Lesson 1 –
Setting up the Python programming environment in a single board computer

- S.P. Chong

Objectives

- In this lesson, you will be given an overview of the module – the **topics** to be covered, the **assessments** and the **schedule**.

- You will then learn to **set up** the Python **programming environment** in a Raspberry Pi, a single board computer (*):
  - You will learn the **key features** of Raspberry Pi.
  - You will learn what **hardware** will be required to use a Raspberry Pi.
  - You will learn how to use the **IDE** (Integrated Development Environment).

(*) The lecturer will brief you and demonstrate the key steps.

The Topics & schedule

Term 1

- Lesson 1 - Raspberry Pi set up (& headless mode)
- Lesson 2 - Python basics (+ Python OOP)
- Lesson 3 - Physical computing
- Lesson 4 - Sensor data collection, storage, visualization & analytics
- Lesson 5 - Sensor data upload to cloud & notification
- Quiz (30% before MST)

Term 2 GP (10%)

- Lesson 6 - A simple web server
- Lesson 7 - GUI (Graphical User Interface)
- Mini Project (20% CD after vacation, 40% implementation before exam)

The Assessments

Online Quiz (30%)

- Coverage: Lessons 1 to 5.
- Format: 75 MCQ's, closed book, using Lock-down Browser.
- Duration: 1.5 hours.
- Week: 6.

Mini Project (20% CD + 40% Implementation)

- Requirements & criteria for marking: refer to "Mini Project Specifications".
- Key points: use Python on Raspberry Pi, to implement an IoT application.
- Work in pair.
- Present idea for critique in 1st week of 2nd term, before implementation.

Internet of Things?

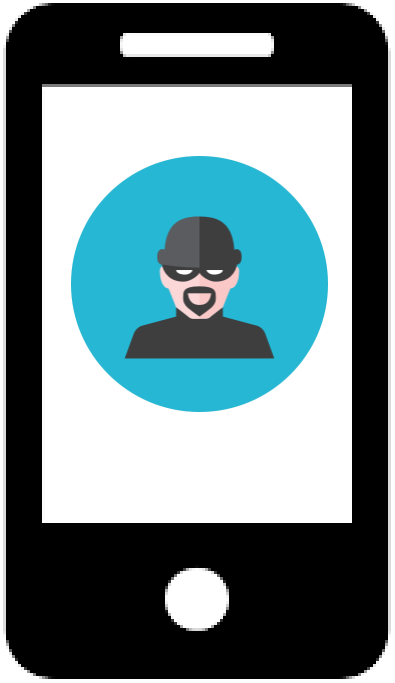
"Things" connected to internet, to exchange data.

IoT helps to improve lives, work & play.

Examples?

Internet of Things? (cont.)

An intruder is detected, when the house owner is away.



A photo is taken, and the house owner is notified.

How can you implement such a project idea?

Internet of Things? (cont.)

This IoT application allows us to travel conveniently from one place to another.

What kind of sensor data is collected, for this to work?

How do the passenger & the driver connect to the internet wirelessly?

Which Cloud platform is Grab residing on? (Hint: Google)

How is the passenger matched to the driver?

How is the passenger charged?

Internet of Things? (cont.)

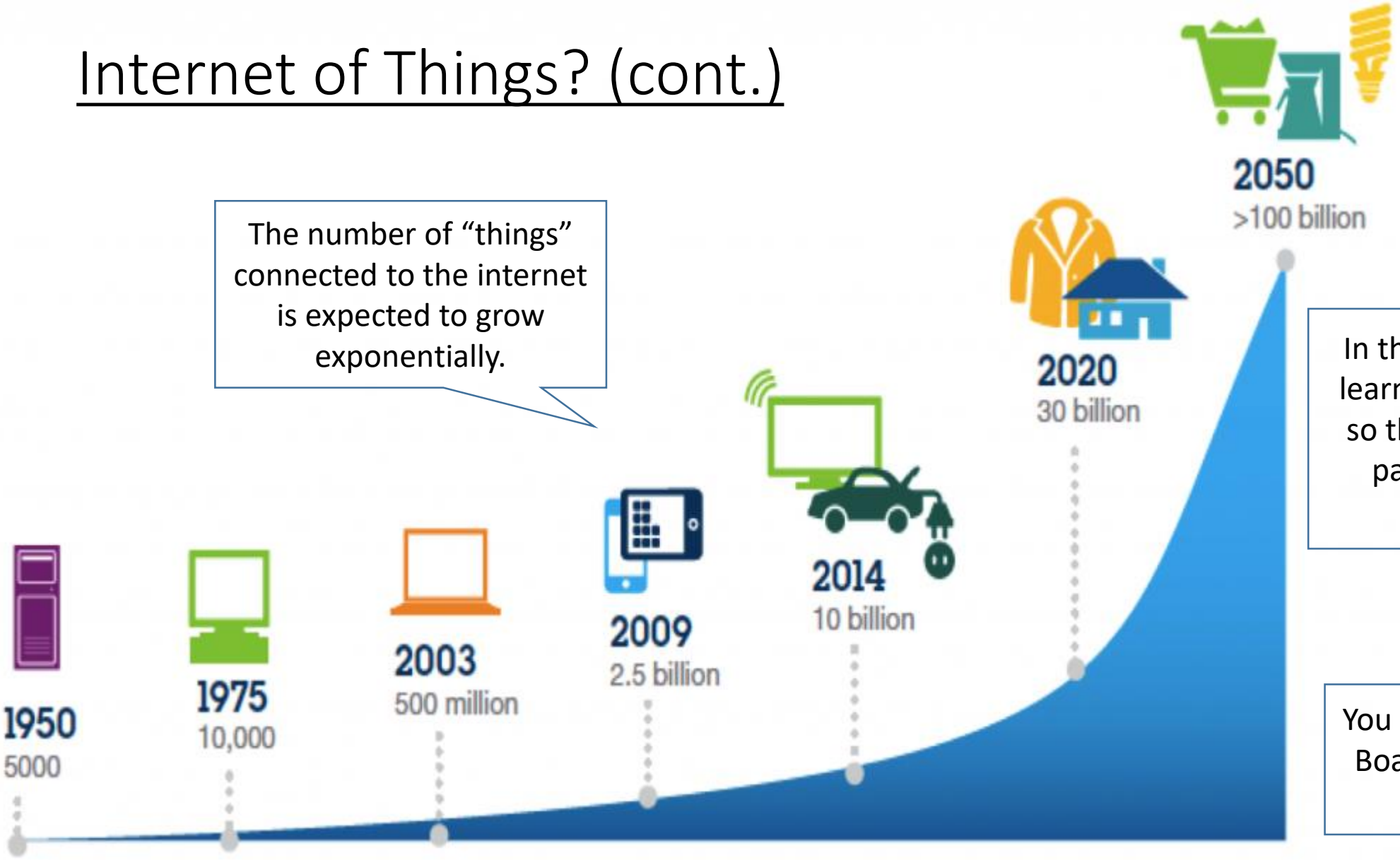
The bin is full.

