

MIT AI2 204

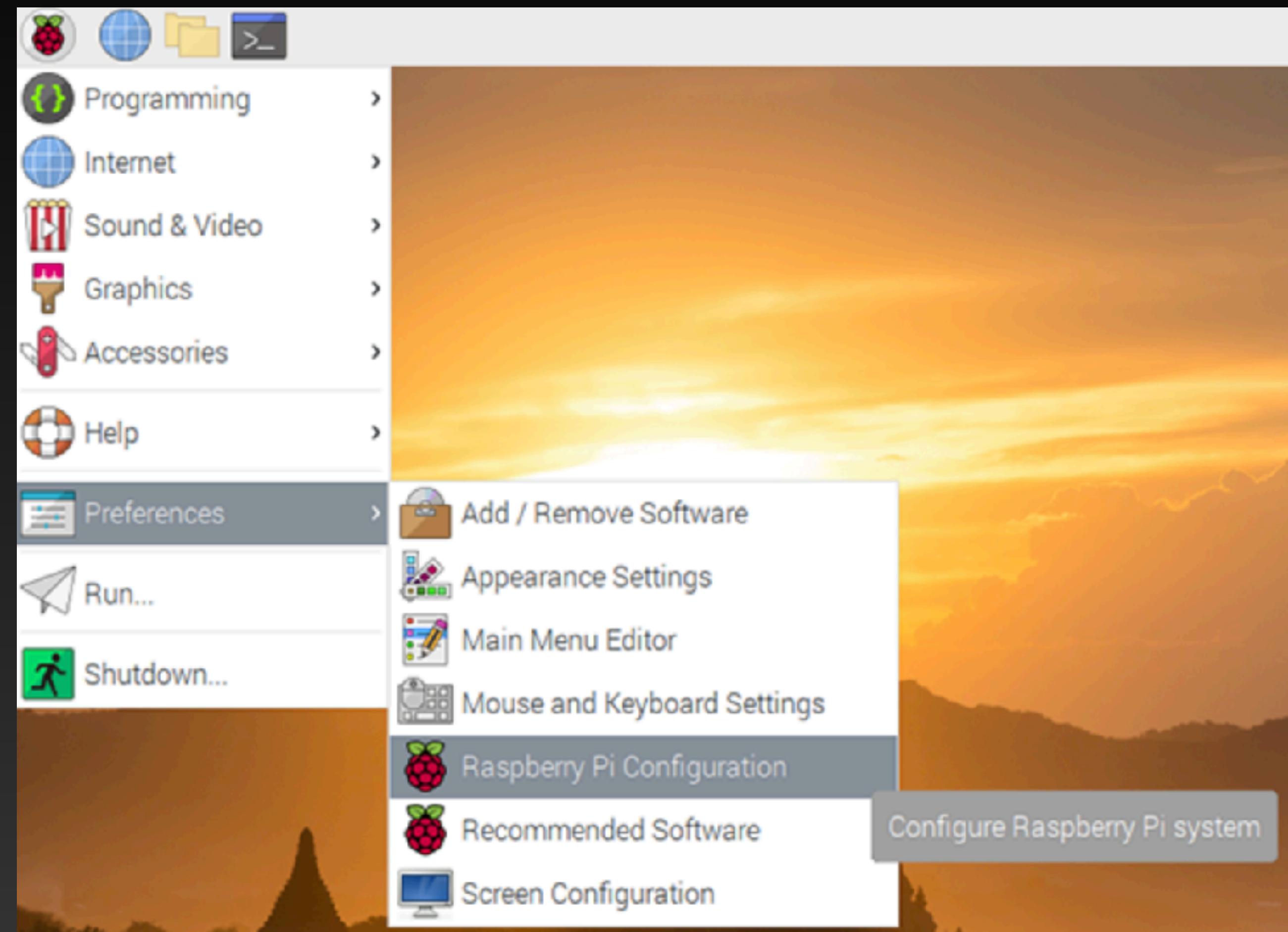
IoT with MIT App Inventor

Fundamental

Xincheng Tang

Configuring your Raspberry Pi

You can control most of your Raspberry Pi's settings, such as the password, through the Raspberry Pi Configuration application found in Preferences on the menu.



Configuring your Raspberry Pi

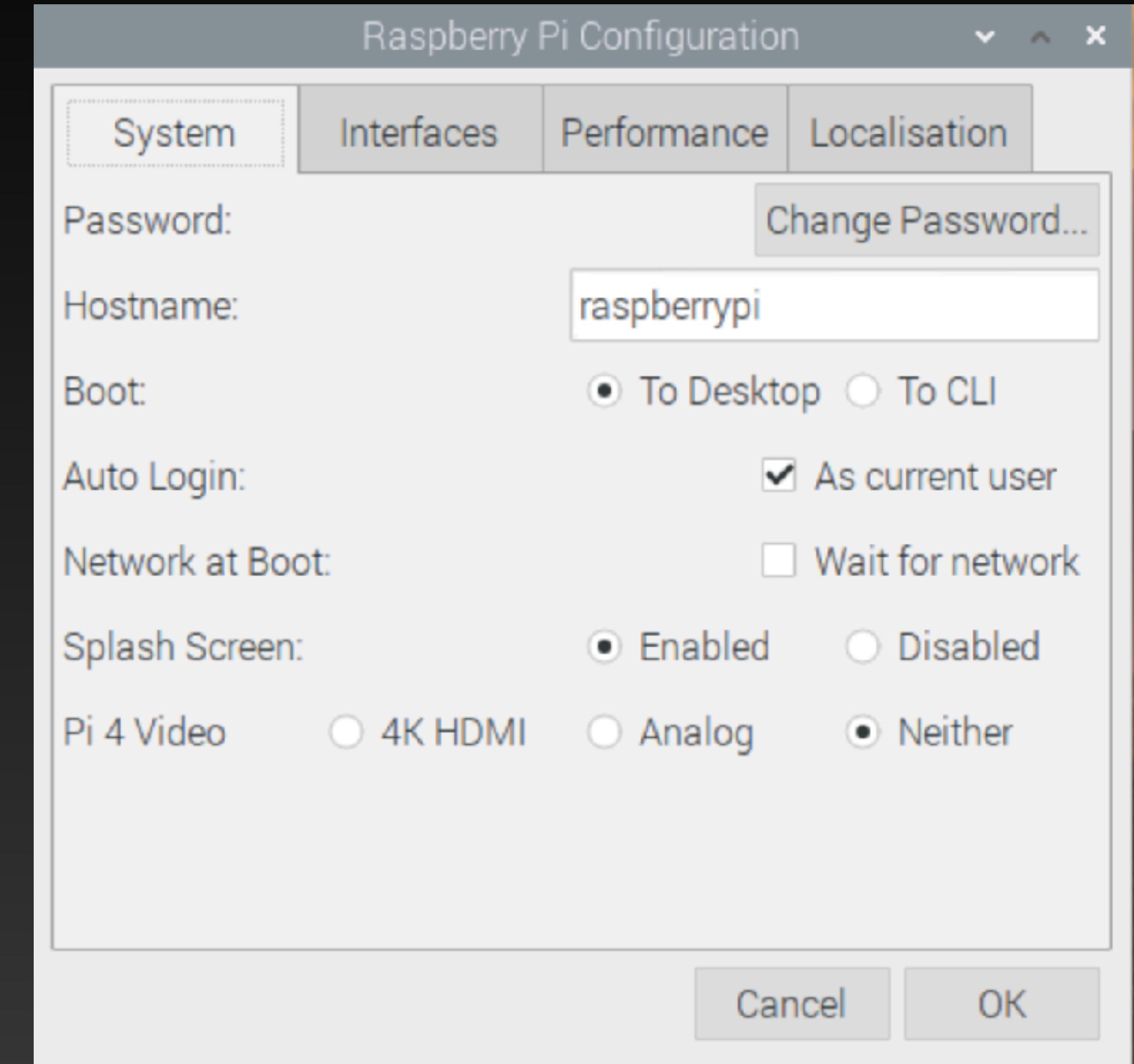
Password — set the password of the `pi` user
(it is a good idea to change the password from the factory default ‘raspberry’)

Boot — select to show the Desktop or CLI (command line interface) when your Raspberry Pi starts

Auto Login — enabling this option will make the Raspberry Pi automatically log in whenever it starts

Network at Boot — selecting this option will cause your Raspberry Pi to wait until a network connection is available before starting

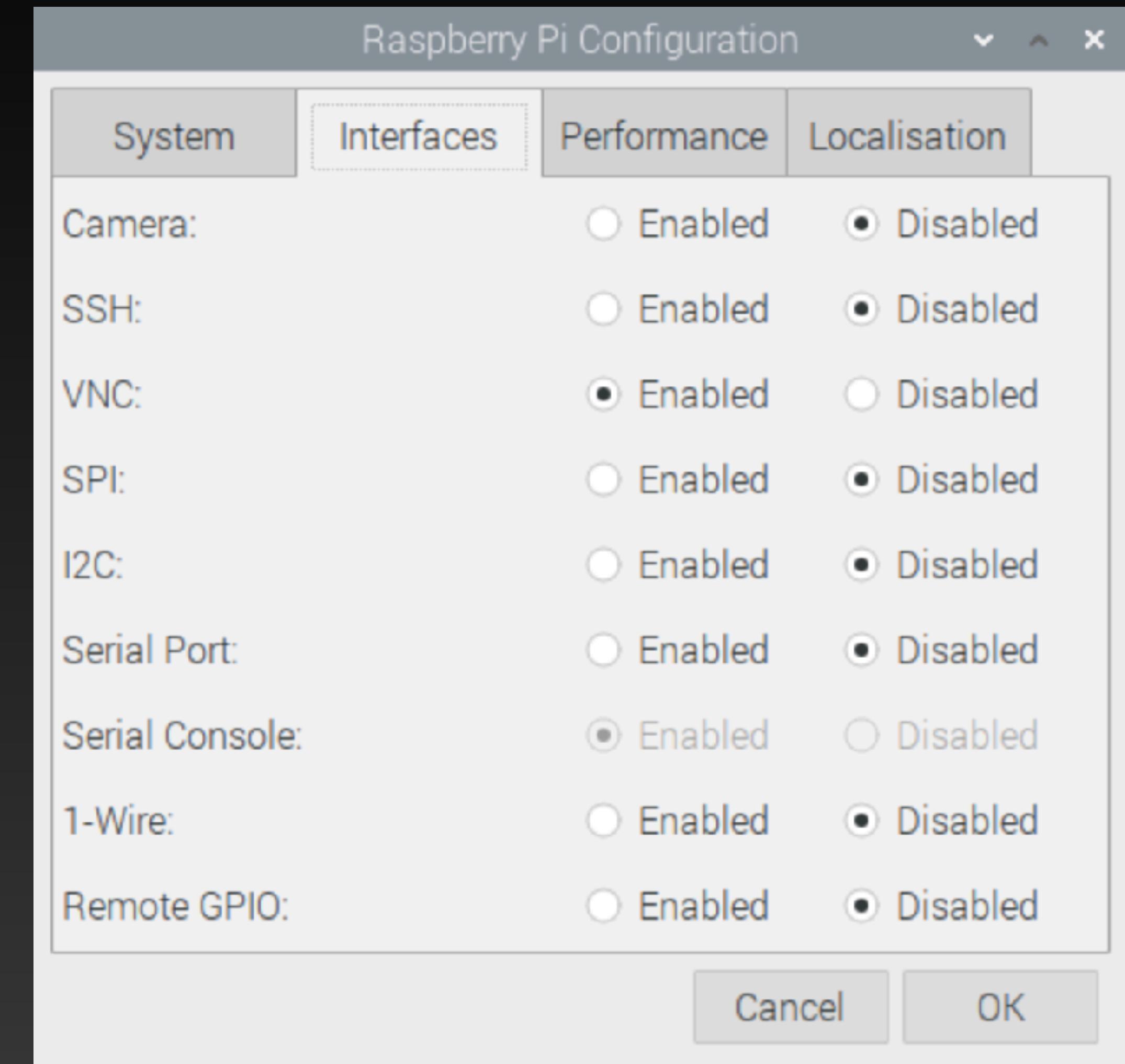
Splash Screen — choose whether or not to show the splash (startup) screen when your Raspberry Pi boots



Configuring your Raspberry Pi

Interfaces

You can link devices and components to your Raspberry Pi using a lot of different types of connections. The Interfaces tab is where you turn these different connections on or off, so that your Raspberry Pi recognizes that you've linked something to it via a particular type of connection.



Configuring your Raspberry Pi

Camera — enable the Raspberry Pi Camera Module

SSH — allow remote access to your Raspberry Pi from another computer using SSH

VNC — allow remote access to the Raspberry Pi Desktop from another computer using VNC

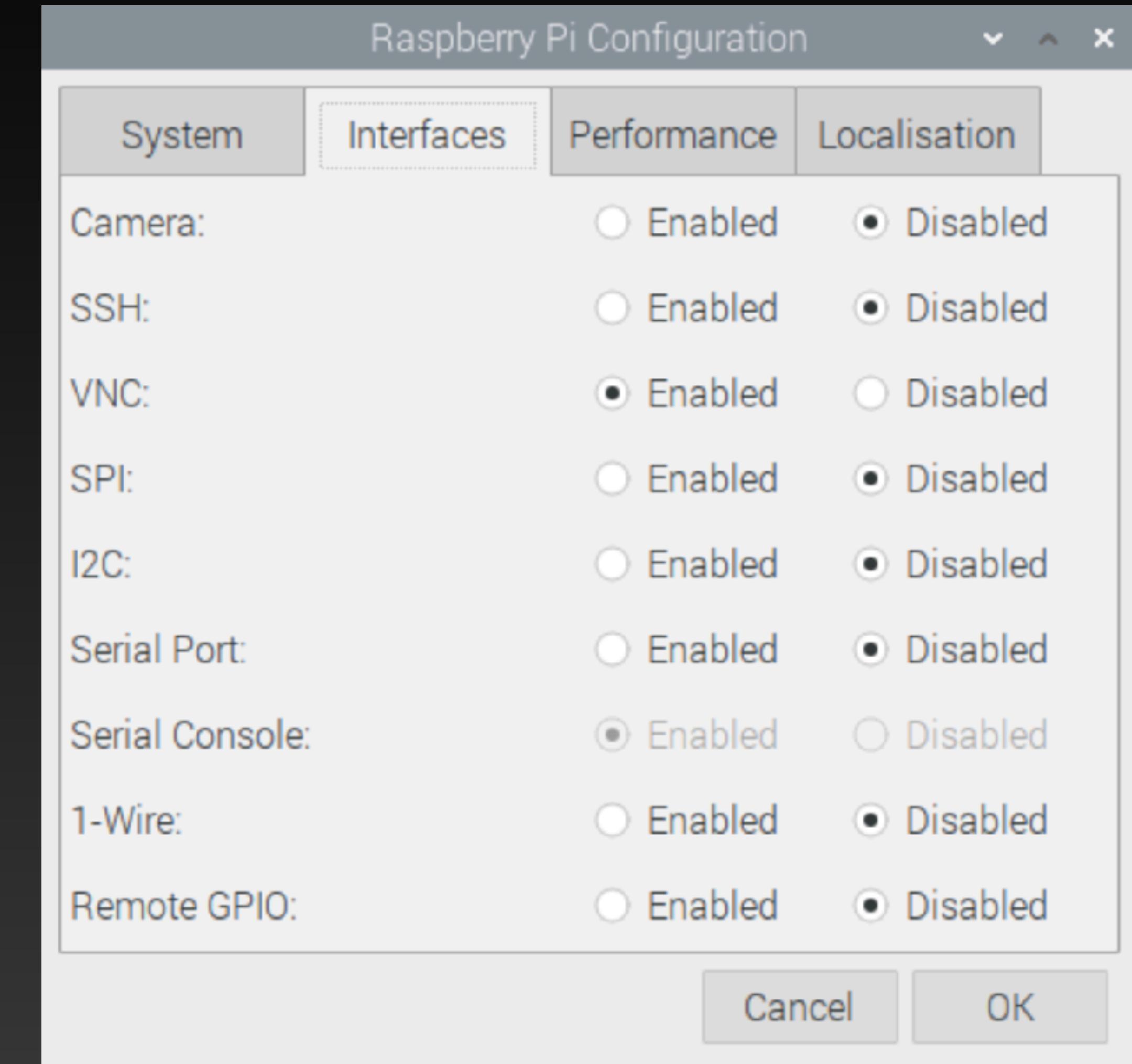
SPI — enable the SPI GPIO pins

I2C — enable the I2C GPIO pins

Serial — enable the Serial (Rx, Tx) GPIO pins

1-Wire — enable the 1-Wire GPIO pin

Remote GPIO — allow access to your Raspberry Pi's GPIO pins from another computer

