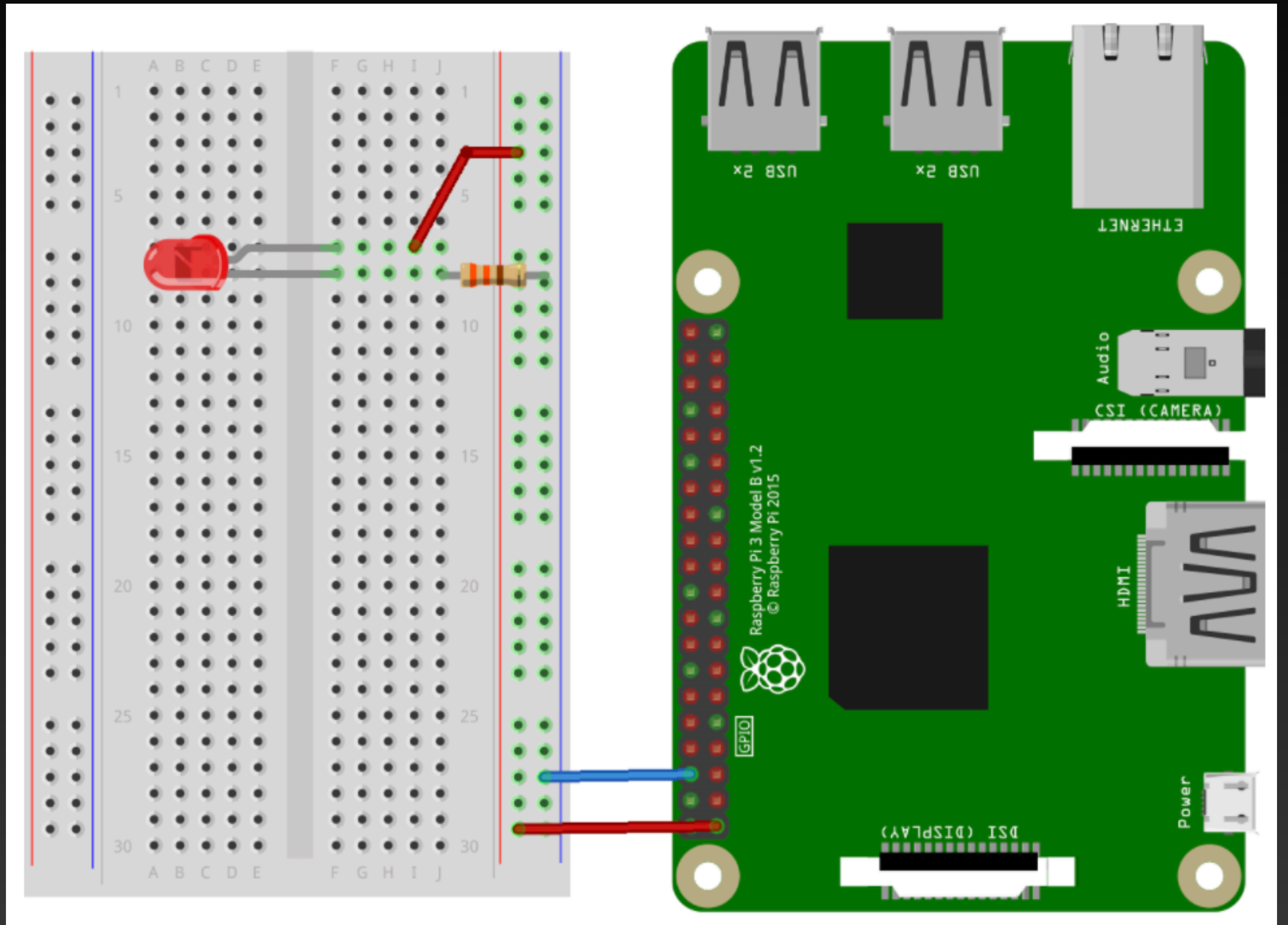
Fundamental

X. Tang

Test your circuit! An LED project

Switching an LED on and off

GPIO Zero is a new Python library which provides a simple interface to everyday GPIO components. It comes installed by default in Raspbian.