The trash collector is notified automatically.

Again, how can you implement such a project idea?

Internet of Things? (cont.)

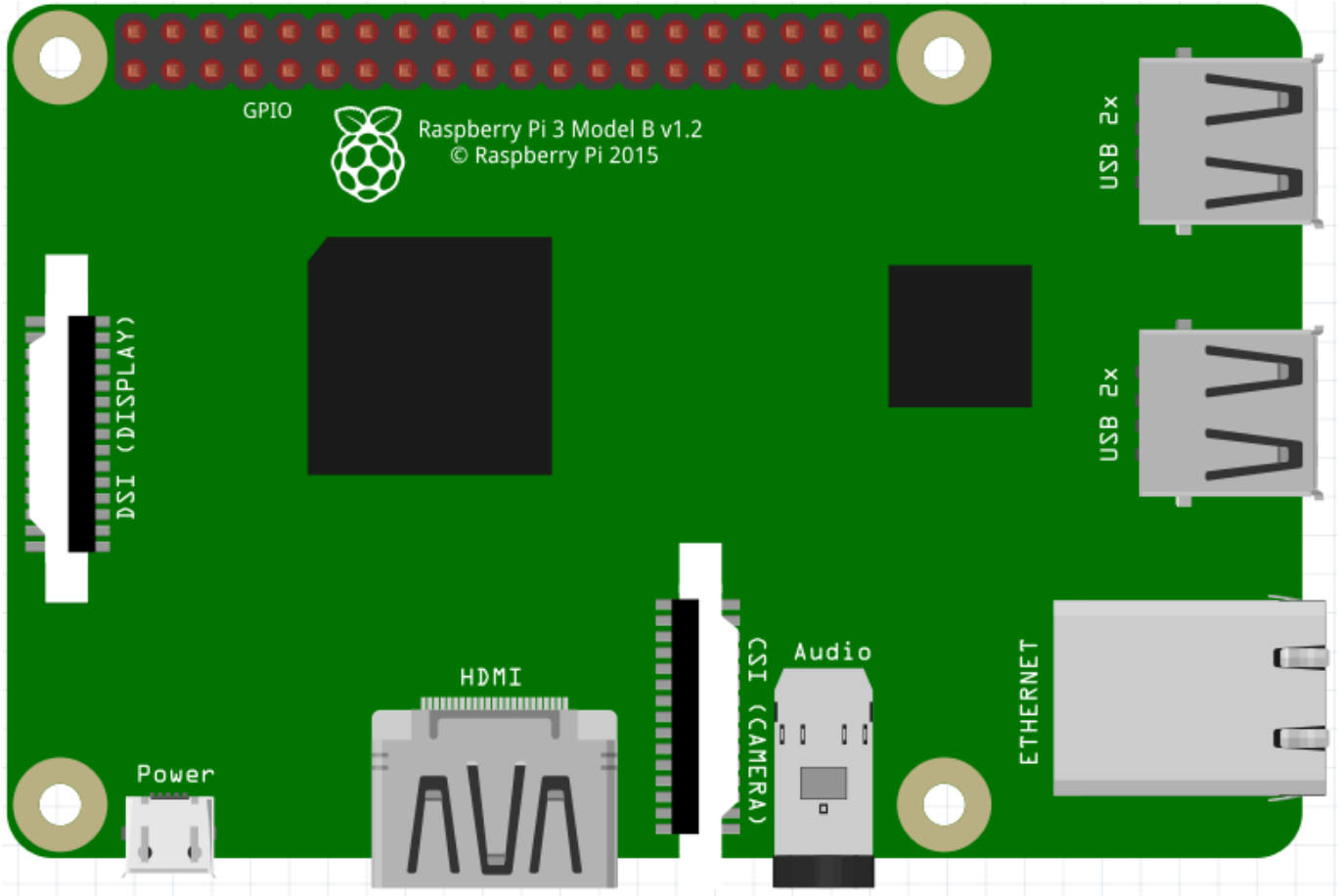
The number of “things” connected to the internet is expected to grow exponentially.



In this elective, you will learn to code in Python, so that you too, can be part of this exciting phenomenon.

You will be using a “Single Board Computer” called Raspberry Pi.

Key features of Raspberry Pi 3 Model B+



1.4GHz 64-bit quad-core processor.

dual-band wireless LAN.

Bluetooth 4.2/BLE.

faster Ethernet.

Power-over-Ethernet support (with separate PoE HAT).