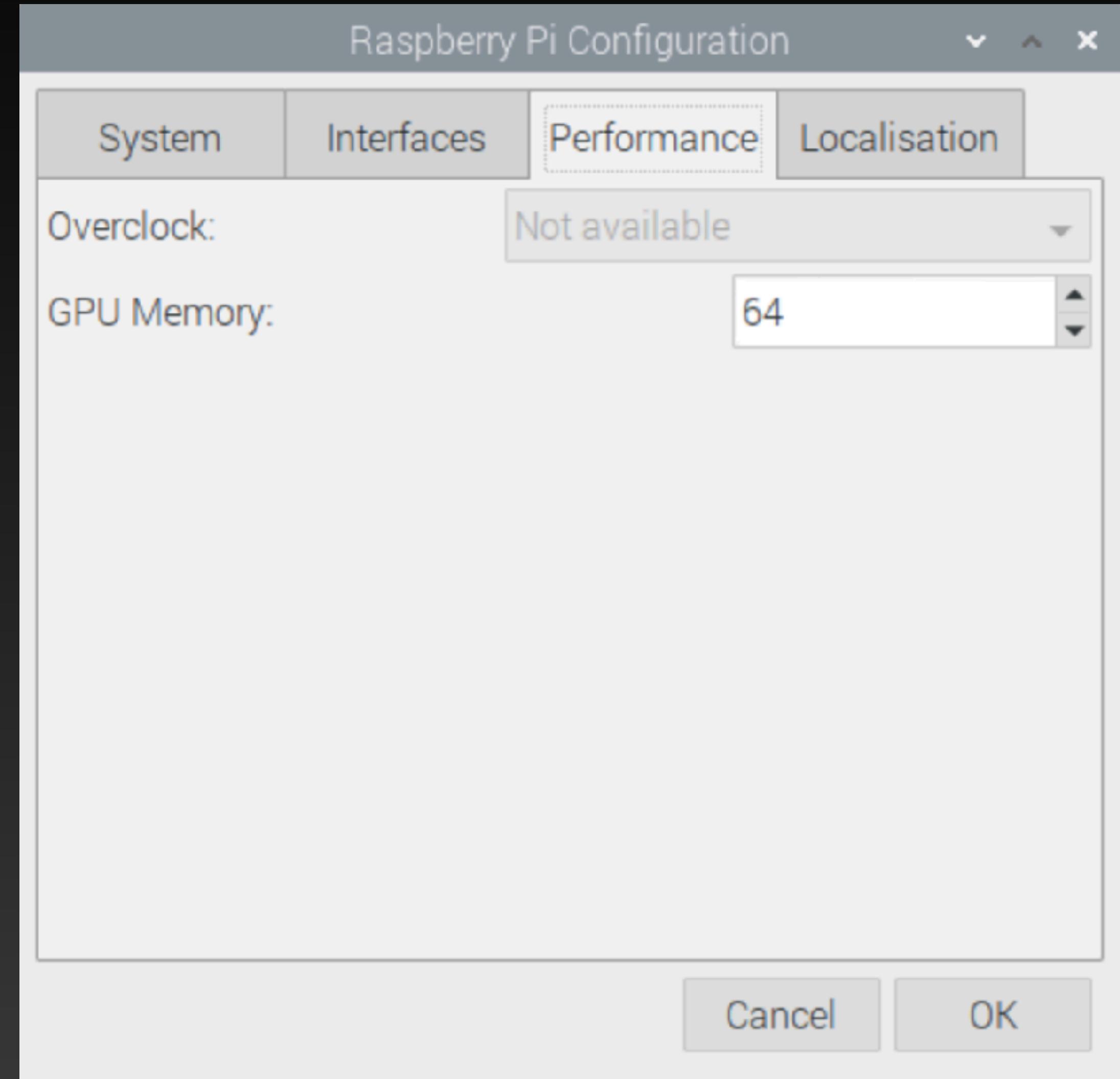


Configuring your Raspberry Pi

Performance

If you need to do so for a particular project you want to work on, you can change the performance settings of your Raspberry Pi in this tab.

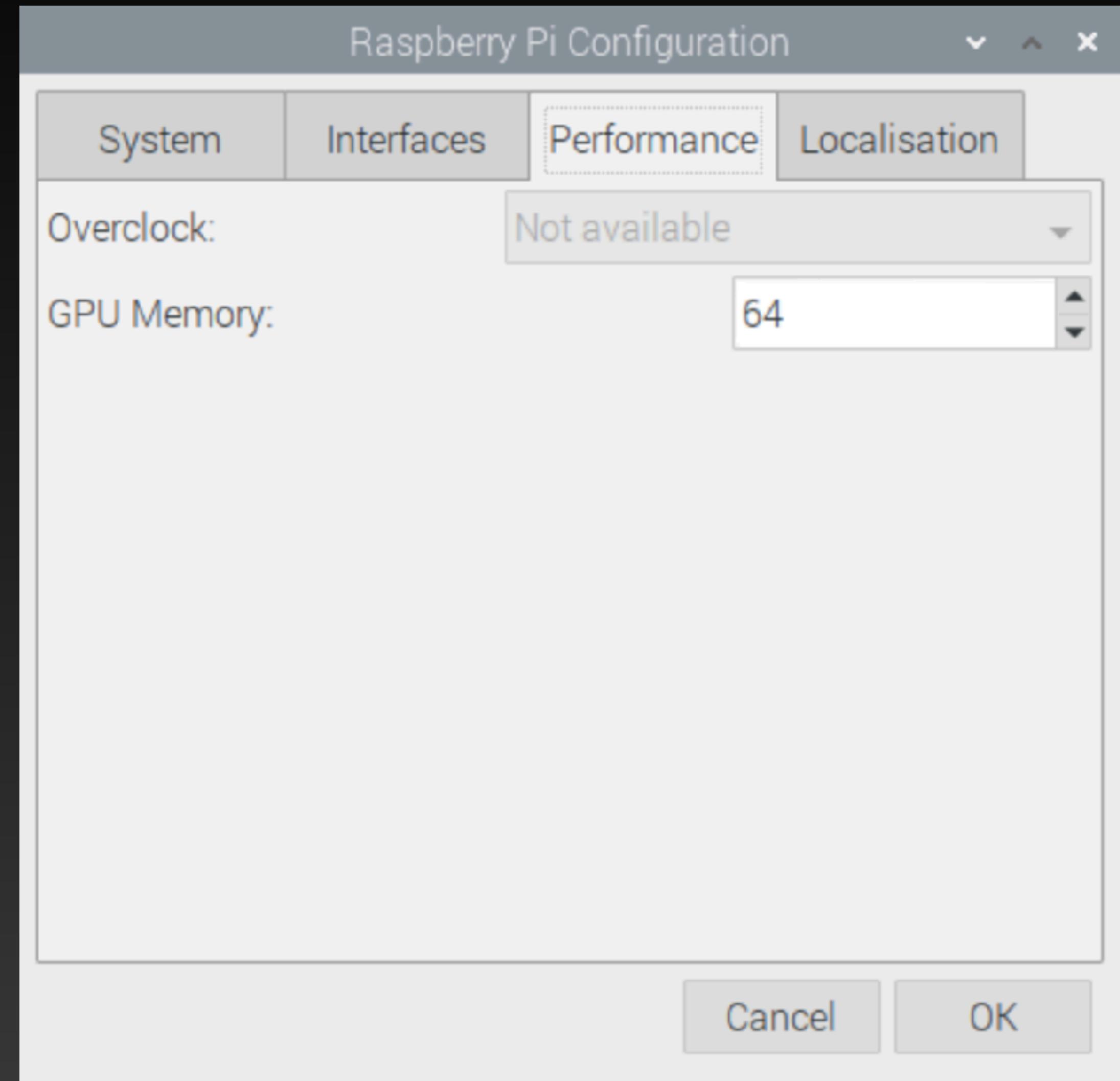
Warning: Changing your Raspberry Pi's performance settings may result in it behaving erratically or not working.



Configuring your Raspberry Pi

Overclock – change the CPU speed and voltage to increase performance

GPU Memory – change the allocation of memory given to the GPU



Configuring your Raspberry Pi

Localisation

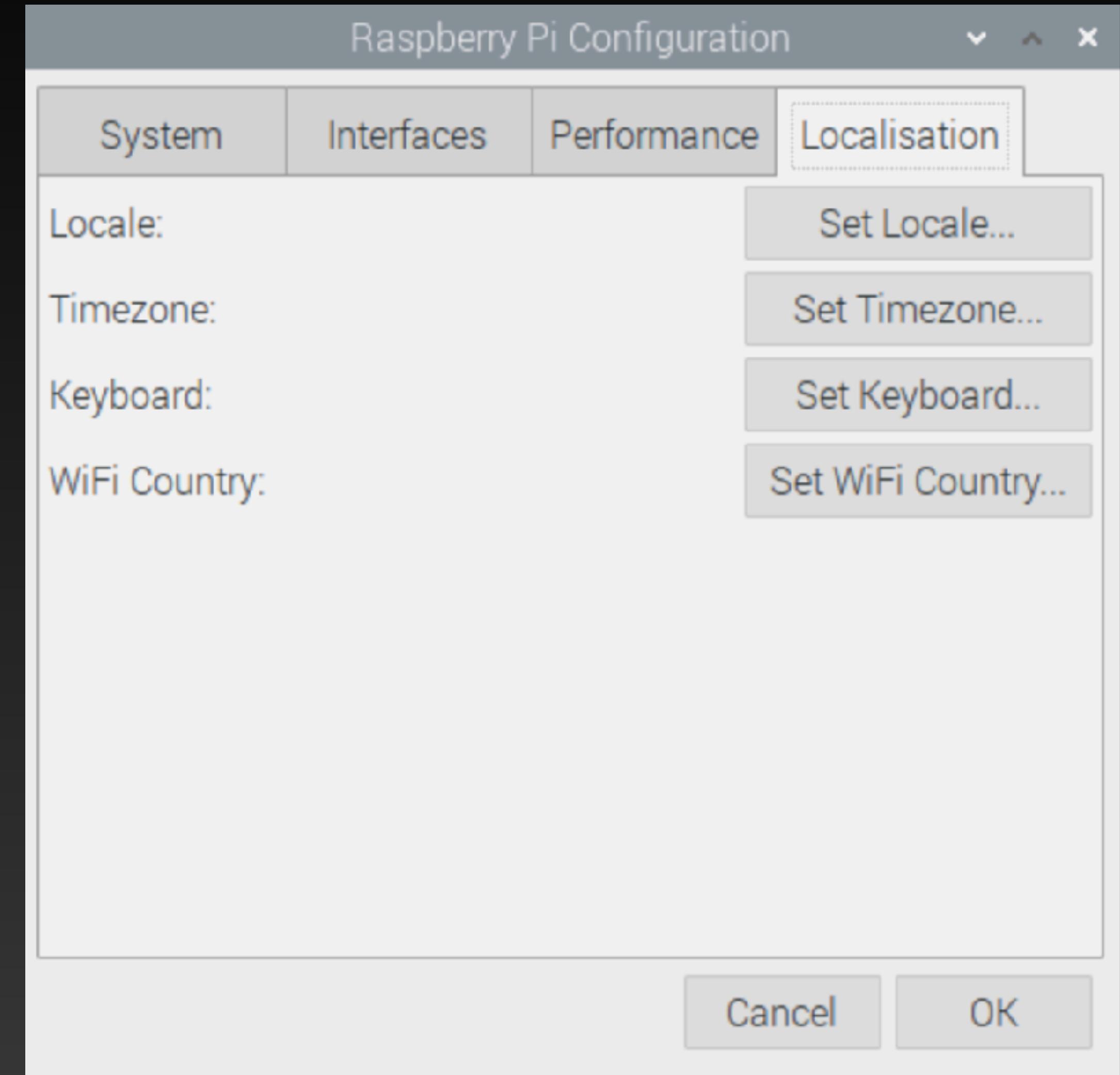
This tab allows you to change your Raspberry Pi settings to be specific to a country or location.

Locale — set the language, country, and character set used by your Raspberry Pi

Timezone — set the time zone

Keyboard — change your keyboard layout

WiFi Country — set the WiFi country code

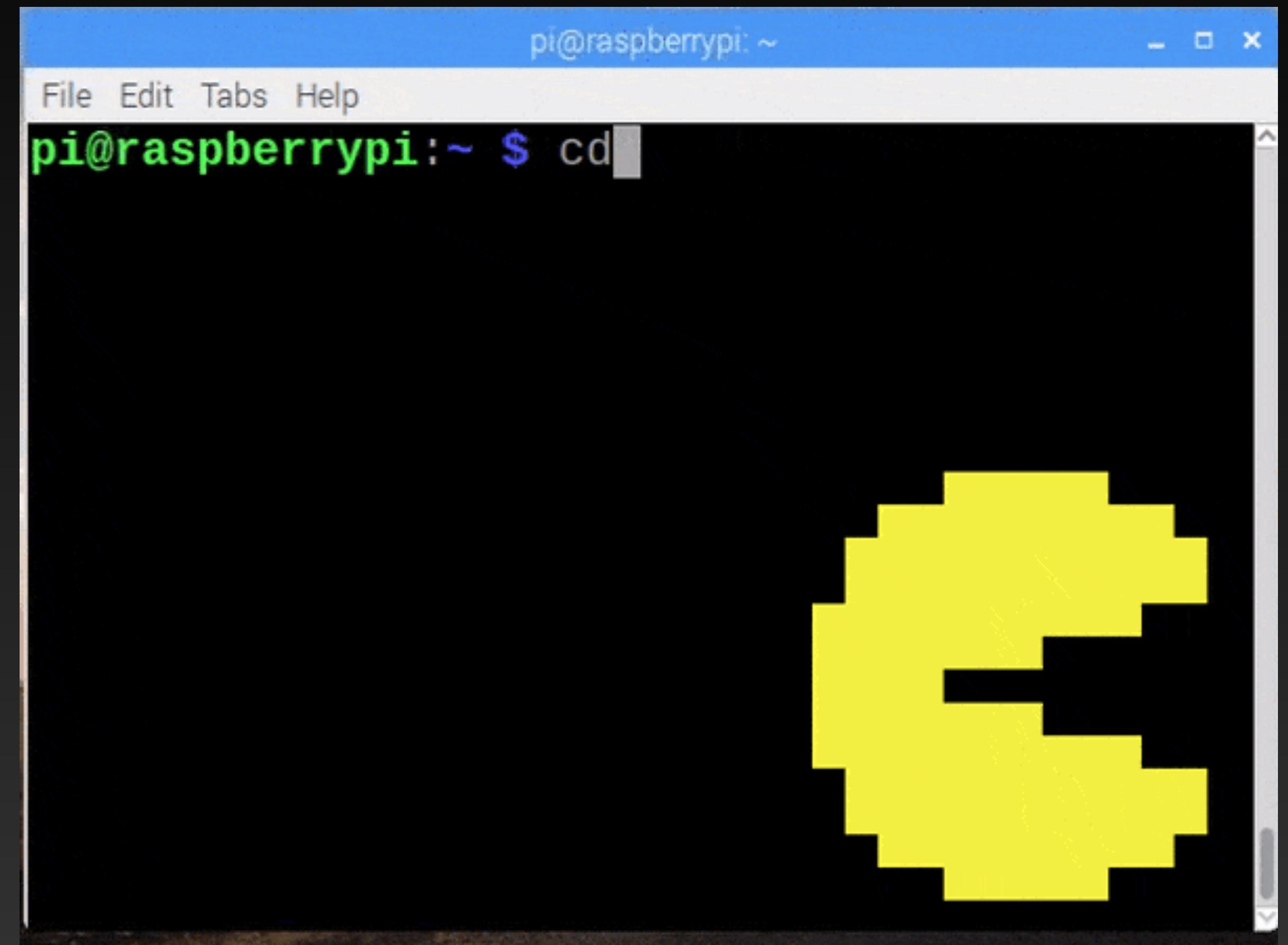


Hack the Raspberry Pi's Terminal

Introduction

Hack the Raspberry Pi's terminal and learn basic cybersecurity skills through a Pac-Man–themed treasure hunt.