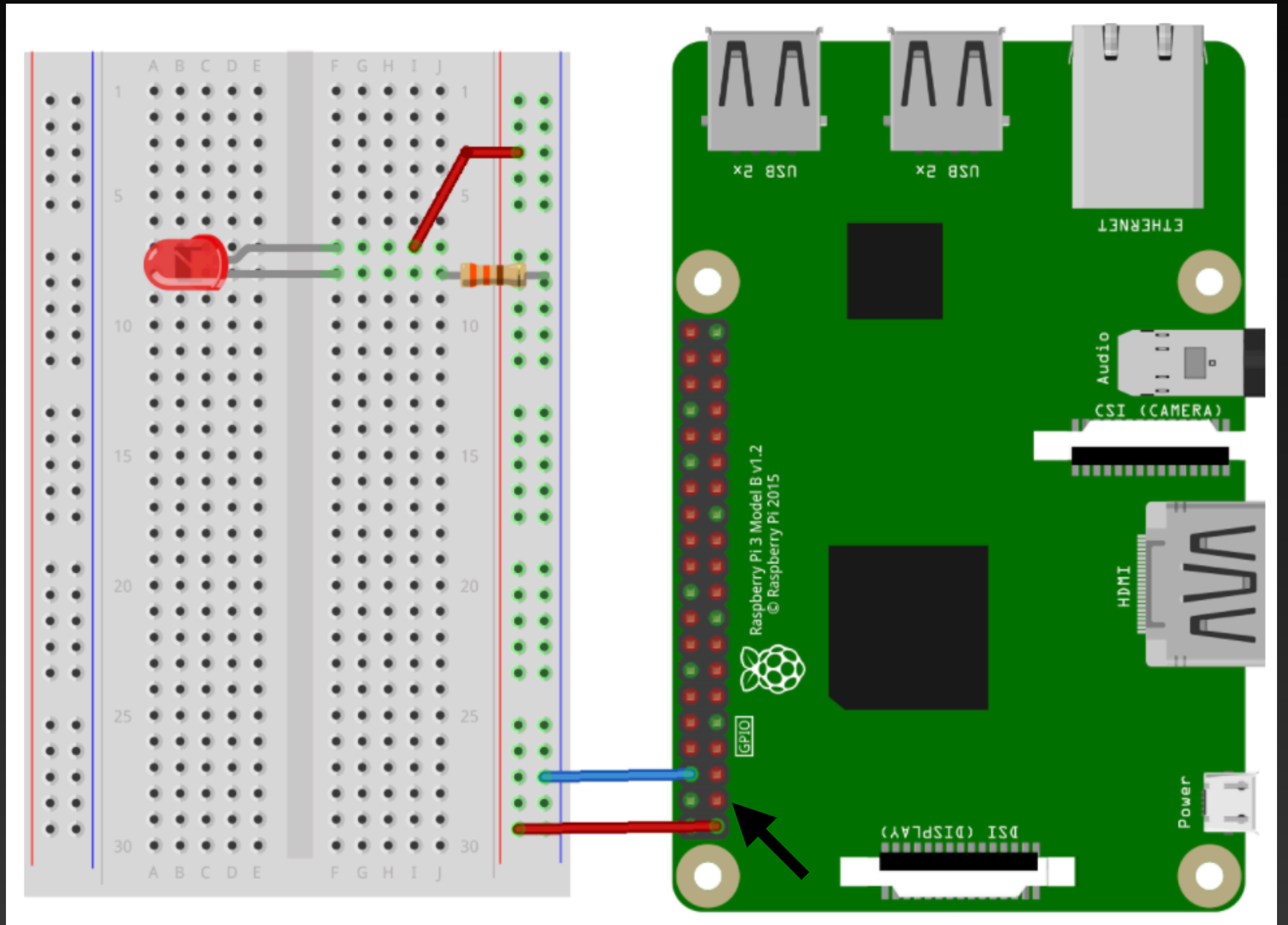


Test your circuit! An LED project

Switching an LED on and off

GPIO Zero is a new Python library which provides a simple interface to everyday GPIO components. It comes installed by default in Raspbian.