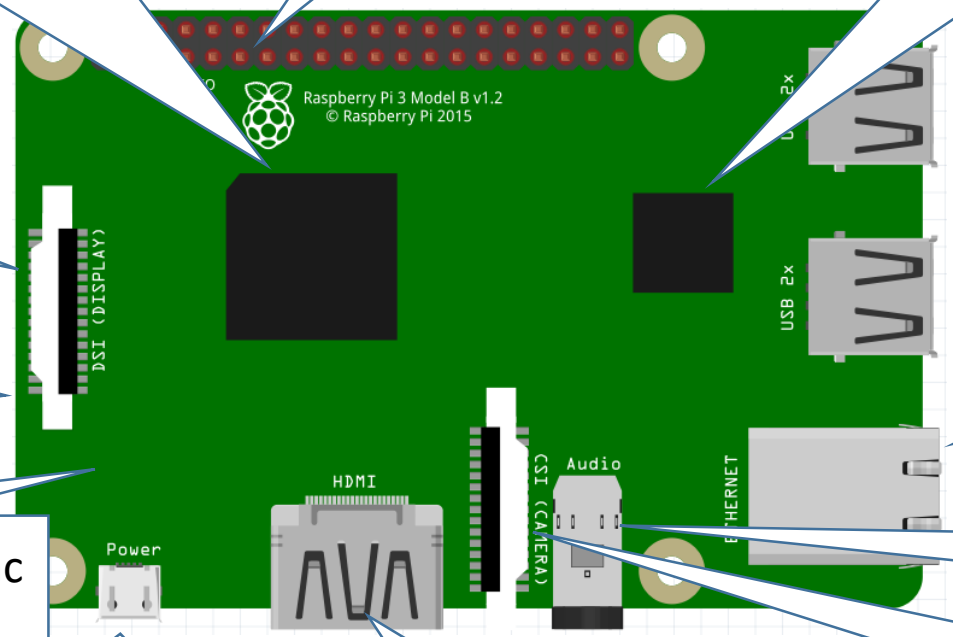
We will not spoon feed you!

You have to find out for yourself what the following terms mean:

What is meant by…

- 64-bit processor
- Quad-core processor
- Dual-band wireless LAN
- BLE
- Ethernet
- PoE HAT

Key features of Raspberry Pi 3 Model B+ (cont.)



Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz.

40-pin GPIO header.

1GB LPDDR2 SDRAM.

4 USB 2.0 ports.

DSI display port for connecting a Raspberry Pi touchscreen display.

Gigabit Ethernet over USB 2.0 (maximum throughput 300 Mbps).

Micro SD port for loading your operating system and storing data.

4-pole stereo output and composite video port.

2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE.

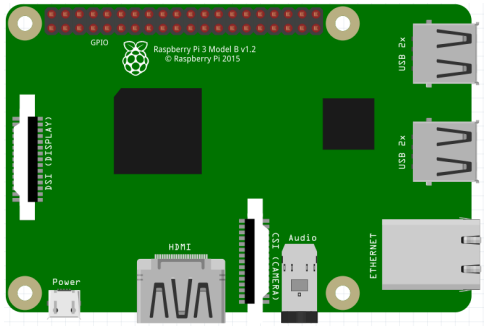
CSI camera port for connecting a Raspberry Pi camera.

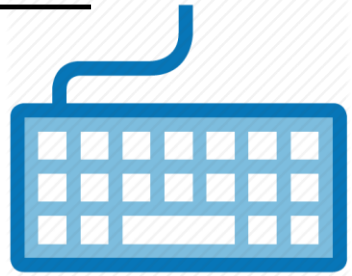
5V/2.5A DC power input.

Full-size HDMI.

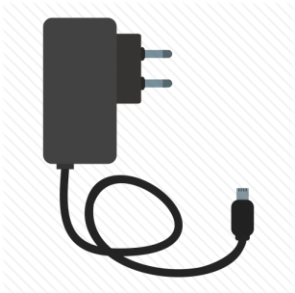
Hardware for using Raspberry Pi

Micro SD card
(16G)





keyboard



5V 2.5A
adaptor

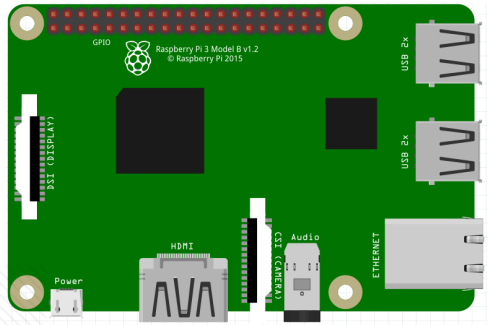
mouse

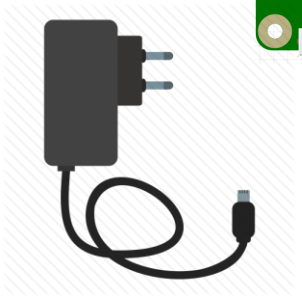
HD monitor
(with HDMI cable)

This is how you use Raspberry Pi as a CPU (Central Processing Unit).

Hardware for using Raspberry Pi (cont.)

Your Micro SD card (16G)





5V 2.5A adaptor

LAN cable

Your laptop

This is how you use a laptop's monitor / mouse / keyboard for the Raspberry Pi.

Hardware for using Raspberry Pi (cont.)