You will take the first steps towards being an awesome ethical hacker! This project will take you on a journey through your Raspberry Pi's terminal and you'll learn to navigate the terminal and protect your computer against nasty attackers.



Hack the Raspberry Pi's Terminal

You will learn some basic hacking and find out about basic cybersecurity concepts and threats.

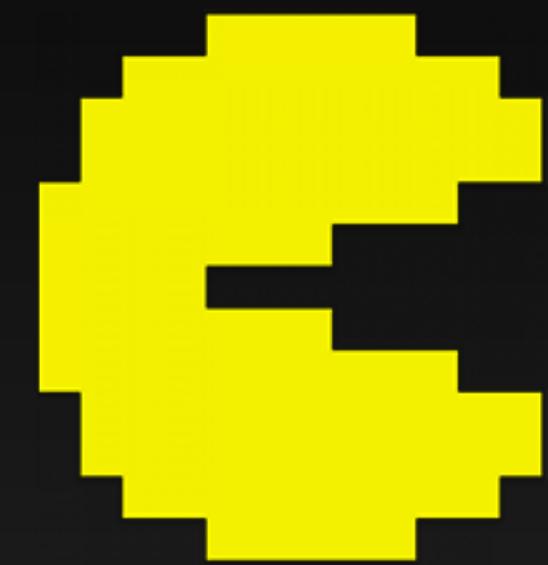
You will also learn how to do these things in the terminal:

- Download and run a script
- Navigate your computer
- Create a directory
- Copy and move files around your computer
- Delete files

Hack the Raspberry Pi's Terminal

Your mission

For this project, you will be Pac-Man:



This is the situation: there is a huge problem with your computer system — it's being attacked by nasty viruses. These viruses are the Pac-Man ghosts:



Pac-Man treasure hunt project

Your goal is to catch and get rid of all of these ghosts! Once you have caught all the ghosts, you can go about collecting your treasure.

To catch the ghosts, you will need to navigate the maze that is your computer. You can navigate your Raspberry Pi computer by using **the command line**.

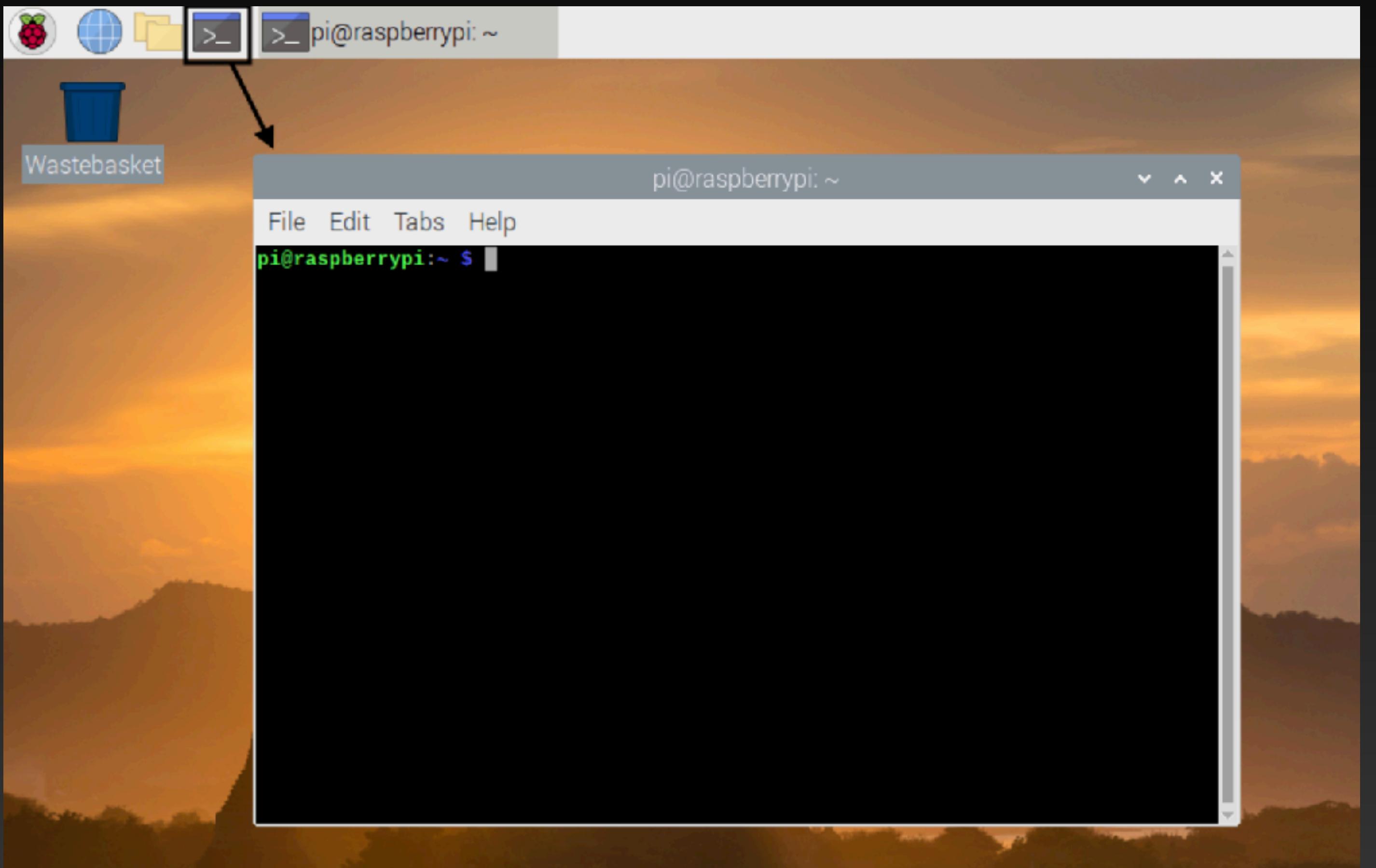
The command line is a text interface for your computer that you can use to give commands to the computer's operating system. From the command line, you can navigate through files and folders on your computer, just as you would with Finder on macOS or Explorer on Windows. The difference is that the command line is fully text-based! The command line is a powerful tool, and with it you can run programs, write scripts to automate tasks, and combine simple commands to handle more difficult jobs.

Pac-Man treasure hunt project

So, to complete the treasure hunt you must:

- Find the ghosts
- Trap the ghosts
- Collect your treasure

To access the command line on your Raspberry Pi, open a terminal window by clicking on the Terminal icon at the top of the screen, or select Accessories and then Terminal in the menu.



Pac-Man treasure hunt project

Start the treasure hunt

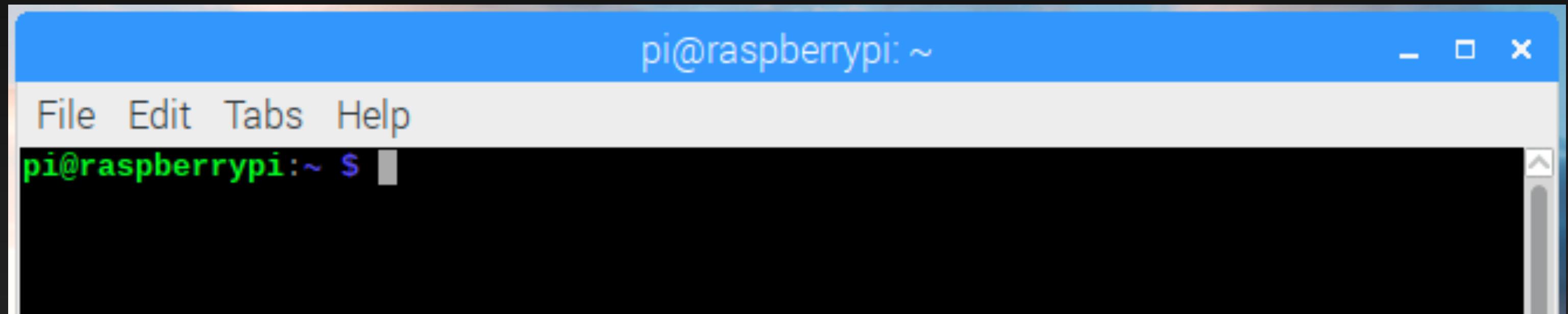
To start the treasure hunt, you will need to download a file that will unleash the Pac-Man ghosts! Please note that you will not download real viruses by completing this step. The files you'll download are completely safe and cannot do any damage to your computer.

In the terminal, you can download and open files or scripts very easily using bash. Bash is a program that listens to your commands and does what you tell it to do. Hackers are concerned with problems in computer systems and programs, and they need to give commands to find these problems and fix them. Therefore, they need to be good at using commands in programs like bash.

Pac-Man treasure hunt project

To access bash, make sure you have opened a terminal window.

In the terminal, you will see \$. This is called a shell prompt. It appears when the terminal is ready to accept a command.



Next to the \$(the shell prompt), type or copy and paste:

```
wget -O - https://rpf.io/pacmanstart | bash
```

Pac-Man treasure hunt project

Press the Enter key to run this line of code. You will see a lot of text appear in the terminal. This means it is downloading the files to start the treasure hunt.

Note: the `| bash` command at the end tells bash to automatically run the file after it is downloaded. Using this command is generally not very secure, because it means the downloaded files will be run without you checking them first. However, here it is the easiest way to start the treasure hunt, and it also shows you how easy it is to download viruses.

Pac-Man treasure hunt project

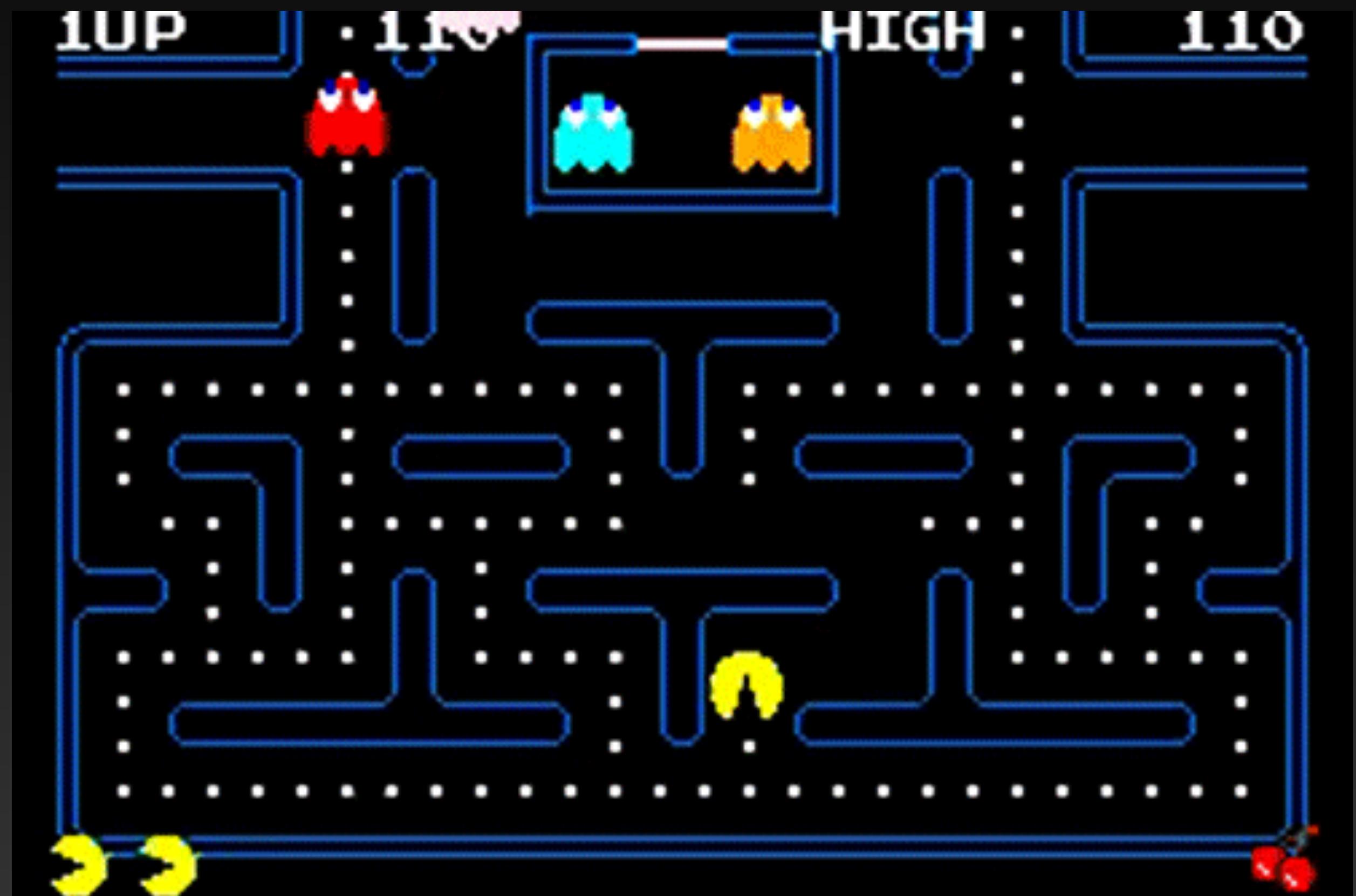
Press Ctrl L(L) to clear the terminal window.

Next you'll learn how to navigate the command line so you can start finding and catching the ghosts.

```
pi@raspberrypi01:~  
File Edit Tabs Help  
sp00ky.txt          100%[=====>]    29  --.KB/s  in 0s  
2022-03-31 19:49:22 (745 KB/s) - '/home/pi/Documents/sp00ky.txt' saved [29/29]  
--2022-03-31 19:49:22-- https://raw.githubusercontent.com/raspberrypilearning/pacman-terminal/master/en/resources/funky.txt  
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 2606:50c0:8000::154, 2606:50  
c0:8003::154, 2606:50c0:8002::154, ...  
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|2606:50c0:8000::154|:443...  
connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 17 [text/plain]  
Saving to: '/home/pi/Documents/funky.txt.2'  
  
funky.txt.2          100%[=====>]    17  --.KB/s  in 0s  
2022-03-31 19:49:22 (435 KB/s) - '/home/pi/Documents/funky.txt.2' saved [17/17]  
--2022-03-31 19:49:23-- https://raw.githubusercontent.com/raspberrypilearning/pacman-terminal/master/en/resources/sue.txt  
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 2606:50c0:8001::154, 2606:50  
c0:8000::154, 2606:50c0:8003::154, ...  
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|2606:50c0:8001::154|:443...  
connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 26 [text/plain]  
Saving to: '/home/pi/Music/sue.txt.3'  
  
sue.txt.3          100%[=====>]    26  --.KB/s  in 0s  
2022-03-31 19:49:23 (248 KB/s) - '/home/pi/Music/sue.txt.3' saved [26/26]  
pi@raspberrypi01:~ $ █
```

Pac-Man treasure hunt project

In this step, you will learn how to get Pac-Man to navigate your Raspberry Pi's terminal, just like when Pac-Man moves around his maze.