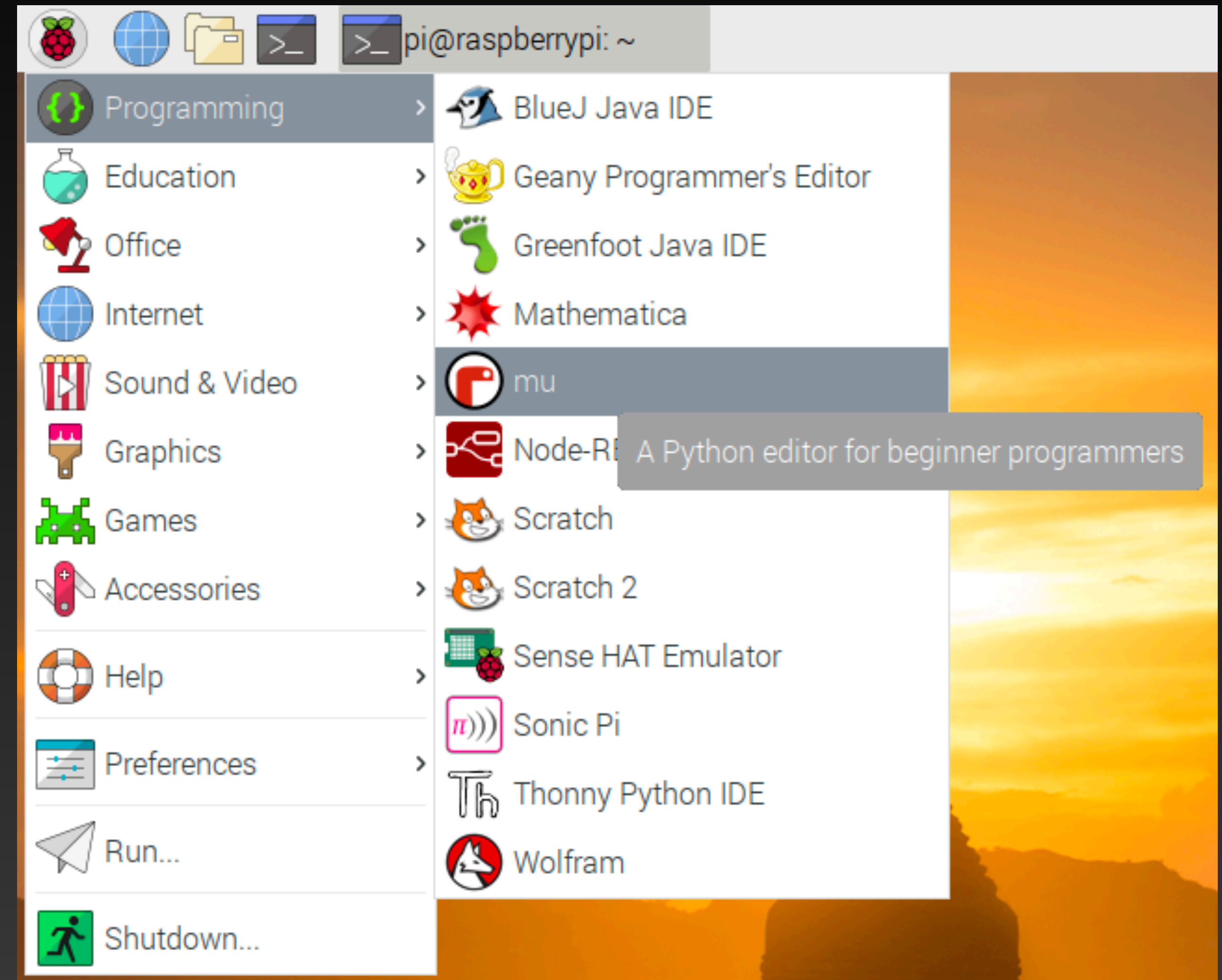
Switch your input to GPIO2



Test your circuit! GPIO Zero library

How to open Mu

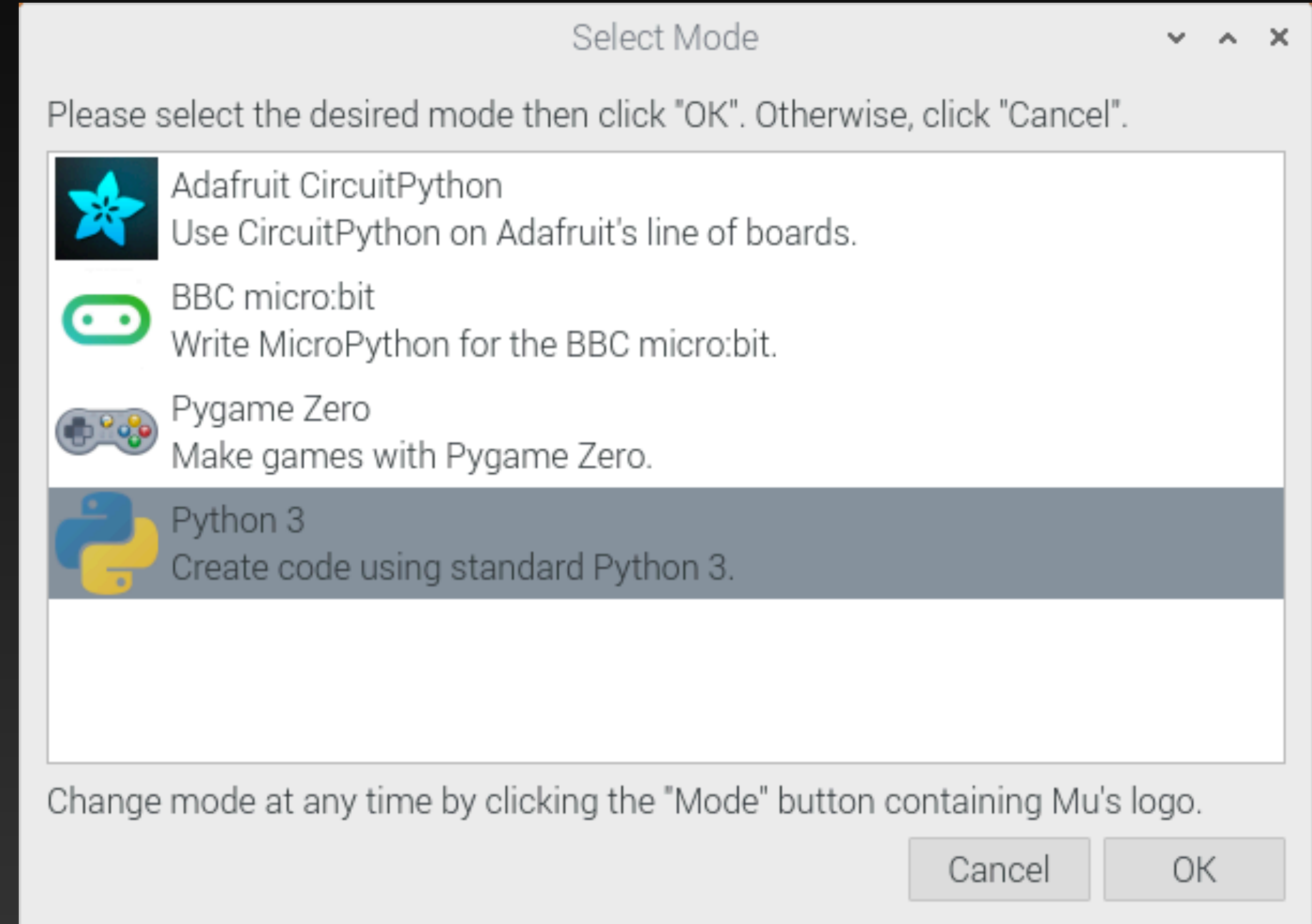
Go to the Programming menu and click on Mu.