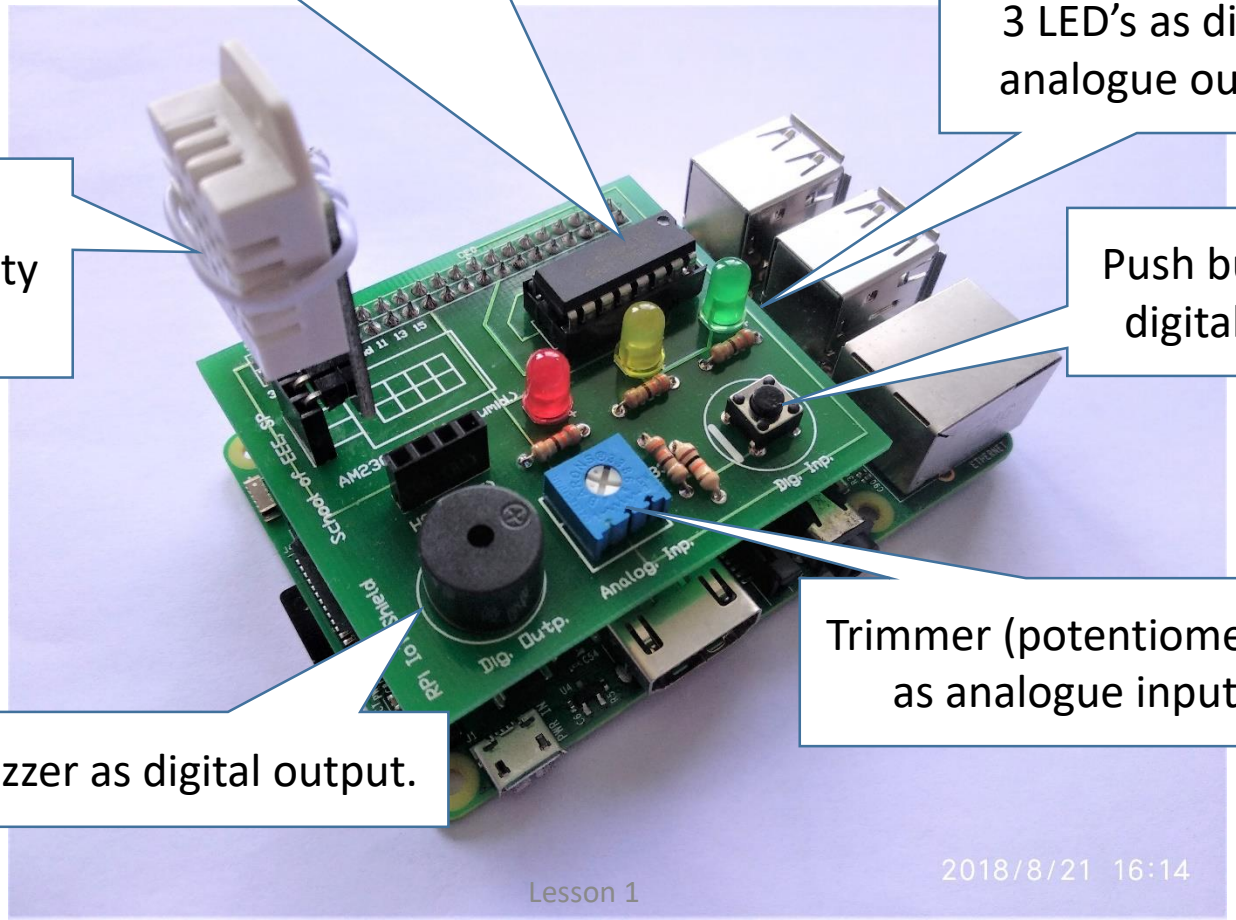
A “RPi IoT shield” can be used with the Raspberry Pi.

MCP3008 ADC (Analog to Digital Converter) with SPI (Serial Peripheral Interface).

3 LED’s as digital / analogue outputs.

AM2302 (DHT22) temperature & humidity sensor.

Push button as digital input.

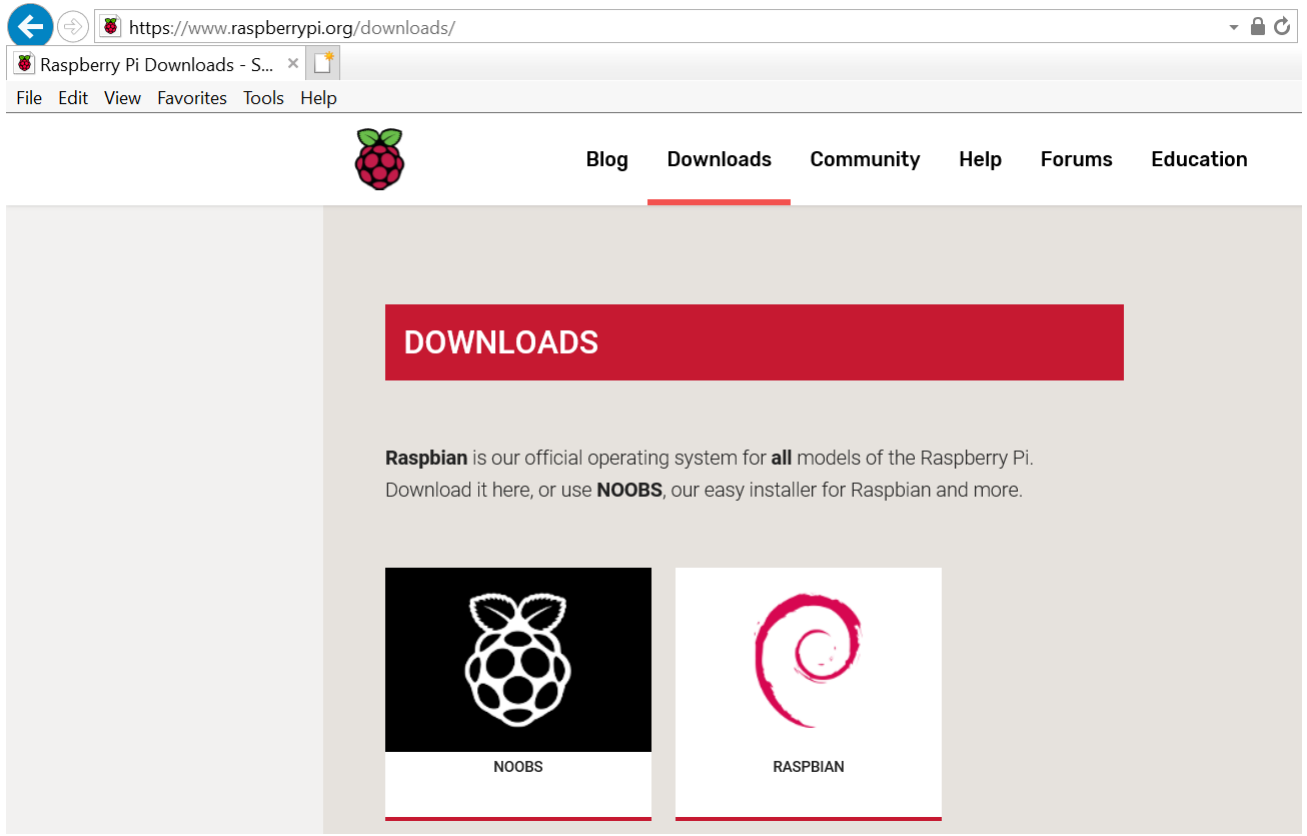


You have to buy your own Raspberry Pi, if you want to connect other IO devices to it.
☺

Trimmer (potentiometer) as analogue input.