Pac-Man treasure hunt project

Listing directory content

The `ls` command looks at the folder you are in and then lists the files and directories inside it (`ls` stands for ‘list’). When you’re using the terminal, folders are called directories.

After the shell prompt, type `ls` and press Enter. Make sure you type the letter `l` as in “lime” and not the number 1.

```
pi@raspberrypi:~ $ ls
Desktop      Downloads    Pictures    python_games  Scratch Projects  Videos
Documents    Music        Public      Scratch       Templates
```

Pac-Man treasure hunt project

```
pi@raspberrypi:~ $ ls  
Desktop    Downloads  Pictures  python_games  Scratch Projects  Videos  
Documents  Music     Public    Scratch      Templates
```

Here, you cannot see any files, because there aren't any, but you can see lots of directories.

The directory, or folder you are in, therefore contains these directories:

Desktop, Downloads, Pictures, python_games, Scratch
Projects, Videos, Documents, Music, Public, Scratch, and Templates

Pac-Man treasure hunt project

Print working directory

`pwd` stands for ‘print working directory’. This command shows where you are in your Raspberry Pi’s file system, meaning it tells you what directory you are currently in in the command line — your working directory! You can use this command at any time to not get lost.

Type `pwd` and press Enter.

```
pi@raspberrypi:~ $ ls  
Desktop    Downloads   Pictures   python_games  Scratch Projects Videos  
Documents  Music      Public     Scratch       Templates  
pi@raspberrypi:~ $ pwd  
/home/pi  
pi@raspberrypi:~ $ █
```

Pac-Man treasure hunt project

```
pi@raspberrypi:~ $ ls  
Desktop    Downloads  Pictures  python_games  Scratch Projects  Videos  
Documents  Music     Public    Scratch       Templates  
pi@raspberrypi:~ $ pwd  
/home/pi  
pi@raspberrypi:~ $ █
```

Here you can see that you are in the `/home/pi` directory. This means you're inside the `pi` directory, which itself is stored inside the `home` directory. (The `/` symbols aren't part of the directory names; the command line just uses them to show you directories.)

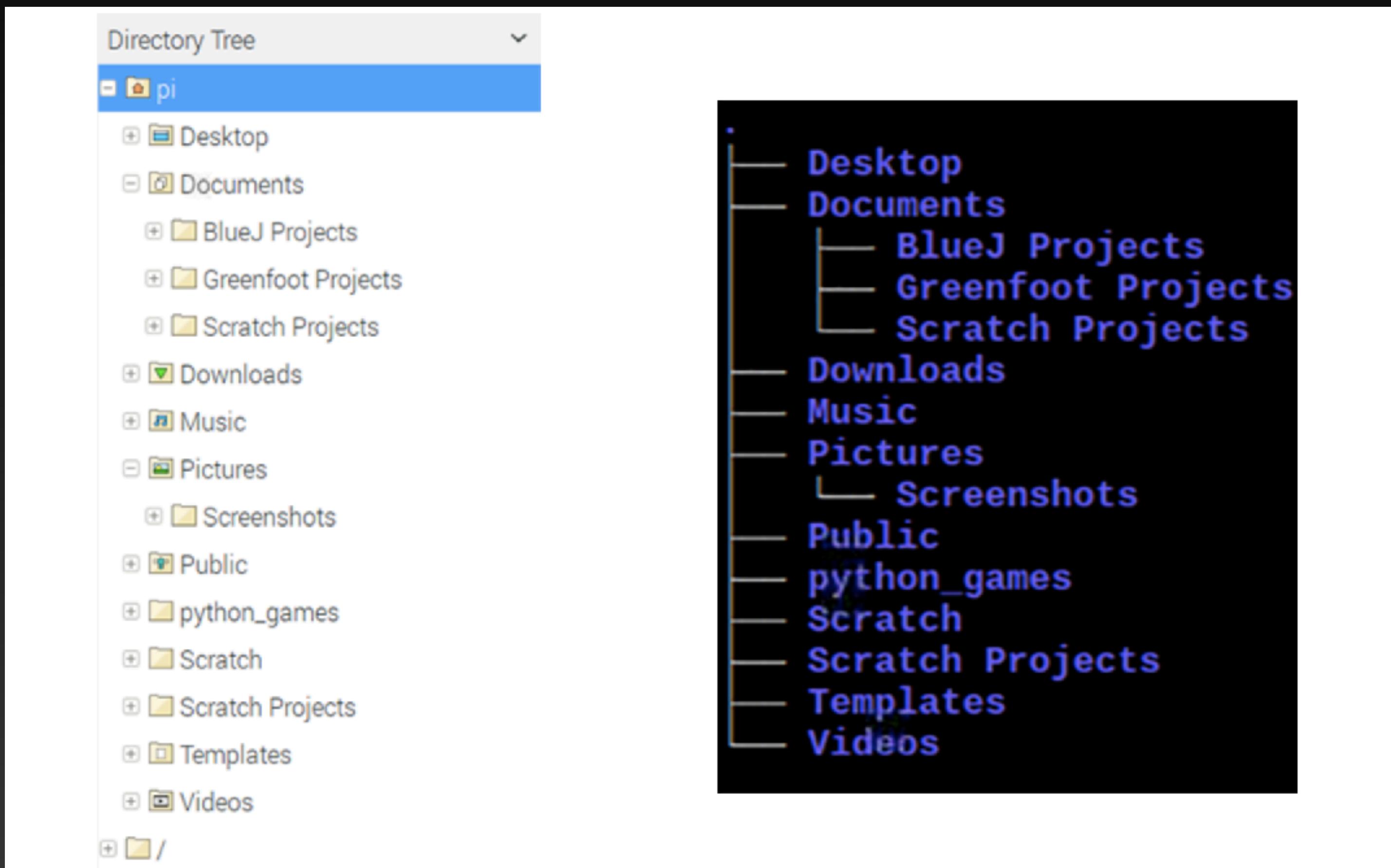
Note:

This directory is your `home` directory. It is where you can save all your files.

If you are using a different username to `pi`, your home directory will have a different directory, but it will always have the shortcut `~`, as you can see at the start of the shell prompt.

Pac-Man treasure hunt project

The file system is the way files and directories on your computer are organized. A computer's file system usually has a tree structure:



Pac-Man treasure hunt project

Change directory

`cd` stands for ‘change directory’. Just the same as you can click on a folder icon in a graphic file manager to go into it, `cd` lets you go into the directory you tell it. To get into a directory, just type the command `cd` followed by a space, and then the name of the directory and a forward slash `/`.

To try moving into a different directory, type `cd Documents/` and press Enter.

```
pi@raspberrypi:~ $ cd Documents/  
pi@raspberrypi:~/Documents $ █
```

Pac-Man treasure hunt project

Now you've navigated into the Documents directory!

The terminal tells you here that you are in Documents:

```
pi@raspberrypi:~ $ cd Documents/  
pi@raspberrypi:~/Documents $ █
```

Pac-Man treasure hunt project

Moving between directories

You can move up (or back, depending on how you want to imagine it) one directory in the file system by typing `cd` followed by a space and two full stops `..`, and then pressing Enter.

`cd ..` navigates up from `home/pi/Documents` back to `home/pi`.

You can quickly check you're in the right directory using the `pwd` command:

```
pi@raspberrypi:~/Documents $ cd ..
pi@raspberrypi:~ $ pwd
/home/pi
pi@raspberrypi:~ $ █
```

Pac-Man treasure hunt project

Now you're ready to catch your first ghost!

Catch and quarantine your first ghost

In this step, you will create a quarantine folder to put your ghosts in. You will then catch all of the ghosts and stick them in this folder so that they stop ghosting around and harming your computer like viruses do.

A quarantine folder isolates suspicious files so they cannot harm your computer. They are usually made automatically by antivirus software and have additional programming attached to them — you will simulate what it is like to create one.

Pac-Man treasure hunt project

First off, you will learn to create your own directories in your file system!

Next to the shell prompt, create a quarantine folder by typing `mkdir quarantine/` and pressing Enter.

`mkdir` stands for ‘make directory’. The command you typed in created a new directory called `quarantine` inside your `home` directory. In the image, you can see that I have also used the `ls` command to check that my directory exists:

```
pi@raspberrypi:~ $ mkdir quarantine/
pi@raspberrypi:~ $ ls
Desktop      Downloads    Pictures    python_games  Scratch      Templates
Documents    Music        Public      quarantine   Scratch Projects  Videos
pi@raspberrypi:~ $
```

Pac-Man treasure hunt project

Let's catch our first ghost!

Navigate into the **Documents** directory using the commands you learned in the last step.

List all the files and directories in the **Documents** directory using the commands you learned in the last step.

Look at the directories. Do you see anything strange that looks like it should not be there?

Pac-Man treasure hunt project

That strange file named `sp00ky.txt` represents a virus. If you've spotted it, you have found the first ghost!

You've found Spooky!



You might have noticed that instead of being called `spooky.txt`, the virus is called `sp00ky.txt`. Hackers, both ethical and not so ethical, replace certain letters with numbers so they cannot be picked up in simple keyword searches. Keep that in mind when you are looking for the other ghosts.

Pac-Man treasure hunt project

Copying files

Copy the ghost and put it in the quarantine folder by typing:

```
cp sp00ky.txt ~/quarantine/
```

`cp` (for ‘copy’) is followed by the name of the file you want to copy and then the directory you want it copied into. Here, we copied the file `sp00ky.txt` and placed it in the `quarantine` folder in the home directory (~).

Pac-Man treasure hunt project

Removing files

Now remove the ghost by typing:

```
rm sp00ky.txt
```

Note: be careful with using `rm` — it deletes things from your computer permanently! There isn't an 'undelete' command, so once you delete something with `rm`, it's gone forever. This is why we copied the file into a new directory before removing it.

You're now ready to safely catch all the ghosts!

Pac-Man treasure hunt project

Catch all the ghosts

Now use all the commands you have learned in the last two steps to move around your Raspberry Pi's file system and find, catch, and quarantine as many ghosts as you can.

Here's a reminder of all the commands you need to navigate the terminal:

Pac-Man treasure hunt project

Essential commands

Here is a summary of basic commands for navigating the command line:

- `ls` lists the contents of the folder (directory) you’re currently in:

```
ls
```

- `pwd` stands for ‘print working directory’ and tells you what directory you are currently in:

```
pwd
```

- `cd` stands for ‘change directory’ and navigates to the directory you want to move into:

```
cd directoryname/
```

- `cd ..` navigates up one directory, meaning it moves you from a subfolder into its parent folder:

```
cd ..
```

Additional commands

These are useful for creating and removing directories and files.

- `mkdir` stands for ‘make directory’ and creates a new directory in the directory you are currently in:

```
mkdir directoryname/
```

- `cp` is followed by the name of the file to copy and then the directory you want it copied into:

```
cp filename.filetype path/to/directory
```

- `rm` stands for ‘remove’ and deletes files (careful with this one, you cannot undo it):

```
rm filename.filetype
```

Pac-Man treasure hunt project

Tips:

Remember to copy the ghosts to the quarantine folder before removing them

Navigate to your quarantine folder to see how many ghosts you've caught

If you want to navigate into a directory with a name that has more than one word in it (for example My Movies), use a \ before each blank space in the name (cd My\ Movies)

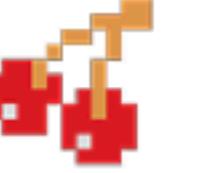
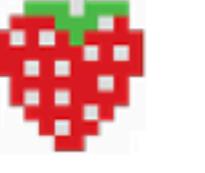
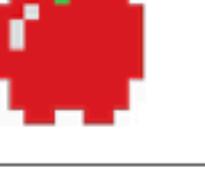
There are seven ghosts in total – can you catch them all?

Pac-Man treasure hunt project

Find the treasure

Now that you have found all the ghosts and put them in a safe folder so they cannot damage your computer, you will look inside the files to collect your treasure.

Here is all the treasure you can collect and the points you get when you find the treasure:

| Treasure | Points |
|---|--------------------|
|  | Cherry 100 |
|  | Strawberry 300 |
|  | Peach 500 |
|  | Apple 700 |
|  | Grapes 1000 |
|  | Galaxian Boss 2000 |
|  | Bell 3000 |
|  | Key 5000 |

Pac-Man treasure hunt project

Collect your treasure

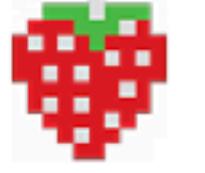
See what treasure you got from catching Spooky!

To view a text file, type `cat` followed by the full name of the file into the terminal window, and then press Enter:

```
cat sp00ky.txt
```

You should see some treasure.

```
pi@raspberrypi:~ $ cd quarantine/  
pi@raspberrypi:~/quarantine $ ls  
sp00ky.txt  
pi@raspberrypi:~/quarantine $ cat sp00ky.txt  
galaxian boss  
grapes  
cherry  
pi@raspberrypi:~/quarantine $
```

| Treasure | Points |
|---|--------------------|
|  | Cherry 100 |
|  | Strawberry 300 |
|  | Peach 500 |
|  | Apple 700 |
|  | Grapes 1000 |
|  | Galaxian Boss 2000 |
|  | Bell 3000 |
|  | Key 5000 |

Pac-Man treasure hunt project

Challenge: get rid of the quarantine folder

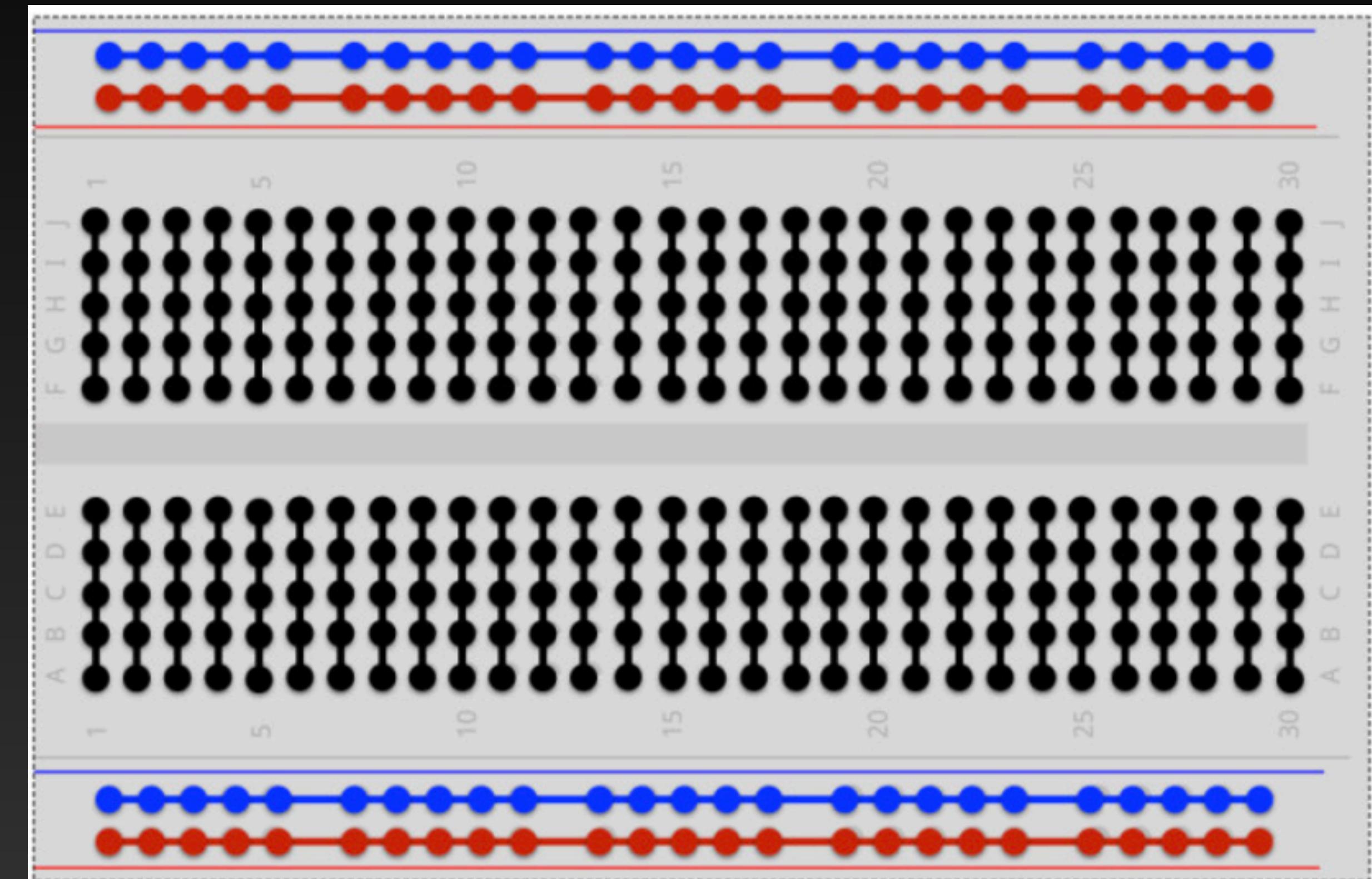
Now that you have finished finding all the ghosts, you must clear your computer of any harmful files.