Test your circuit! GPIO Zero library

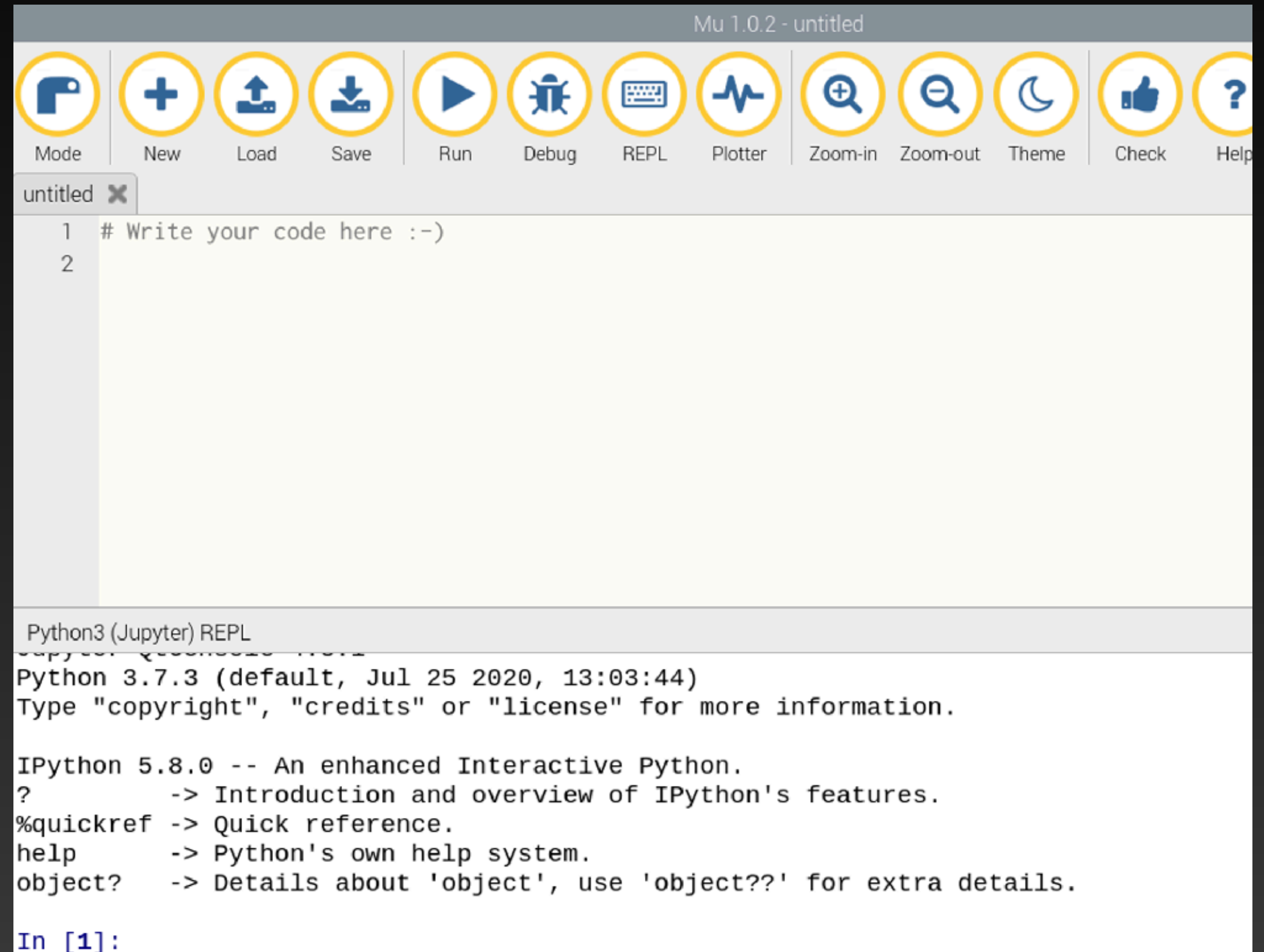
Then choose the mode in which you want to use Mu.

Choose Python 3 if you are creating a new Python script.