Buzzer as digital output.

Setting up the Raspberry Pi OS (Operating System)



The official website for Raspberry Pi is www.raspberrypi.org
Take a look at the resource available here!

The OS can be downloaded here, and installed on a micro SD card.

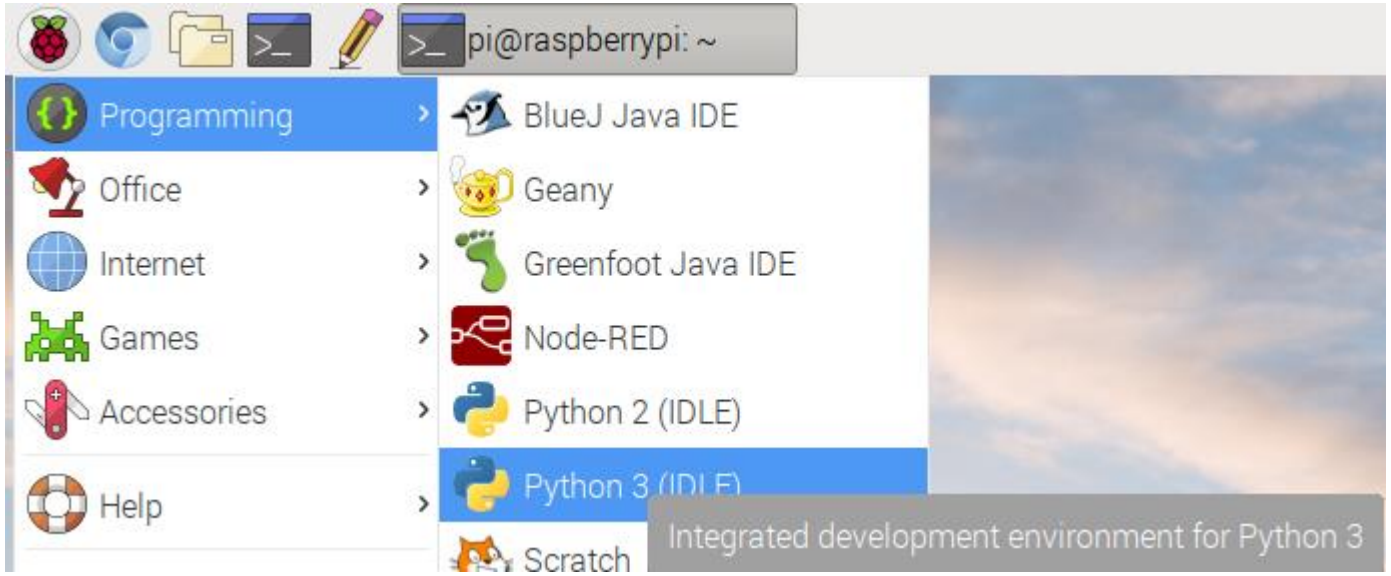
The lecturer can do a demo, or show you a video.
You should do the same when you have your SD card and internet access.

Setting up the Raspberry Pi OS (Operating System) (cont.)

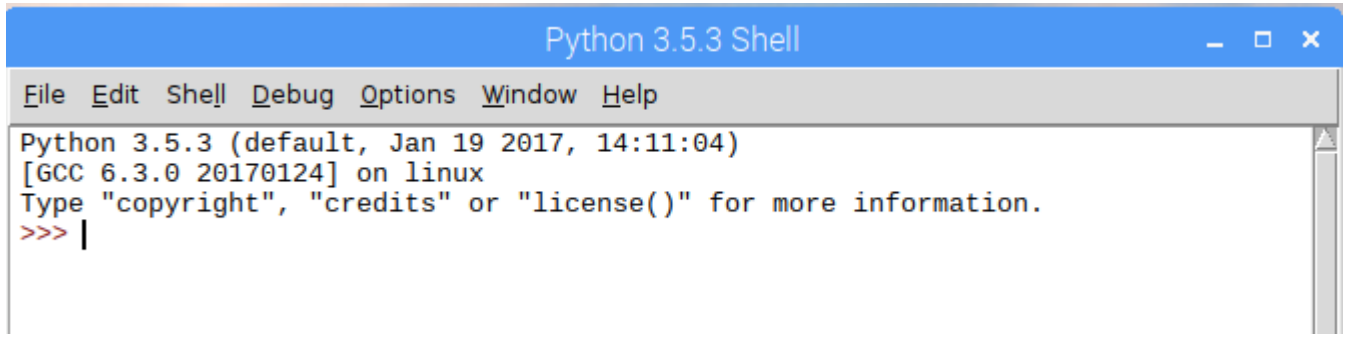
Key steps:

1. Download **NOOBS** (New Out of the Box Software) from the link given.
2. Download **SD Formatter** for Windows (- Google it).
3. Unzip, install the SD Formatter and use it to format the SD card (8G or above).
4. Then unzip the NOOBS into the **SD card**.
5. Insert the SD card into the Raspberry Pi. Connect a **mouse**, a **keyboard**, a **monitor** and **power** up the Raspberry Pi.

