CamJam EduKit 2 - Sensors - Introduction

Project Setting up your Raspberry Pi

Description Set up your Raspberry Pi and run your first python program to print "Hello World" to the screen. You will not be connecting any of the contents of the CamJam EduKit to the Raspberry Pi for this short exercise.

The CamJam EduKit 2 Worksheets

The CamJam EduKit 2 – Sensors is the second kit in the CamJam EduKit series. You can download the worksheets, for free, at http://camjam.me/edukit.

The EduKits are compatible with all flavours of the Raspberry Pi.

Notes and Assumptions

Throughout these worksheets, it is assumed that you are using the latest Raspberry Pi OS, and you are editing your code with Thonny (preinstalled with Raspberry Pi OS).

The latest version of Python 3 requires a 'virtual' python environment (venv) to be set up on your Pi so that you can install one of the libraries. You will be taken through the simple steps of creation of the 'venv' in this worksheet.

In this version of the worksheets, you will be using the GPIO Zero Python Library to interact with the Raspberry Pi's GPIO pins.

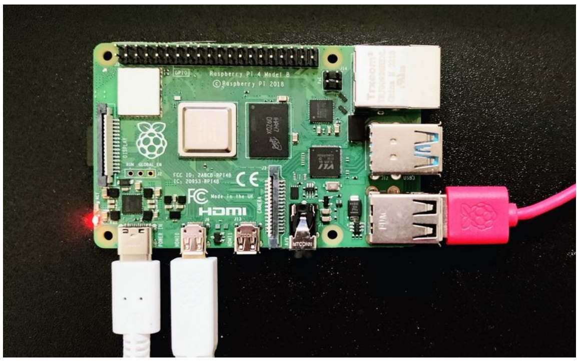
If you are viewing these worksheets on your Raspberry Pi, you should not copy and paste the code from the worksheets as the spacing at the beginning of lines will not necessarily be pasted correctly and the code will not always work; the indentation at the beginning of Python code is important – it is Python's way of recognising how code should be grouped in 'conditions', 'loops' and 'functions'. Instead, type it in or download the code from GitHub with the instructions at the end of this worksheet.

Equipment Required

For this EduKit, you will require:

- A Raspberry Pi
- An SD card to fit your version of the Pi (16GB recommended) with Raspberry Pi OS with Desktop operating system on it. Instructions for setting up Raspberry Pi OS can be found on the Raspberry Pi website (https://www.raspberrypi.org/downloads/).
- Monitor & cable to connect to the HDMI or composite output of your Pi
- A keyboard and mouse
- A Raspberry Pi power supply
- The EduKit 2 kit, available from The Pi Hut (https://thepihut.com/edukit)

Setting up your Raspberry Pi



Find your Raspberry Pi.

- Plug in the microSD card (or SD on the original Models A and B).
- Plug the HDMI/video cable into the Pi and the monitor.
- Plug the keyboard and mouse into the USB ports. You will need an adaptor if you are using a Pi Zero, and a USB hub if you only have one USB port available.
- Plug in the power cable.

When all wired up it should look like the one above, depending on the Raspberry Pi you are using.

Updating Raspberry Pi OS

It is good to keep your Raspberry Pi's operating system up to date with the latest fixes and improvements. You can only do this if your Raspberry Pi is **connected to the internet**. It may take some time (perhaps up to an hour), so you should only do this when you have time.

To update Raspberry Pi OS, open a terminal by either clicking on the icon on the menu bar that looks like this:

>_

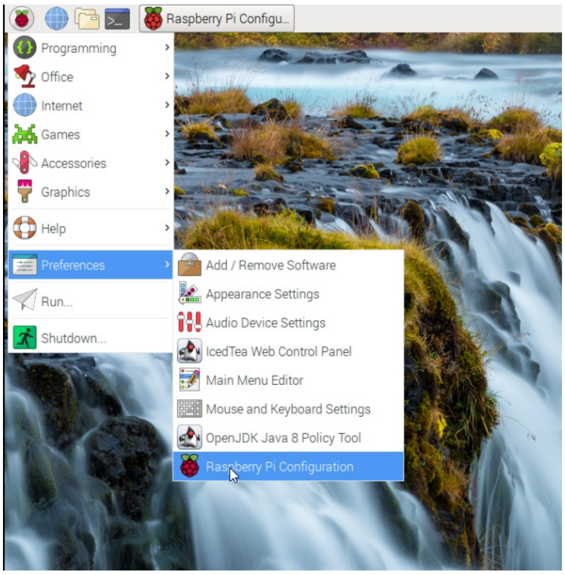
Or choosing Accessories → Terminal on the menu bar at the top.

Type in the two commands below, one after the other, leaving each command to complete before starting the next.