- Delete the whole **quarantine** folder without copying any of the files in it. Use this link www.raspberrypi.org/documentation/linux/usage/commands.md or look up online how to recursively remove a directory.
- Check the folder has been deleted by listing the contents of your **home** directory.
-

Introducing Breadboard

Most of our projects are tested using a small piece of plastic known as a breadboard. Officially, it's known as a 'solderless breadboard' because it enables you to use circuit parts without soldering them together.

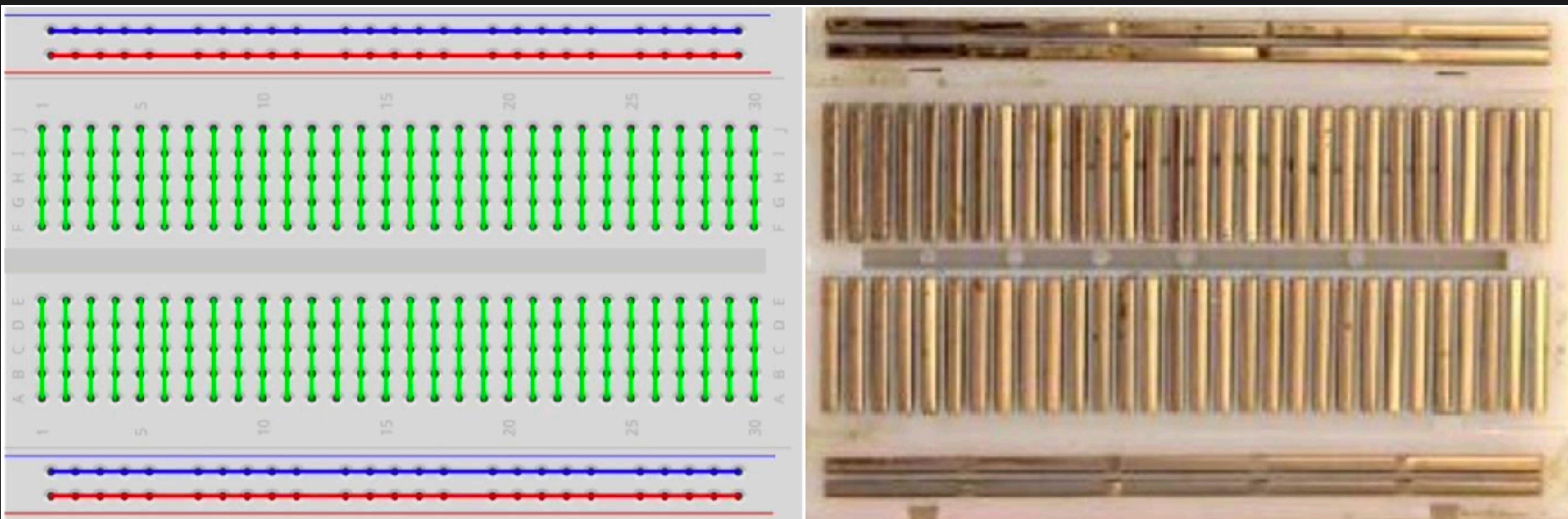
Electrical components are connected by pushing them into the holes in a breadboard. These holes are connected in strips, as shown in the main image. If you push a wire, or a different component, into one hole in a strip, and another wire into the hole next to it, it's as if you'd physically joined (or soldered) the two wires.



Introducing Breadboard

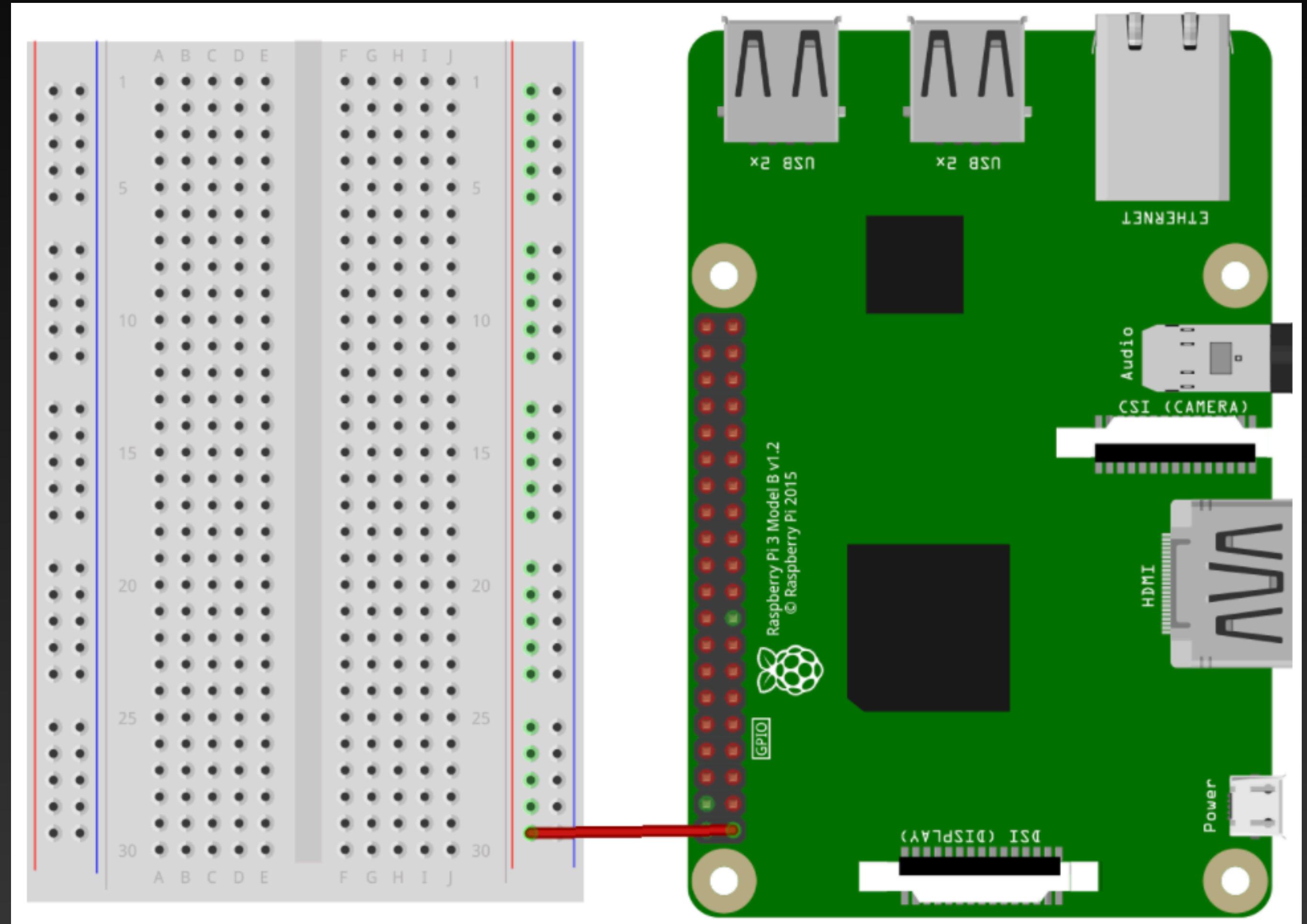
Breadboard

Here we have a small breadboard as an example of how the rows of holes (sockets) are electrically attached. The left picture shows the ways the pins have shared electrical connection and the right picture shows the actual internal metal, which connect these rows electrically.



Test your circuit! An LED project

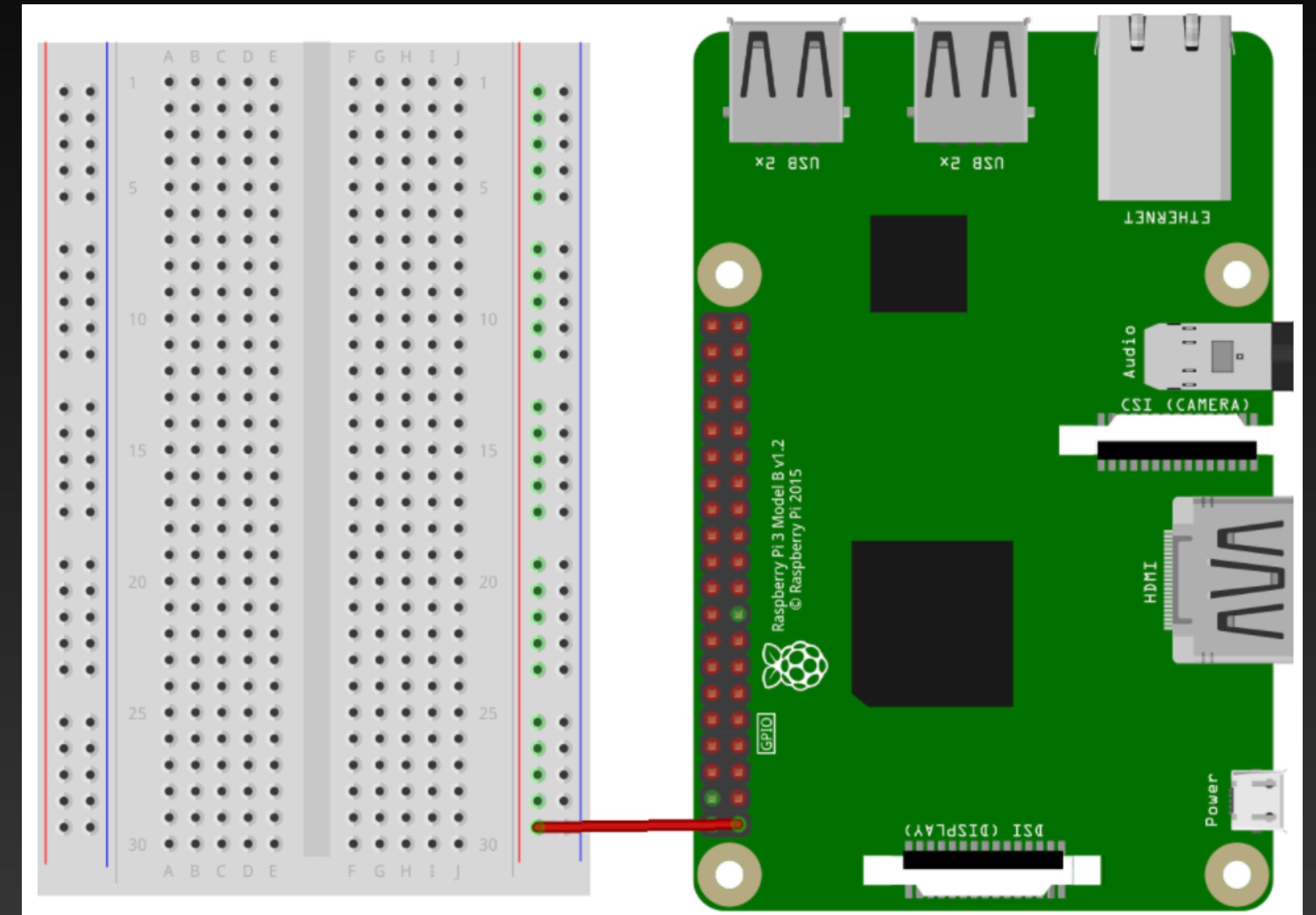
- Using a breadboard.
Breadboard, LED light,
Resistor, Male-to-female
jumper leads, Male-to-
male jumper leads



Test your circuit! An LED project

The breadboard's live rail

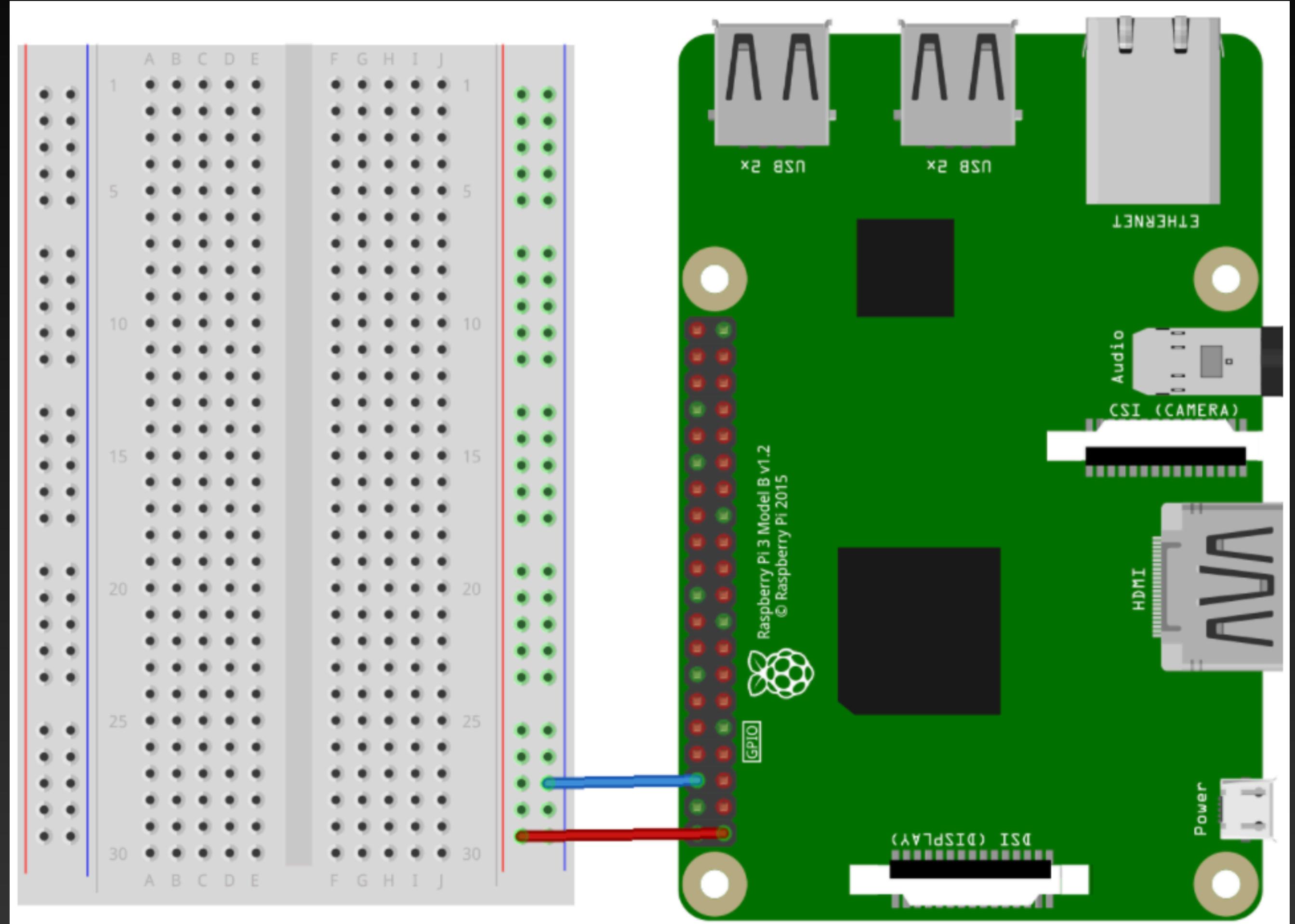
Take a female-to-male jumper lead (the color of the wire doesn't matter) and connect the female end to a 3.3V pin on the Raspberry Pi. Place the male end of the lead into a hole on the red rail on the breadboard.