Writing & running your first Python program

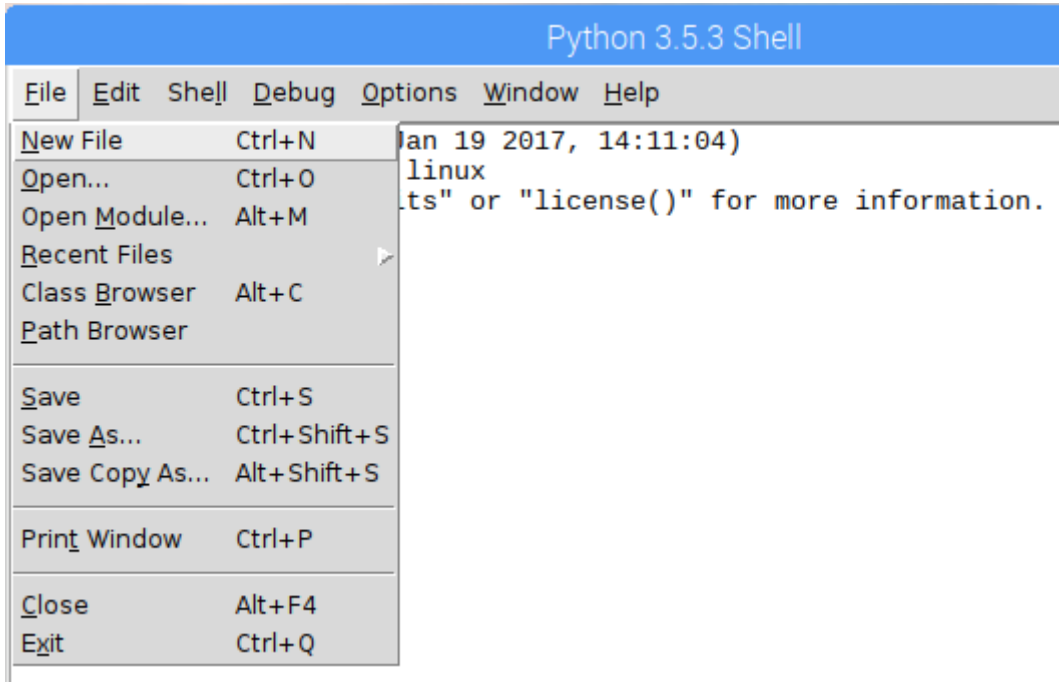


Click Raspberry Pi ⇨ Programming ⇨ Python 3 (IDLE)



In the Python 3.x.y Shell that pops up, you can enter your Python program (line by line, to be interpreted), but…

Writing & running your first Python program (cont.)



We will enter the Python program into a file…
So, click File ⇨ New File

Writing & running your first Python program (cont.)

Type these lines of code.

Untitled

File Edit Format Run Options Window Help

```
import RPi.GPIO as GPIO
from time import sleep
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
GPIO.setup(27,GPIO.OUT)
while(True):
    GPIO.output(27,1)
    print("LED turns on...")
    sleep(1)
    GPIO.output(27,0)
    print("LED turns off...")
    sleep(1)
```

This is a “minimum keystroke” program. Guess what that means?

- The **RPi.GPIO** library (or more correctly called **module**) is imported and named as **GPIO**.
- The **sleep** function is imported from the **time** library.
- …setmode…BCM means we will refer to the GPIO pins by their **GPIO numbers**, rather than their physical pin numbers.
- …setwarnings(False) means if a GPIO pin is **already in use**, there will be no warning messages, if the lines of code refer to the pin.
- GPIO 27 is made an **output pin**, since this is connected to an **LED**.
- while(True): means the following lines will **execute forever**.

Writing & running your first Python program (cont.)

Untitled

File Edit Format Run Options Window Help

```
import RPi.GPIO as GPIO
from time import sleep
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
GPIO.setup(27,GPIO.OUT)
while(True):
    GPIO.output(27,1)
    print("LED turns on...")
    sleep(1)
    GPIO.output(27,0)
    print("LED turns off...")
    sleep(1)
```

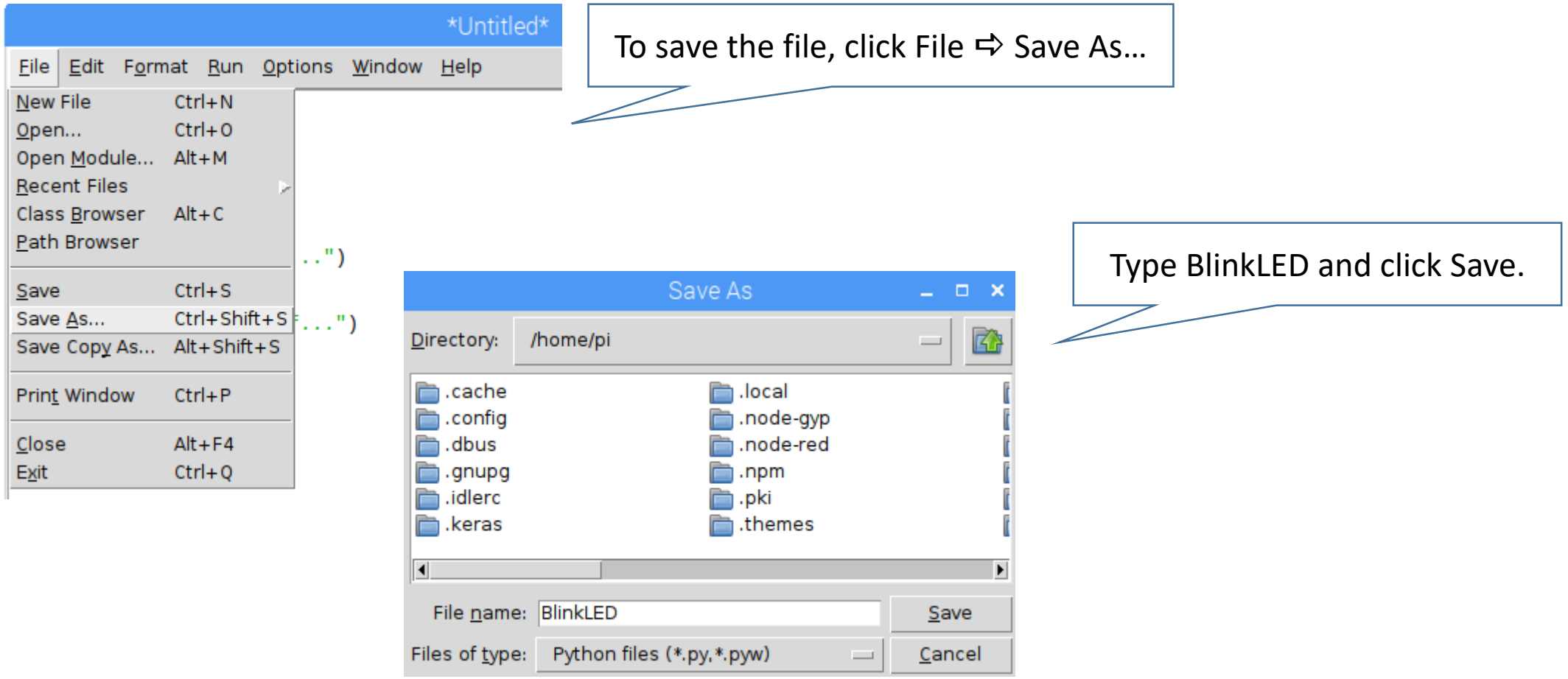
You can also write
while (1): or
while True: or
while 1:

Results? LED blinks on and off…

In the loop:

- The LED is turned **on**.
- A **message** is printed onto the screen.
- A **delay** of 1 second is introduced.
- The LED is turned **off**.
- A message is printed onto the screen.
- A delay of 1 second is introduced.

Writing & running your first Python program (cont.)



Writing & running your first Python program (cont.)

The file name has been changed.

BlinkLED.py - /home/pi/BlinkLED.py (3.5.3)

File Edit Format Run Options Window Help

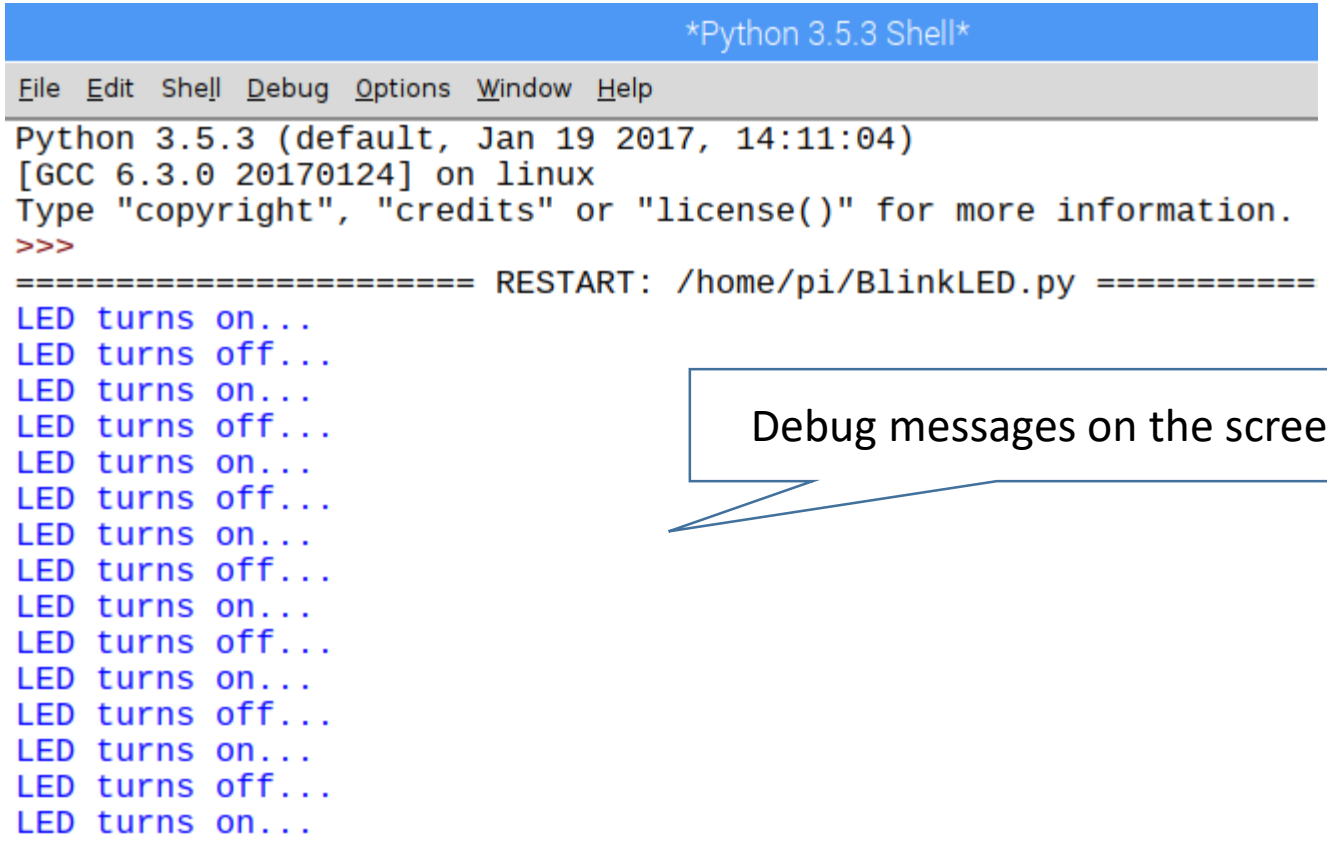
```
import RPi.GPIO as GPIO
from time import sleep
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
GPIO.setup(27,GPIO.OUT)
while(True):
    GPIO.output(27,1)
    print("LED turns on...")
    sleep(1)
    GPIO.output(27,0)
    print("LED turns off...")
    sleep(1)
```

To run the program, press F5.

Writing & running your first Python program (cont.)

Red LED on the IoT shield.