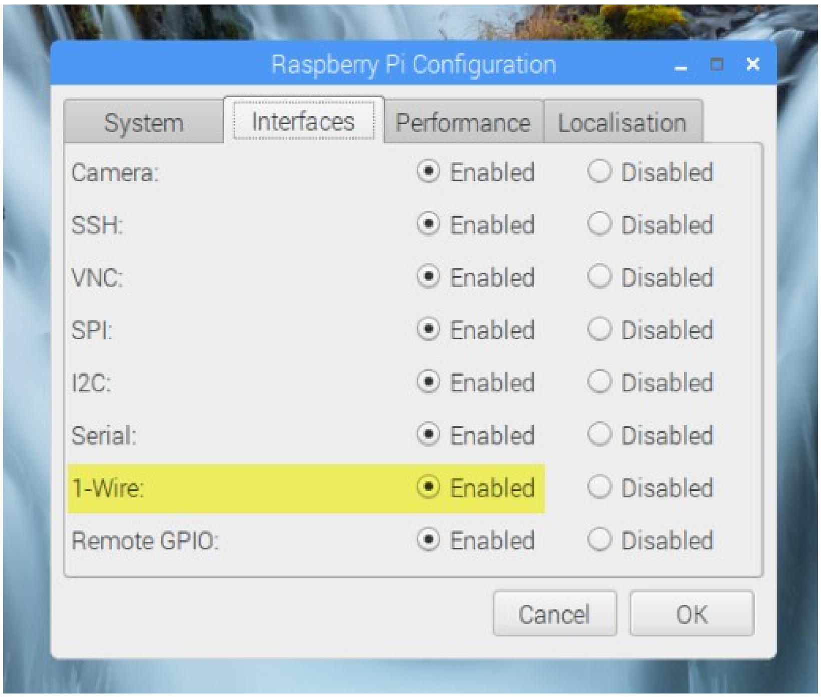
```
sudo apt update
sudo apt upgrade
```

Setting up your Raspberry Pi for CamJam EduKit 2

Before you can use the temperature sensor in worksheet 3, you need to turn on the 1-Wire protocol. To do this, open the Raspberry Pi Configuration tool from the Preferences menu.



In the Raspberry Pi Configuration tool, go to the ‘interfaces’ tab, and enable the 1-Wire interface, as shown below. Click on the OK button, and reboot your Raspberry Pi.



Camera: ● Enabled ○ Disabled

SSH: ● Enabled ○ Disabled

VNC: ● Enabled ○ Disabled

SPI: ● Enabled ○ Disabled

I2C: ● Enabled ○ Disabled

Serial: ● Enabled ○ Disabled

1-Wire: ● Enabled ○ Disabled

Remote GPIO: ● Enabled ○ Disabled

Installing the Temperature Sensor Python Library

NOTE: Installing of PIP Python libraries requires a virtual environment to be created.

Creating a Virtual Environment

Before you can download and install the PIP Python library, you have to create a ‘virtual environment’, usually referenced as ‘venv’. This is a separate area in which you can install isolated copies of libraries, meaning that the libraries are only available in that ‘venv’ and not in any other ‘venv’s you have on your Pi. You can create as many venv’s on your Pi as you want, and install different libraries in each. In fact, you can even install different versions of the same library in different venv’s.

This may sound complex, but is actually easy to create and manage.

Lets create your virtual environment. Open a terminal by either clicking on the icon on the menu bar that looks like this:

Or choosing Accessories → Terminal on the menu bar at the top. Type the following:

```
python3 -m venv ~/EduKit2venv
```

What this command is doing is creating a new virtual environment (using the Python module ‘venv’) in a folder called ‘EduKit2venv’, located under your ‘home’ folder. However, if you run Python 3 now, you will still not be using the venv. To ‘activate’ it so that it is your current Python 3 environment, run:

```
source ~/EduKit2venv/bin/activate
```

You will now see (EduKit2venv) preceding your username and Pi hostname on the terminal prompt line.

You will save your code into this EduKit2venv folder.

Now, all you need to do when running your code within this venv is to remember to activate it using (if the venv name does not appear at the beginning of the command prompt line):

```
source ~/EduKit2venv/bin/activate
```

Install the Library

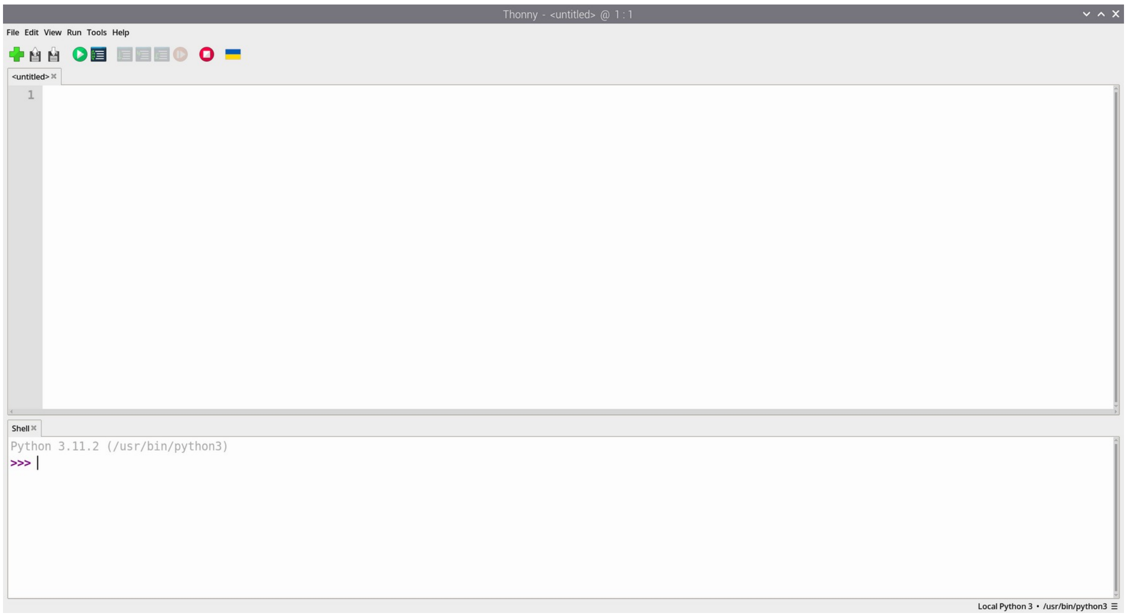
You will need to install an additional Python library, called W1ThermSensor, used in worksheet 3. **Make sure your Pi is connected to the internet**, open a terminal window and type the following:

```
python3 -m pip install w1thermsensor
```