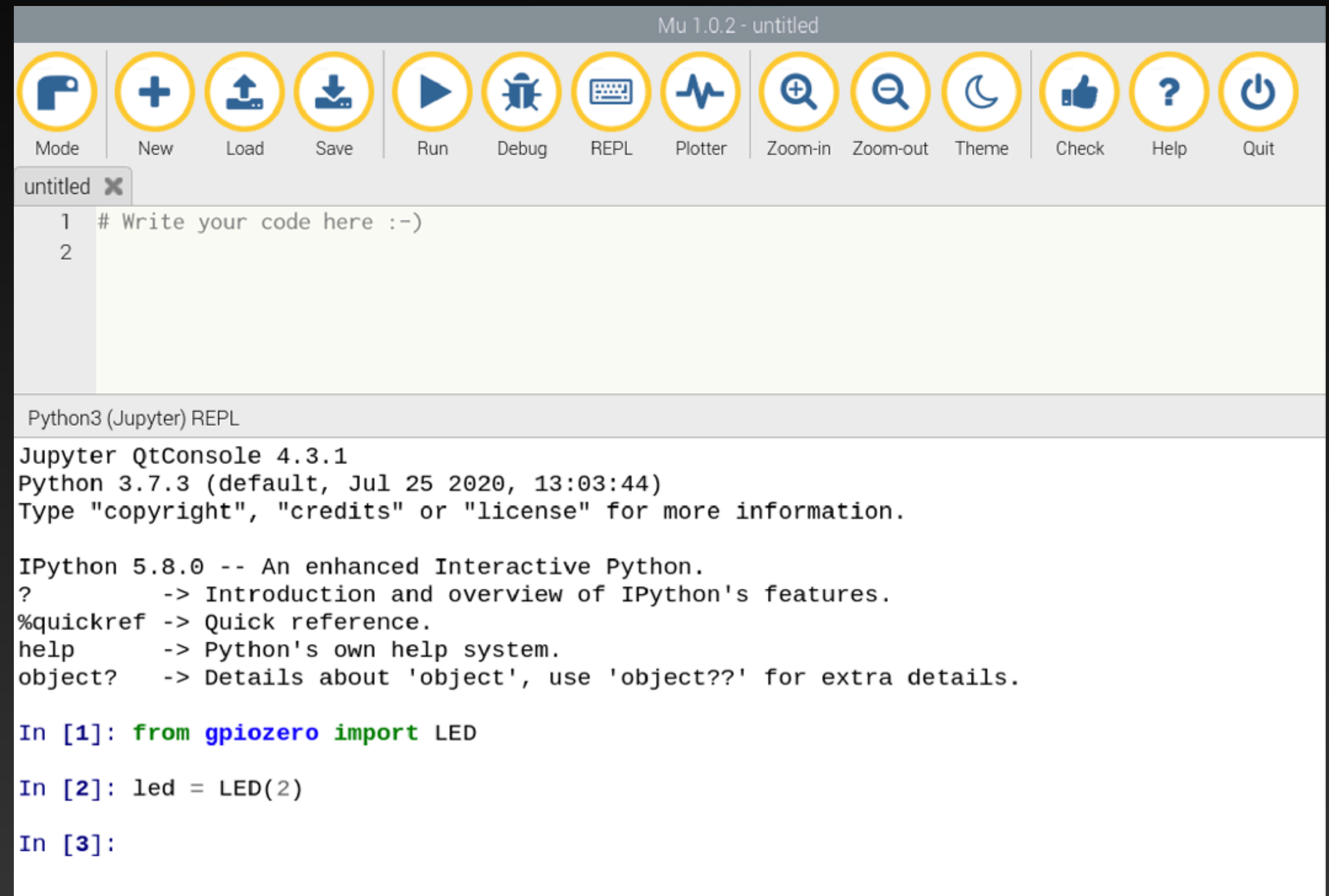
Test your circuit! GPIO Zero library

You can switch an LED on and off by typing commands directly into the REPL. Click on the REPL button in the menu bar.



Test your circuit! GPIO Zero library

First import the GPIO Zero library, and tell the Pi which GPIO pin you are using - in this case pin 2.



The screenshot shows the Mu Python IDE interface. At the top, the title bar reads "Mu 1.0.2 - untitled". Below it is a toolbar with icons for Mode, New, Load, Save, Run, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. The main window has a tab labeled "untitled" and a code editor with two lines of code:

```
1 # Write your code here :-)  
2
```

Below the code editor is a Python3 (Jupyter) REPL console. It displays the following text:

```
Jupyter QtConsole 4.3.1  
Python 3.7.3 (default, Jul 25 2020, 13:03:44)  
Type "copyright", "credits" or "license" for more information.  
  
IPython 5.8.0 -- An enhanced Interactive Python.  
?                -> Introduction and overview of IPython's features.  
%quickref        -> Quick reference.  
help             -> Python's own help system.  
object?         -> Details about 'object', use 'object??' for extra details.  
  
In [1]: from gpiozero import LED  
  
In [2]: led = LED(2)  
  
In [3]:
```

Test your circuit! GPIO Zero library

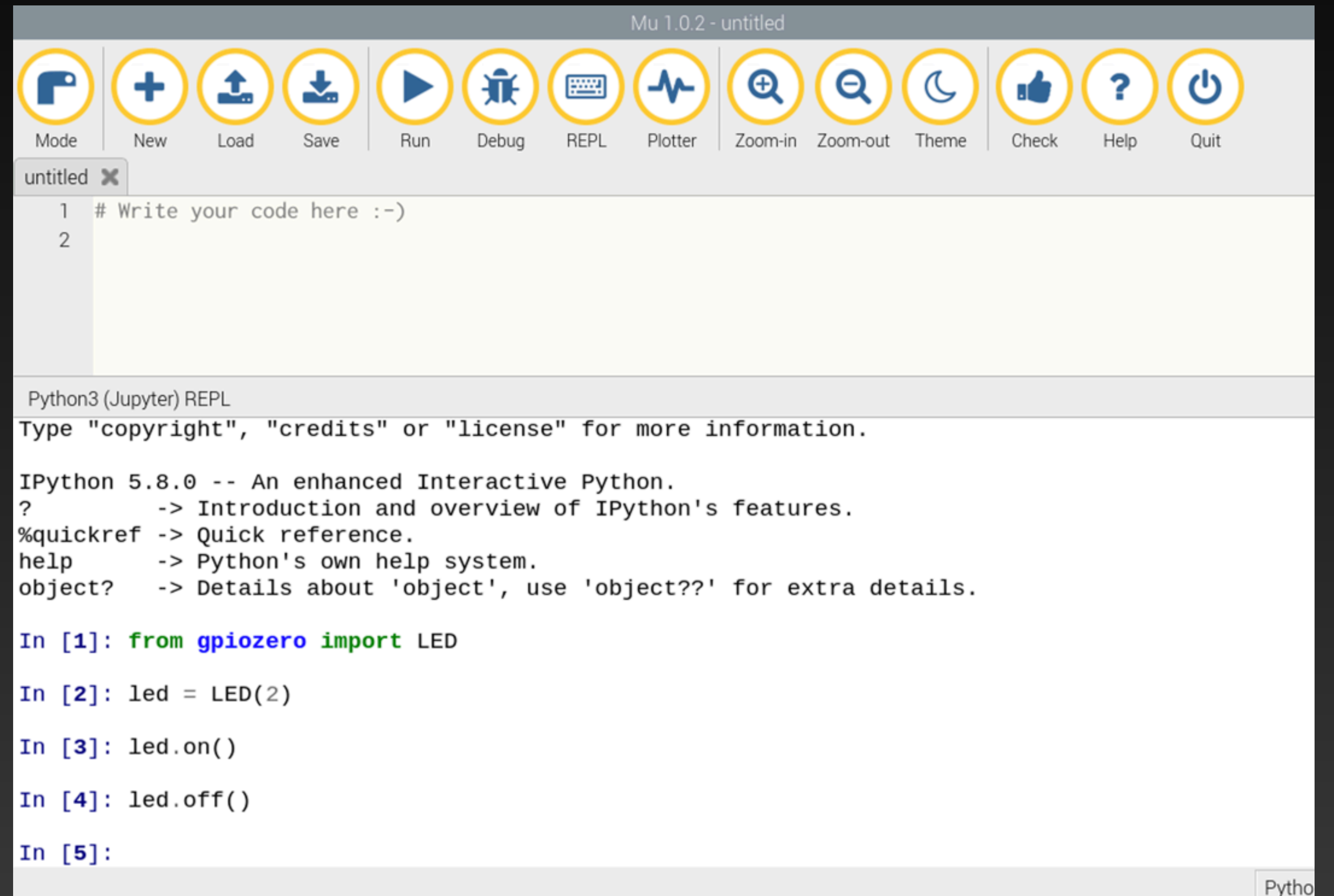
To make the LED switch on, type the following and press Enter:

```
led.on()
```