Test your circuit! An LED project

The breadboard's ground rail

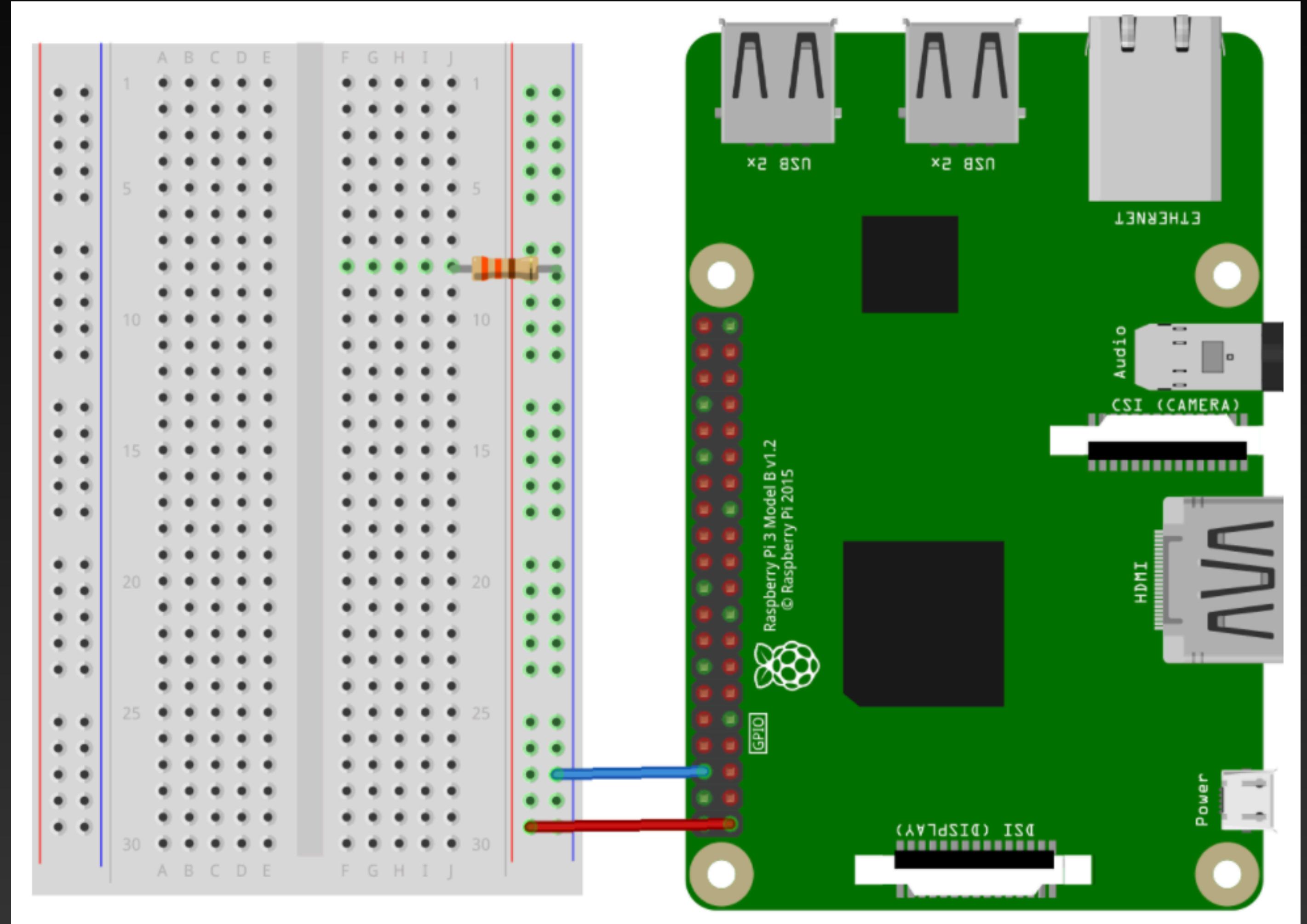
Take another female-to-male jumper and connect the female end to a ground (GND) pin on the Raspberry Pi. The male end goes into a hole on the blue (ground) rail. All blue holes now act as a ground pin.



Test your circuit! An LED project

Add a resistor to the breadboard

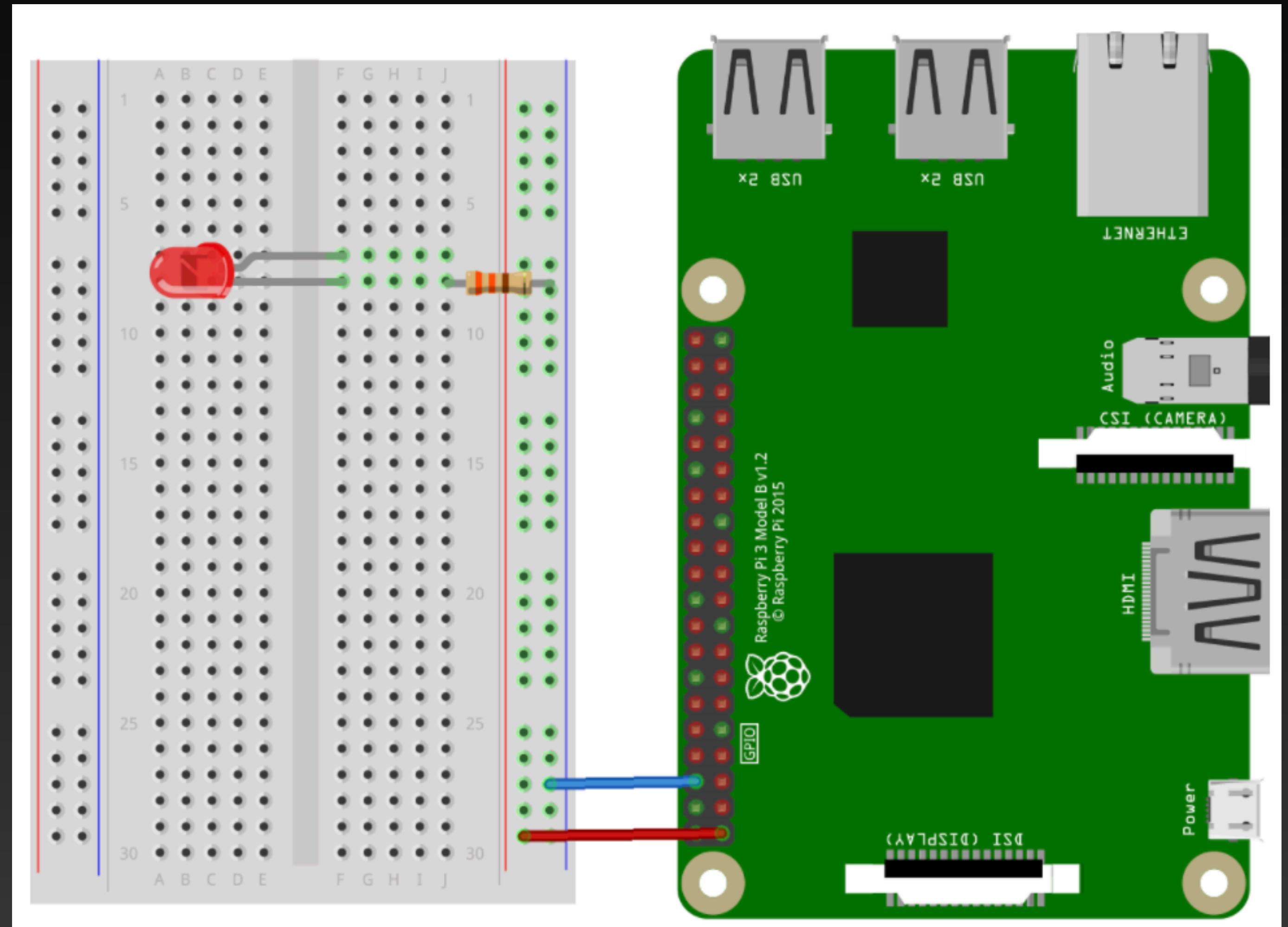
Take a resistor and connect one leg of it to a hole on the ground rail of the breadboard. It's now linked to the ground pin of the Raspberry Pi (via the jumper lead we used in the previous step). Take the other leg and connect it to a hole on the main breadboard.



Test your circuit! An LED project

Add the LED to the breadboard

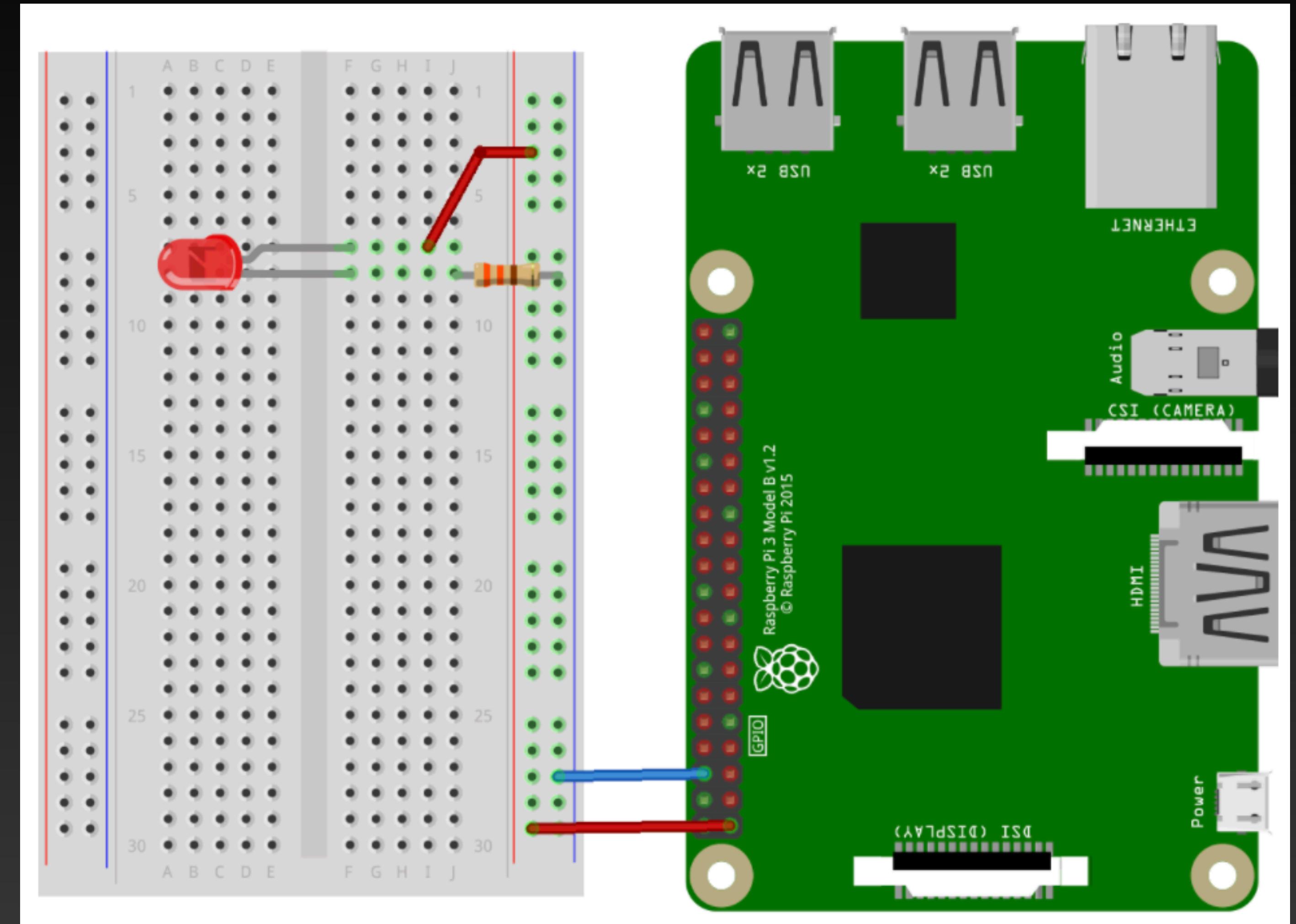
Take an LED component and look at the legs. Notice that one of the legs is shorter than the other. Place the shorter leg in a hole on the same row as the resistor. This leg is now connected to the resistor (which is linked to the ground rail, and therefore to the ground pin on the Raspberry Pi).



Test your circuit! An LED project

Wire up the breadboard

Place the longer leg in a hole on the next row along. Now take another male-to-male jumper lead and place one end in the hole next to the long leg of the LED. Place the other end in a hole on the red live rail to complete the circuit. The LED lights up.

