

Raspberry PI + Python Lab

Goal: To get acquainted with the PI, how it is built, useful ports; produce simple python code.

Outcome: Deploy a simple piece of python code into the PI.

Part I

Have a look at chapters 1 and 2 from the official Raspberry PI beginner's guide. Thereafter, have a look at the paragraph *Raspberry Pi - Linux Shell* from

https://www.tutorialspoint.com/raspberry_pi/raspberry_pi_introduction.htm.

At this point you should have a general sense of what the PI is. You can connect a raspberry PI to a monitor through, e.g., an HDMI cable, as well as a mouse and keyboard. You can try this, and see that it is just like any other computer, almost. Alternatively, and a more practical option, you can connect to the PI through wifi (some models), or an ethernet cable.

For the purposes of this tutorial, you can simply connect to the PI through the cable, and login to the PI from your own computer through **ssh**. The PI has already been configured such that ssh is enabled, and its static IP is set to 192.168.2.10, with network mask 255.255.255.0. On your side, depending on whether you are on Mac/Win/Linux, you need to setup a static IP equal to **192.168.2.X**, e.g. **192.168.2.2**. You can **ssh** from the command prompt (command line/terminal).

Setting up a static IP

This will look a bit differently depending on which OS you're on.

MacOS:

Go to system preferences, and select Network. Identify the connection, click on it, next to Configure ipv4, select manually from the drop-down menu. Fill in the ip address, and subnet mask. Click apply.

Linux:

Find the Network preferences, select the wired connection, and click on settings (the gear shaped icon). You will then see a dialog box, go to the ipv4 tab, in the ipv4 method select manual. Fill in the ip address and mask.

Windows:

Go to control panel, then Network and Internet, then Network and sharing center, right click on the ethernet connection (located somewhere on the right side), and then properties. Thereafter, select the Internet Protocol Version 4 (TCP/IPv4), and click properties. Assign the IP and mask. Click ok. Congrats, you're now done. (<https://pureinfotech.com/set-static-ip-address-windows-10/>)

SSH-ing into the PI

Let's have a quick look at how you can login to the PI from each OS. You will need this to check the outcome of your assignment at the end of the tutorial.

MacOS/Linux/Windows:

Open a terminal and type the following (powershell on windows):

```
$ ssh ubuntu@192.168.2.10
```

Where ubuntu is the username on the pi. You will be prompted for a password, type: *turtlebot*

Now you can type commands to the shell of the PI from your own laptop. You can try again the simple commands from the *Raspberry Pi - Linux Shell*.

Using scp to transfer files back and forth

There is a simple command that can be used to transfer a file from your laptop to the PI and vice versa.

```
$ scp file.txt ubuntu@192.168.2.10:/remote/directory
```

Where the file.txt is on your laptop, and you'd like to transfer to /remote/directory on the PI.

You will use this command to transfer the code you will write to the PI when you want to test.

Additional Reading

Read the following titles from the tutorials for absolute beginners at https://www.tutorialspoint.com/raspberry_pi/raspberry_pi_introduction.htm:

1. Raspberry Pi – Introduction
 - a. Intro text + Generations and Models paragraphs
 - b. History paragraph
 - c. For interested ones: Speed + Connectivity Specifications
2. Raspberry Pi - Getting Started
 - a. Retailers and Distributors can be skipped
3. Connecting Raspberry Pi
 - a. Checkout the diagram for our model
 - b. Insert SD or MicroSD card paragraph
 - c. Connect Raspberry Pi to Devices paragraph
4. Raspbian configuration
 - a. Configuration tool (pass, hostname, boot, auto login, network at boot) paragraph

Part II

Now, together with the knowledge from the python tutorial go ahead with two exercises:

1. Write a small python code that creates an empty file in a folder of your choosing. Afterwards:
 - a. Research on how to make a script or piece of code run at boot time.
 - b. Transfer your code to the PI, and make it such that it runs at boot time.
 - c. Check your result on the PI after a reboot.
2. Replicate the example from the PI slides for switching the led on/off.