To make it switch off you can type:

```
led.off()
```

Your LED should switch on and then off again. But that's not all you can do.



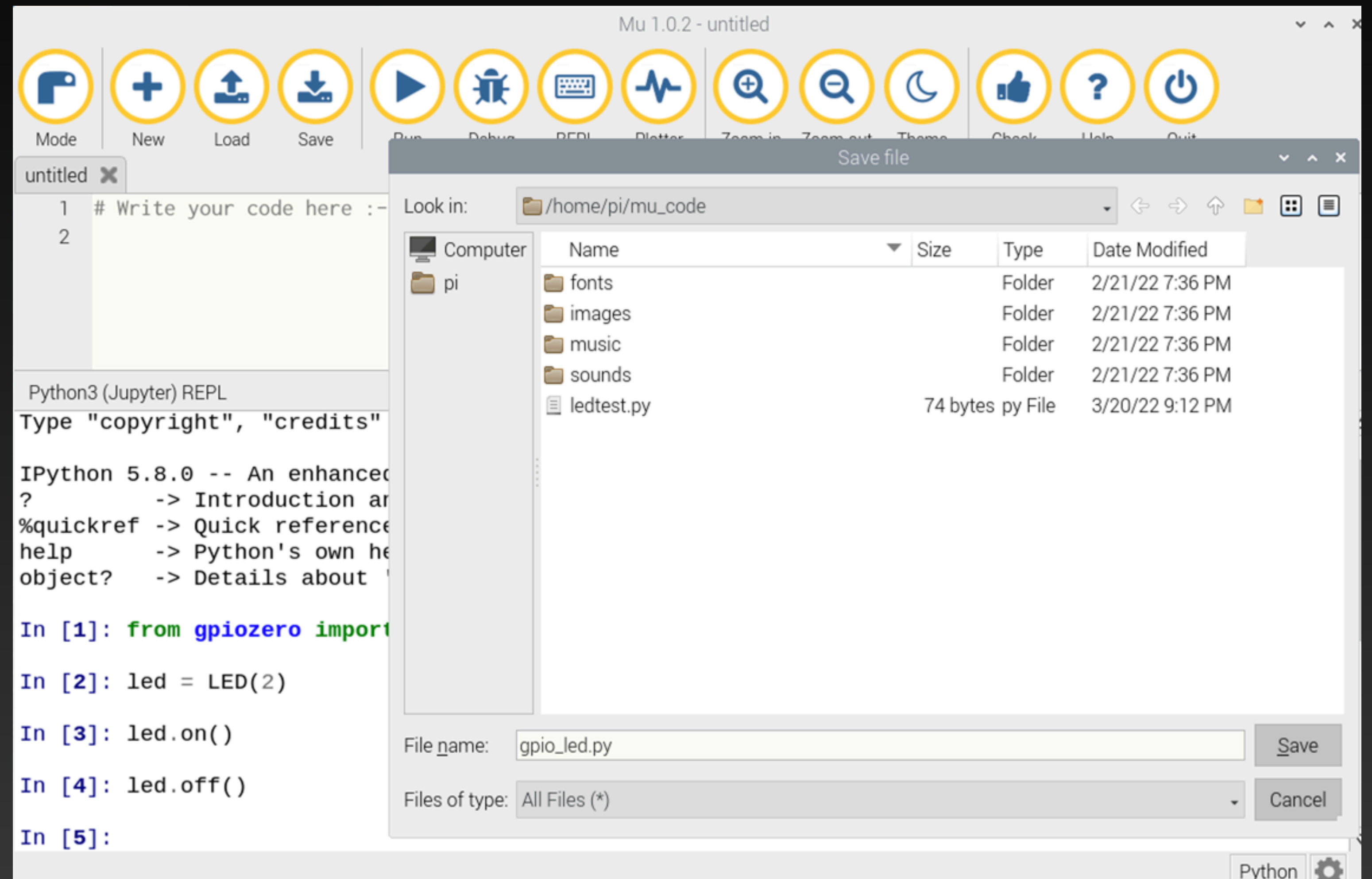
The screenshot shows the Mu Python IDE interface. The top bar displays 'Mu 1.0.2 - untitled'. Below it is a toolbar with icons for Mode, New, Load, Save, Run, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. The main editor area shows a file named 'untitled' with two lines of code: `1 # Write your code here :-)` and `2`. Below the editor is a Python3 (Jupyter) REPL window. It displays the following text: `Type "copyright", "credits" or "license" for more information.`, `IPython 5.8.0 -- An enhanced Interactive Python.`, `? -> Introduction and overview of IPython's features.`, `%quickref -> Quick reference.`, `help -> Python's own help system.`, and `object? -> Details about 'object', use 'object??' for extra details.`. The REPL also shows the execution of the following code: `In [1]: from gpiozero import LED`, `In [2]: led = LED(2)`, `In [3]: led.on()`, `In [4]: led.off()`, and `In [5]:`. The bottom right corner of the REPL window shows the text 'Pytho'.