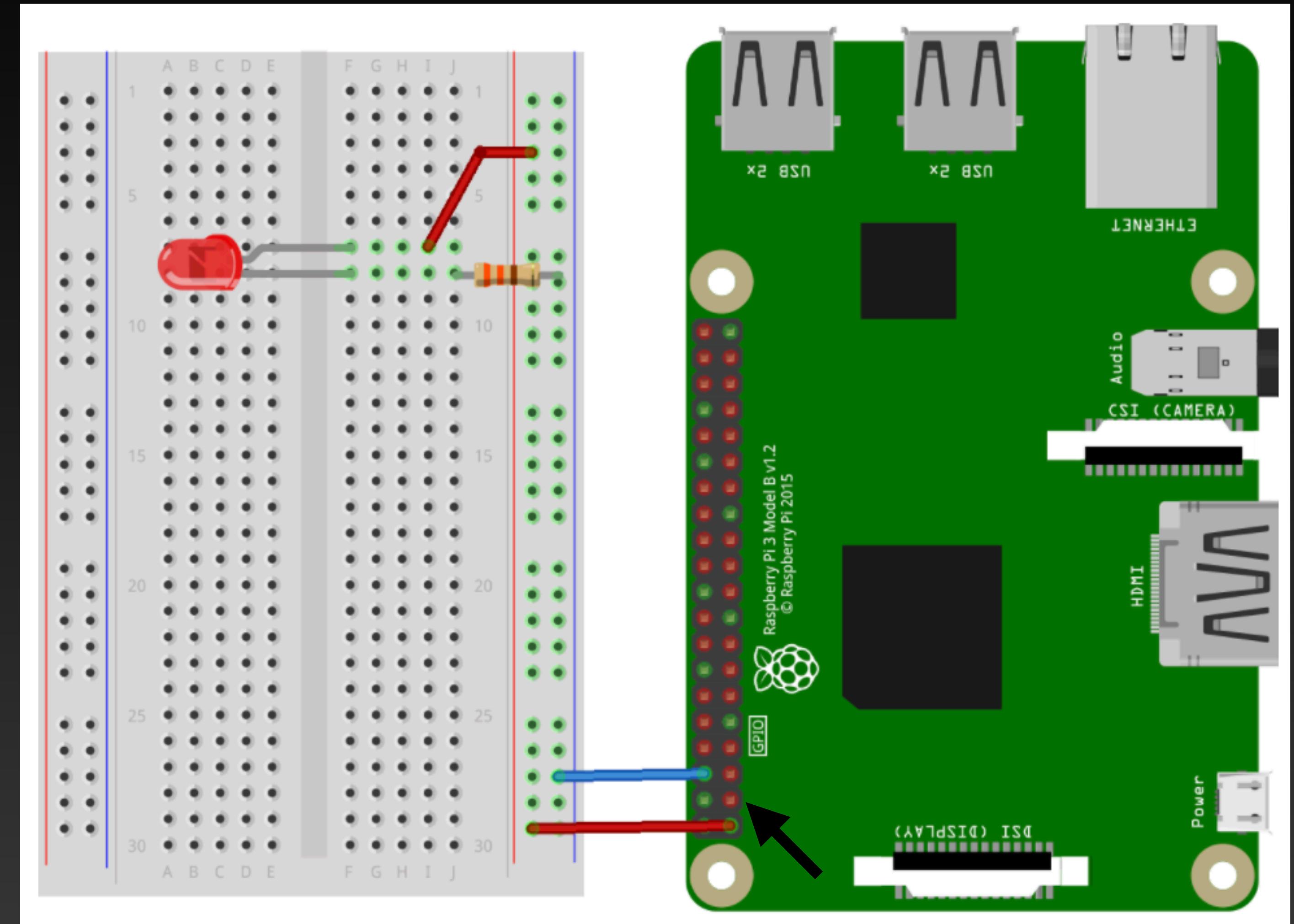


Test your circuit! An LED project

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.

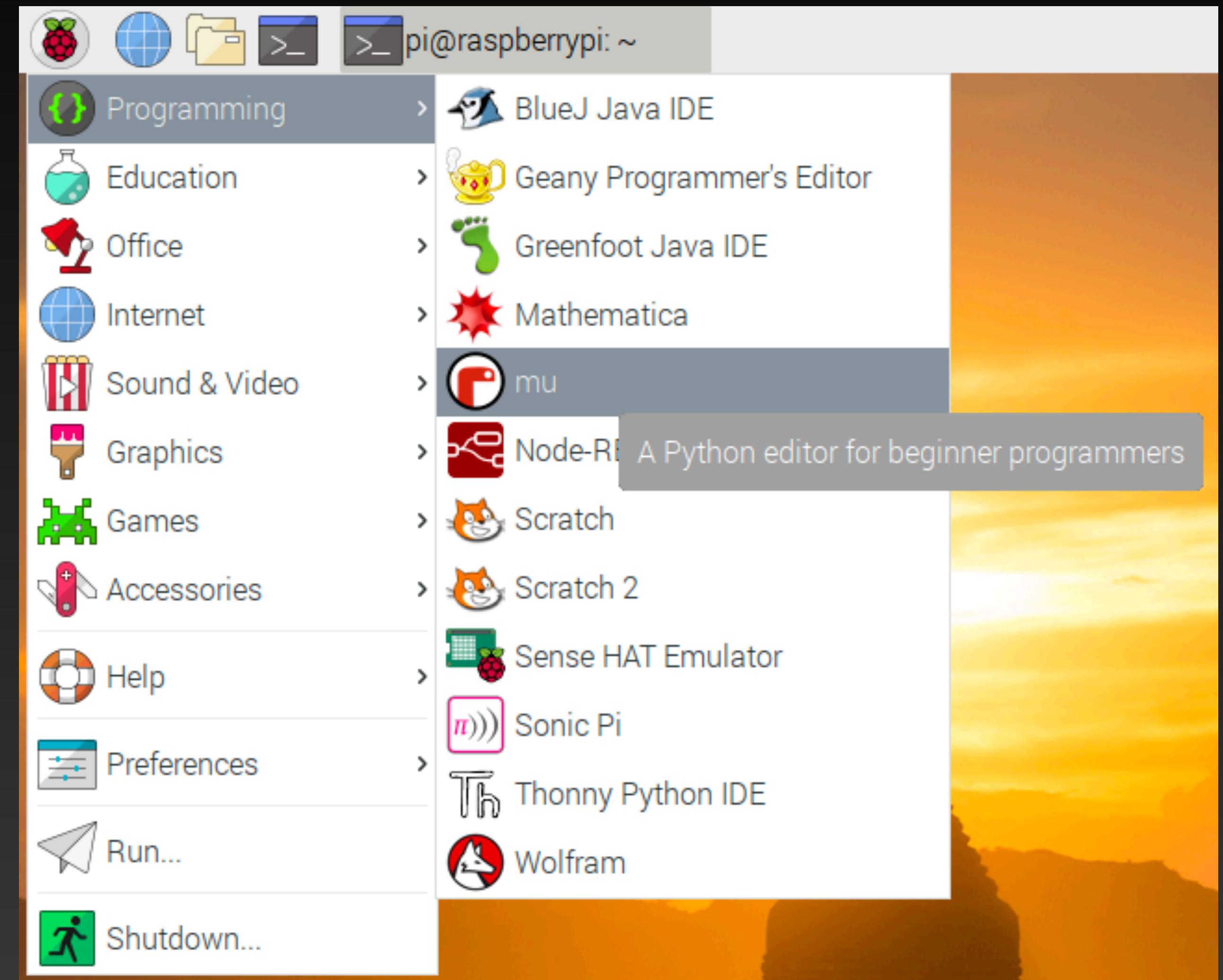
Switch your input to GPIO2



Test your circuit! GPIO Zero library

How to open Mu

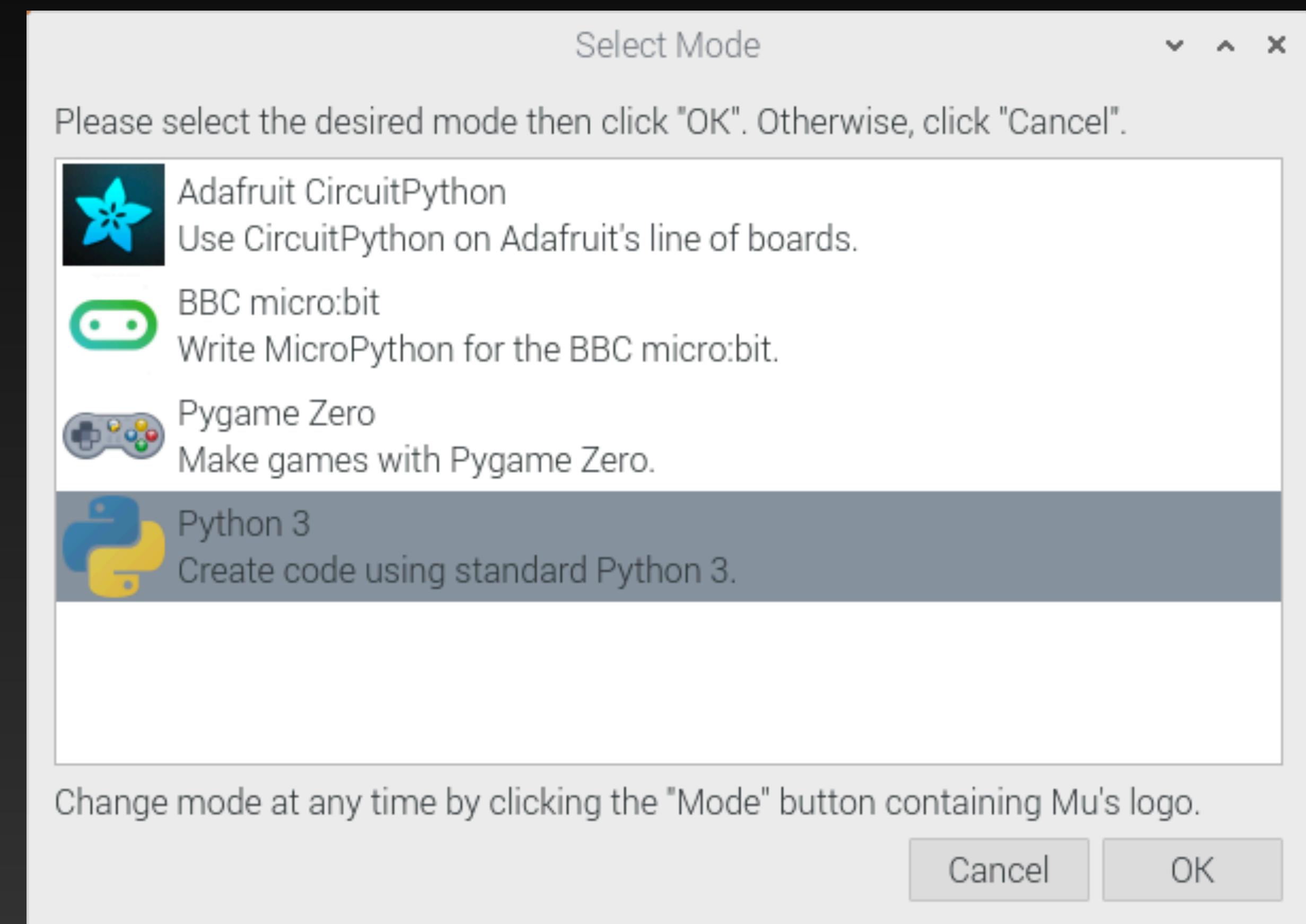
Go to the Programming menu and click on Mu.



Test your circuit! GPIO Zero library

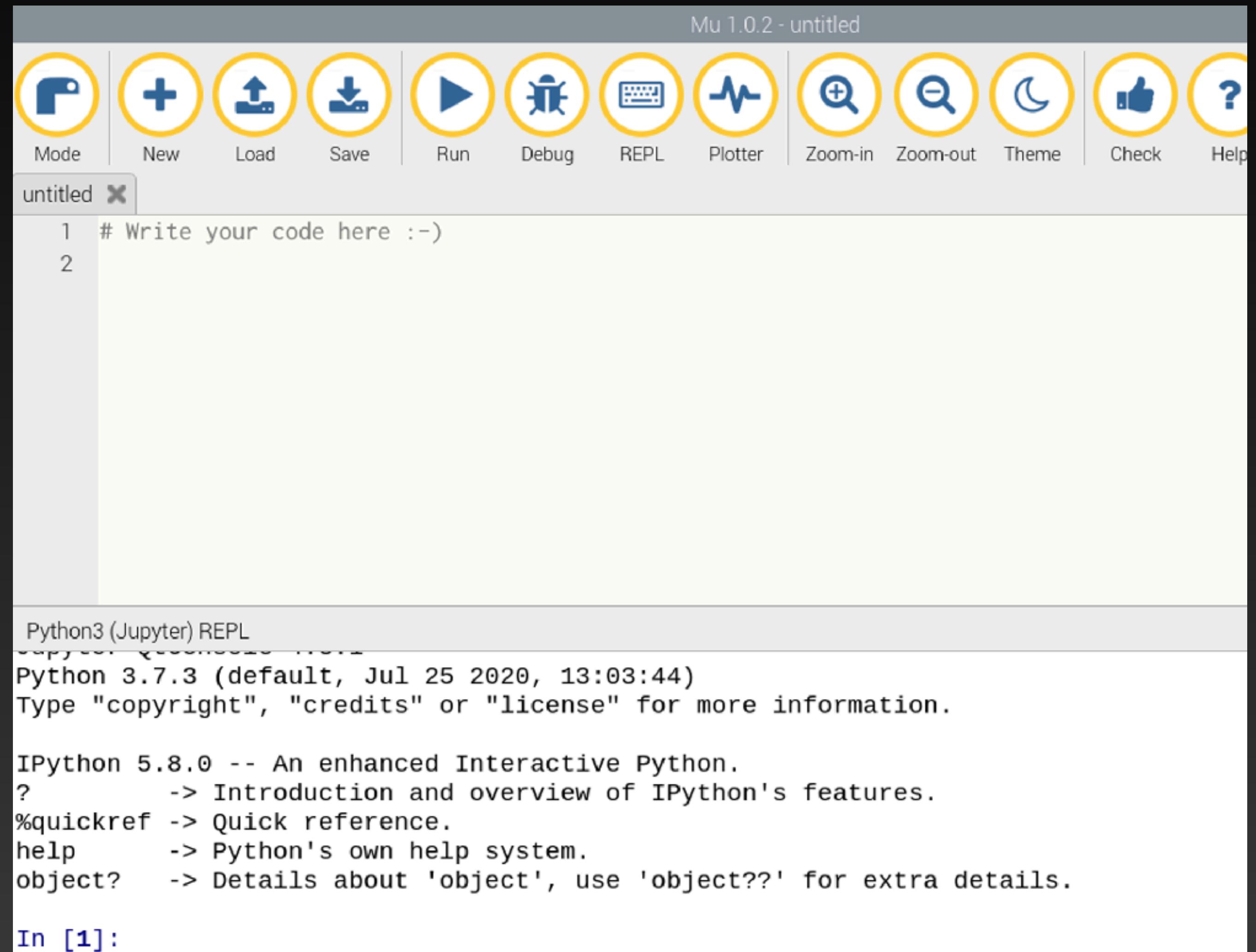
Then choose the mode in which you want to use Mu.

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.