The ‘w1thermsensor’ Python library will now only be available when running code from within that venv.

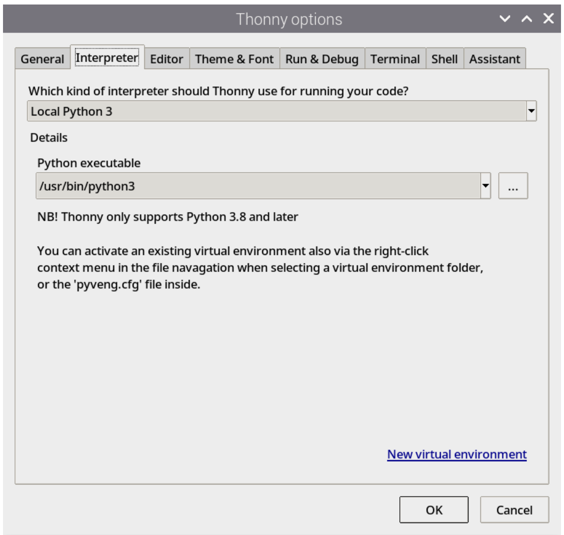
Configuring Thonny to use the Venv

By default, Thonny will use the Raspberry Pi-wide Python environment, and not the virtual one just created. To change to the venv, open ‘Thonny’ from the Menu (under Programming). You should see something similar to below:


Thonny - <untitled> @ 1 : 1
File Edit View Run Tools Help
<untitled>
Shell
Local Python 3 • /usr/bin/python3

```
Python 3.11.2 (/usr/bin/python3)
>>> 
```

At the bottom right of the Thonny window you will see the words ‘Local Python 3 - /usr/bin/python’. This is the Raspberry Pi-wide environment. Left click on the text, then select ‘Configure Interpreter…’. The Options window will be open:



Which kind of interpreter should Thonny use for running your code?

Local Python 3

Details

Python executable

/usr/bin/python3

NB! Thonny only supports Python 3.8 and later

You can activate an existing virtual environment also via the right-click context menu in the file navagation when selecting a virtual environment folder, or the 'pyveng.cfg' file inside.

Select the three dots next to the Python Executable field, and navigate to the venv you created earlier, which should be located at /home/pi/EduKit2venv/bin/python. Then click OK.

Now the code that Thonny runs will be run using your Python 3 venv.

Writing Code

You are now going to create your first small piece of Python code that will simply print “Hello World” to the screen.

Open ‘Thonny’ from the Menu (under Programming), and create a file using the ‘New’ in the File menu (or use Ctrl+N).

Type in the following code exactly as seen into your preferred editor:

```
# Print Hello World!
print("Hello World!")
```

Everything on the same line after a ‘#’ is a comment and will be ignored by Python.

Save the file in the EduKit2venv folder created above, calling the file `1-helloworld.py`.

Running the Code

To run your code, select the menu option Run → Run Module, or press F5.

You will see “Hello World!” printed to the Python Shell.

Downloading the EduKit Code from GitHub

The code written for the CamJam EduKit, and listed in the worksheets, is also stored on GitHub. Follow these instructions to download all the EduKit code.

GitHub Repository

All the repositories for each EduKit can be found online at https://github.com/CamJam-EduKit. You may download individual files or whole repositories from there.

Installing Git

Before you can ‘clone’ the code from GitHub, you must first ensure that the Git tool is installed on your Raspberry Pi. To do this your Raspberry Pi must be connected to the internet. Open a terminal window and first update the Pi repository using:

```
sudo apt update
```

Then install GIT using:

```
sudo apt install git-core
```

Downloading to the Raspberry Pi

The EduKit code and worksheets can be downloaded using the following command:

```
cd ~
git clone https://github.com/CamJam-EduKit/EduKit2.git
```

The code will be stored in the `Code` subfolder under the `EduKit2/CamJam Edukit 2 – GPIO Zero` folder.