Test your circuit! Flashing an LED

With the help of the `time` library and a little loop, you can make the LED flash.

Create a new file by clicking New.

Save the new file by clicking Save. Save the file as `gpio_led.py`.



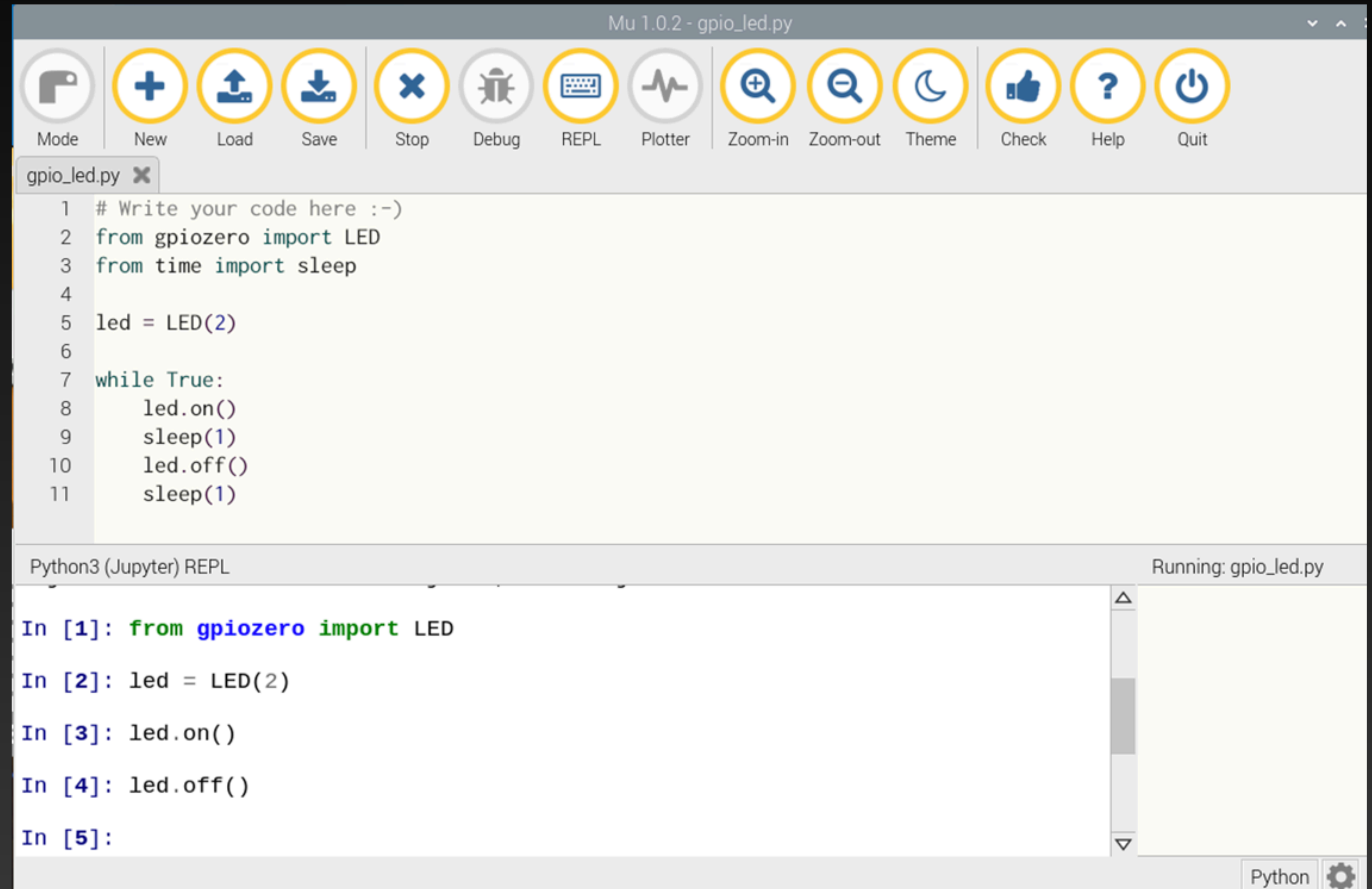
Test your circuit! Flashing an LED

Enter the following code to get started:

```
from gpiozero import LED
from time import sleep
```

```
led = LED(2)
```

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



The screenshot shows the Mu Python IDE interface. The title bar reads "Mu 1.0.2 - gpio_led.py". The top toolbar contains icons for Mode, New, Load, Save, Stop, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. The main editor area displays a Python script in a file named "gpio_led.py":

```
1 # Write your code here :-)
2 from gpiozero import LED
3 from time import sleep
4
5 led = LED(2)
6
7 while True:
8     led.on()
9     sleep(1)
10    led.off()
11    sleep(1)
```