The screenshot shows the Mu code editor interface. The title bar reads "Mu 1.0.2 - untitled". The toolbar contains icons for Mode, New, Load, Save, Run, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, and Help. A code editor window titled "untitled" contains the following Python code:

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

Below the code editor is a terminal window titled "Python3 (Jupyter) REPL" showing the following output:

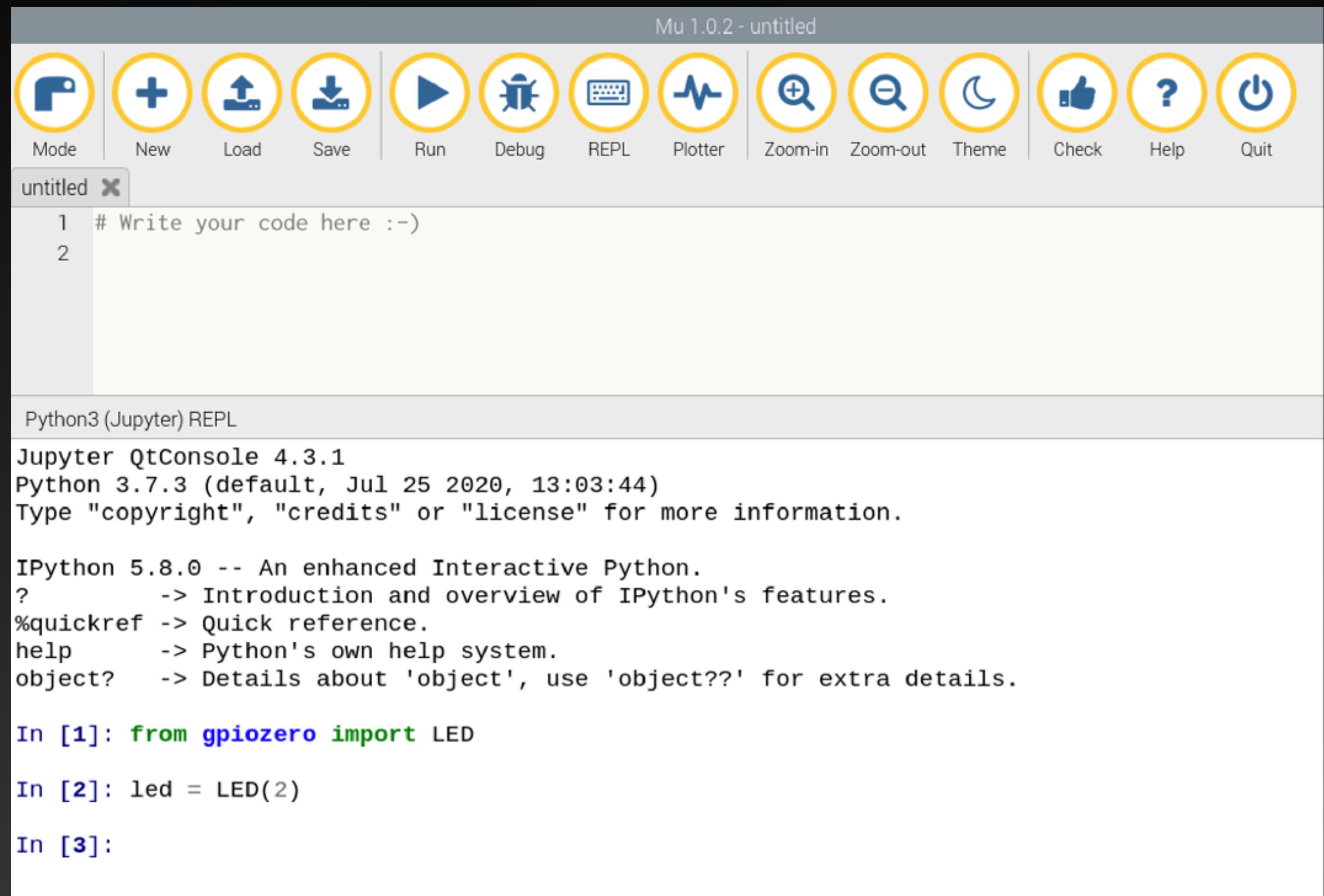
```
Python 3.7.3 (default, Jul 25 2020, 13:03:44)
Type "copyright", "credits" or "license" for more information.

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]:
```

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 1.0.2 IDE interface. The toolbar at the top includes icons for Mode, New, Load, Save, Run, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. The main code editor window has a tab labeled "untitled" with the file extension ".py". The code in the editor is:

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

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

```
Python3 (Jupyter) REPL
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

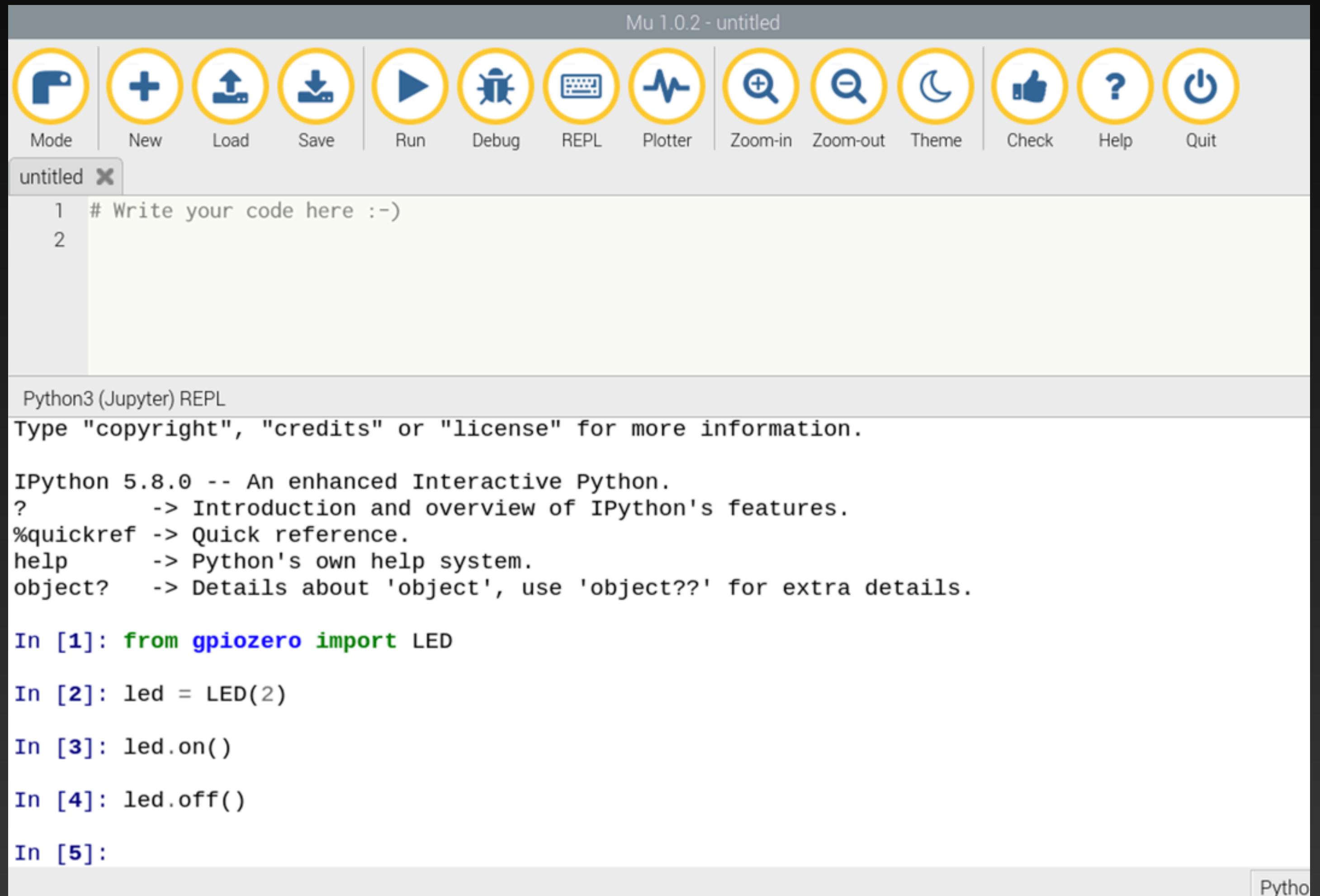
To make the LED switch on, type the following and press Enter:

```
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 1.0.2 IDE interface. The toolbar at the top has icons for Mode, New, Load, Save, Run, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. The main window has a title bar "Mu 1.0.2 - untitled" and a status bar "untitled X". The code editor contains the following Python code:

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

Below the code editor is a Python3 (Jupyter) REPL window. It displays the IPython help menu and a series of Jupyter cell inputs (In [1] through In [5]) showing the import of the GPIOZero library and the creation and control of an LED object.

```
Python3 (Jupyter) REPL
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.

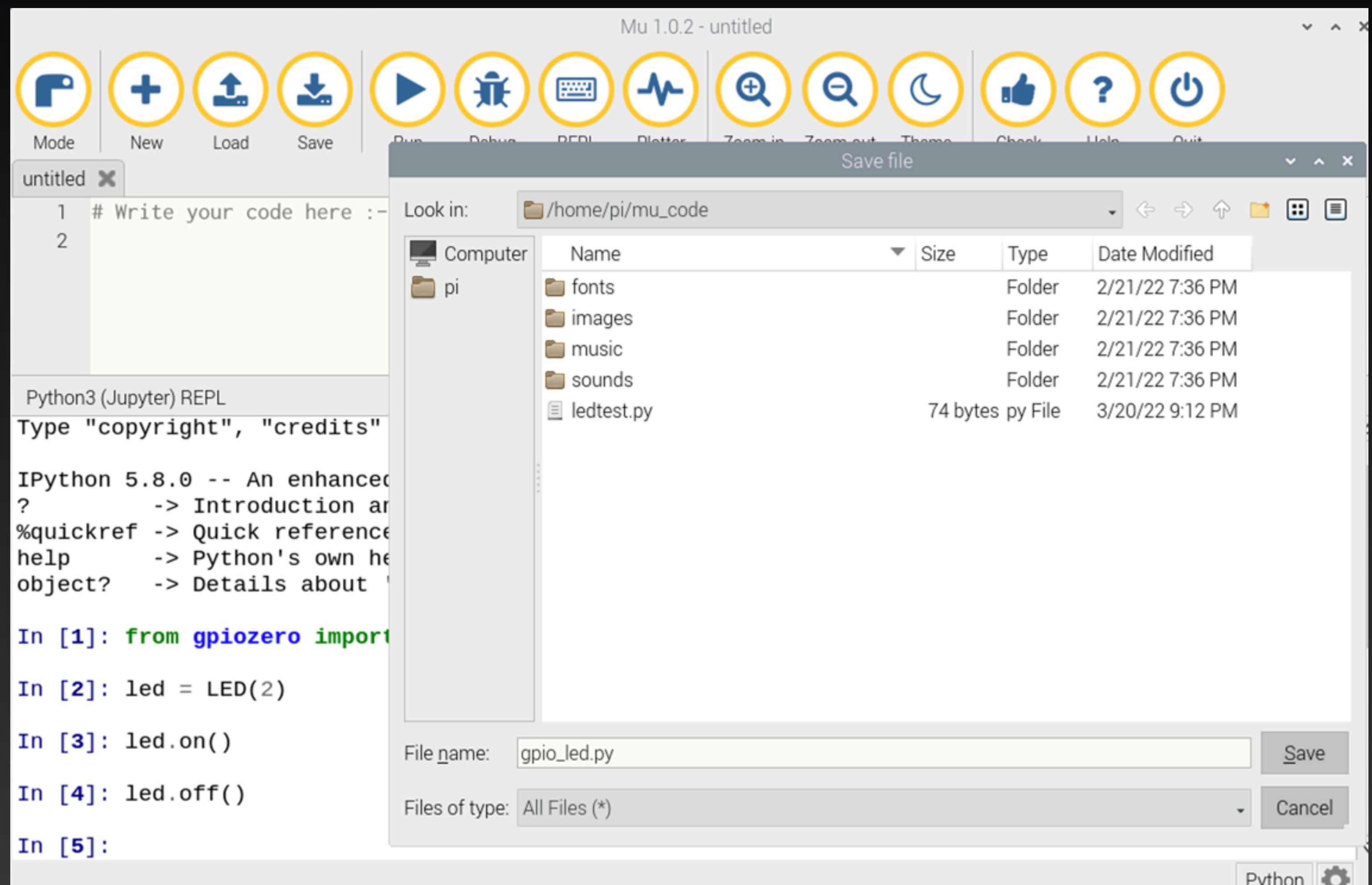
In [1]: from gpiozero import LED
In [2]: led = LED(2)
In [3]: led.on()
In [4]: led.off()
In [5]:
```

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 1.0.2 Python IDE interface. The top bar displays the title "Mu 1.0.2 - gpio_led.py". The toolbar contains icons for Mode, New, Load, Save, Stop, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. The main code editor window is titled "gpio_led.py" and contains the following Python 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 code editor is a "Python3 (Jupyter) REPL" window. It shows the command history:

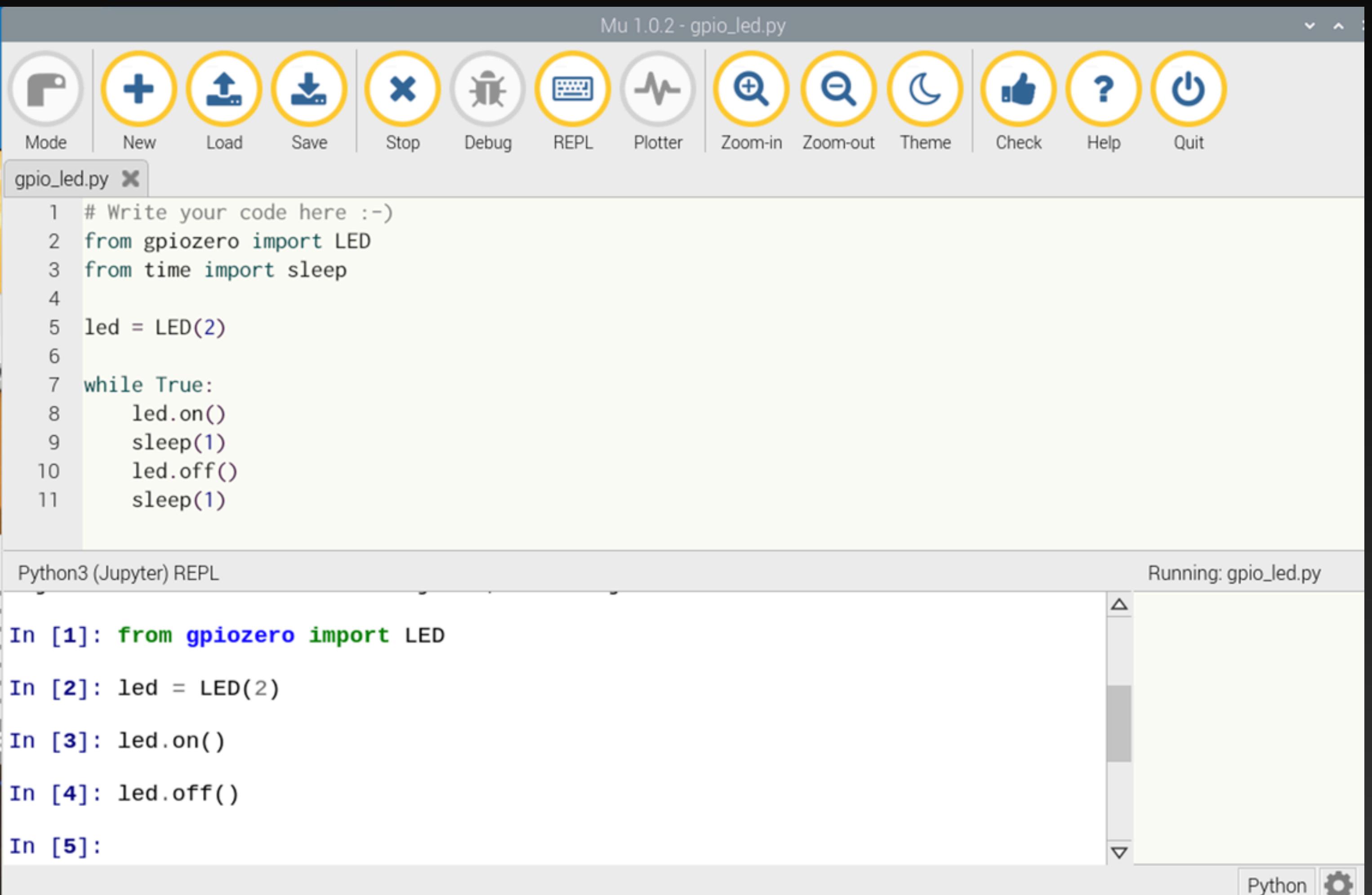
```
In [1]: from gpiozero import LED  
In [2]: led = LED(2)  
In [3]: led.on()  
In [4]: led.off()  
In [5]:
```

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

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 1.0.2 Python IDE interface. The title bar reads "Mu 1.0.2 - gpio_led.py". The toolbar contains icons for Mode, New, Load, Save, Stop, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, Help, and Quit. The main code editor window displays the following Python 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 code editor is a "Python3 (Jupyter) REPL" window with the status "Running: gpio_led.py". It shows the following command history:

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

At the bottom right of the REPL window are buttons for "Python" and "Settings".

IoT with MIT App Inventor

Office hour