Below the editor is the "Python3 (Jupyter) REPL" panel, which shows the execution of the code in five steps:

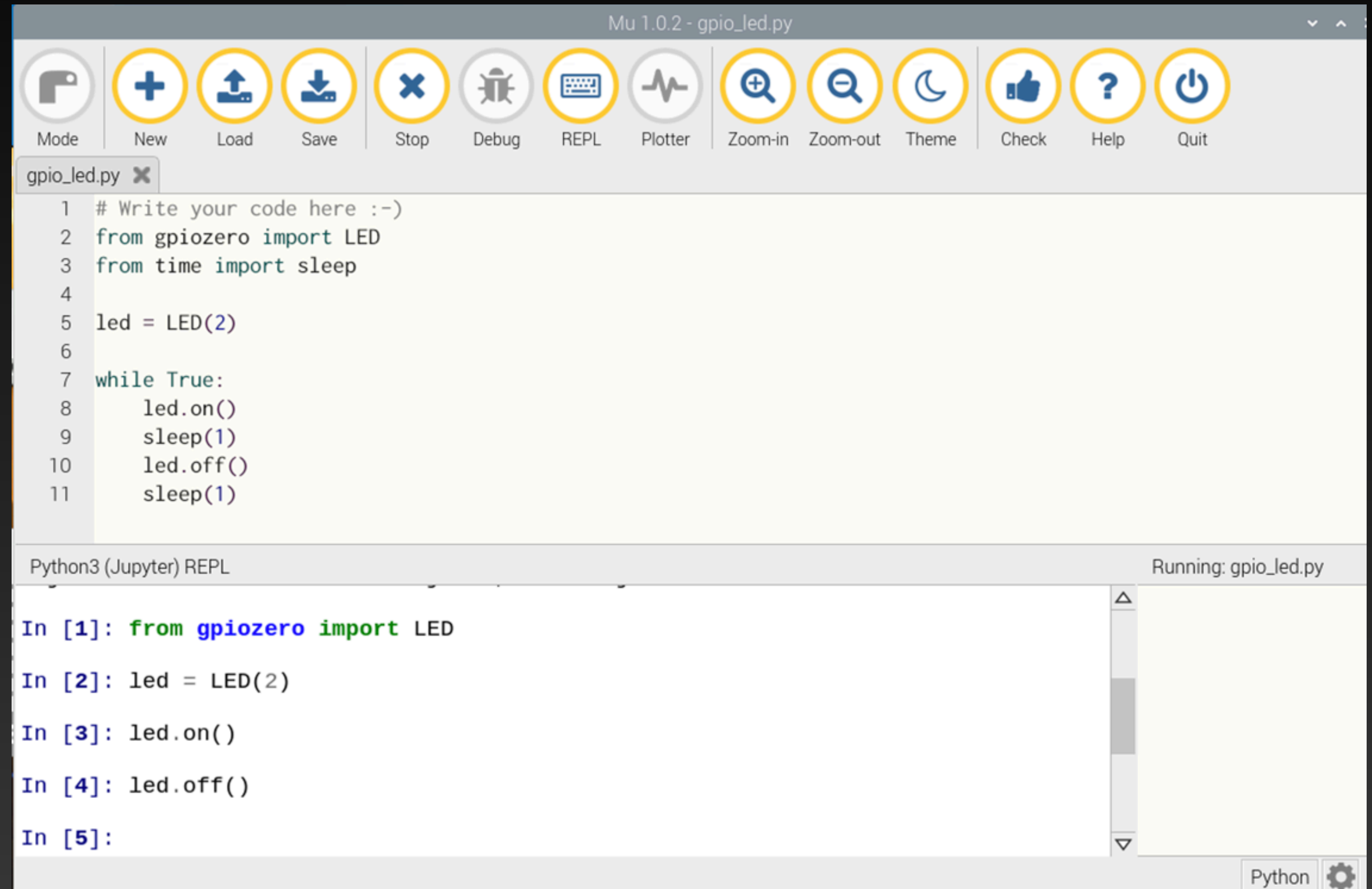
```
In [1]: from gpiozero import LED
In [2]: led = LED(2)
In [3]: led.on()
In [4]: led.off()
In [5]:
```

The status bar at the bottom right indicates "Running: gpio_led.py" and "Python".

Test your circuit! Flashing an LED

Save the file and run the code with by clicking on Run.

The LED should be flashing on and off. To exit the program click Stop.



The screenshot shows the Mu Python IDE interface. The title bar reads "Mu 1.0.2 - gpio_led.py". The top toolbar contains icons for Mode, New, Load, Save, Stop, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. The main editor displays a Python script named `gpio_led.py` with the following code:

```
1 # Write your code here :-)  
2 from gpiozero import LED  
3 from time import sleep  
4  
5 led = LED(2)  
6  
7 while True:  
8     led.on()  
9     sleep(1)  
10    led.off()  
11    sleep(1)
```

Below the editor is the Python3 (Jupyter) REPL. It shows the execution of the code in five steps:

```
In [1]: from gpiozero import LED  
In [2]: led = LED(2)  
In [3]: led.on()  
In [4]: led.off()  
In [5]:
```

The status bar at the bottom right indicates "Running: gpio_led.py" and "Python".

Python Basics

- Comments, literal constants, numbers, quotes.
- Operators and Expressions
- Control Flows
- Functions
- Modules

<https://automatetheboringstuff.com/2e/chapter1/>

<https://automatetheboringstuff.com/2e/chapter2/>