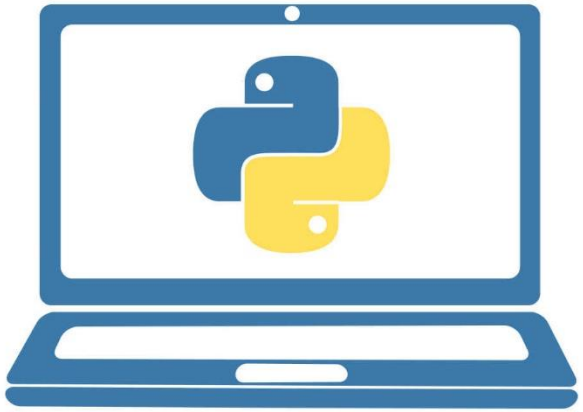
Writing & running your first Python program (cont.)



Debug messages on the screen.

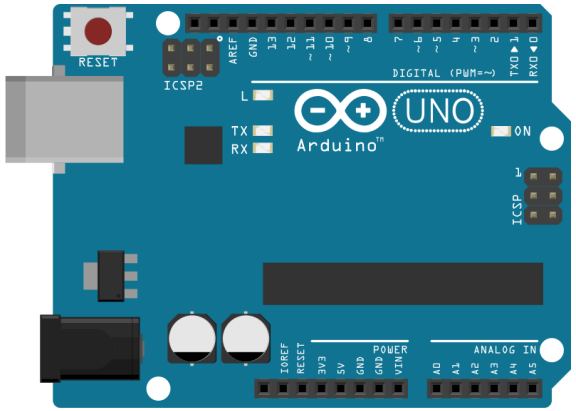
You have just written your first Python program.

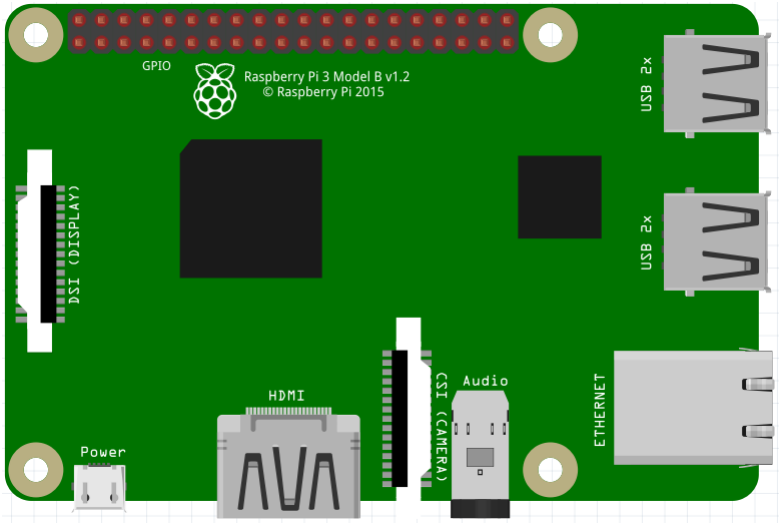
Lab Exercises



- Exercise 1.1 – Understanding some computer terms
- Exercise 1.2 – Finding out more about Raspberry Pi
- Exercise 1.3 – Setting up the Raspberry Pi OS
- Exercise 1.4 – Writing your first Python program

Exercise 1.1 – Understanding some computer terms



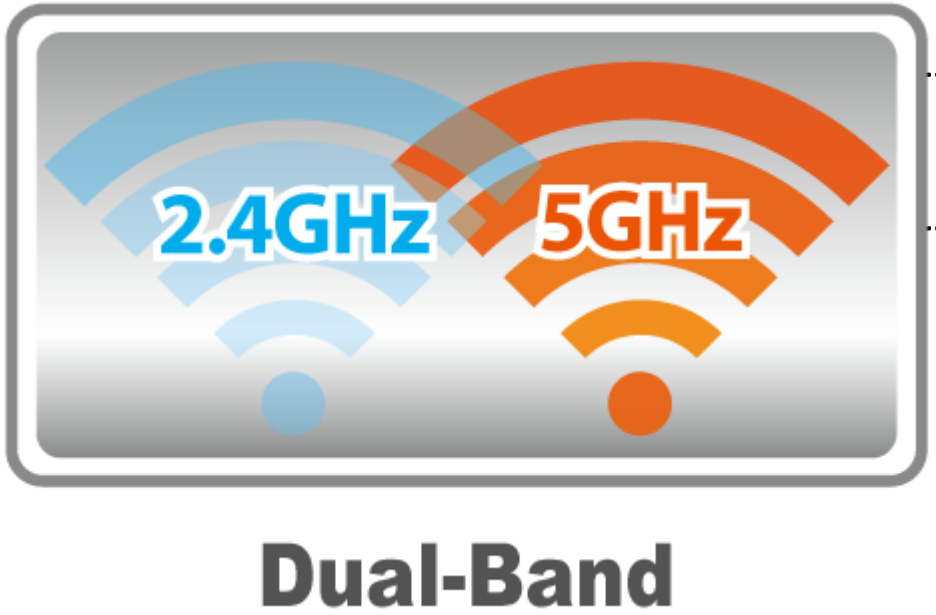


Find out the answers to the following questions:

Q1. What are the main differences between a 8-bit microcontroller running at 16 MHz (such as Arduino UNO), and a quad-core 64-bit processor running at 1.4 GHz (such as Raspberry Pi 3B+)?

Exercise 1.1 – Understanding some computer terms (cont.)

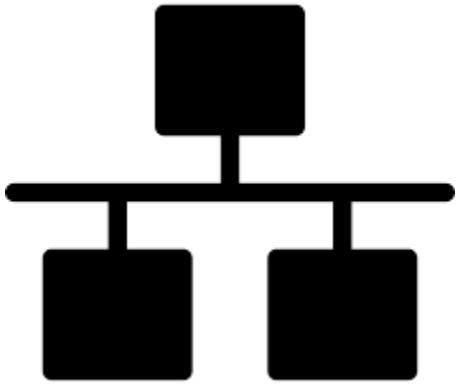
Q2. What is meant by dual-band wireless LAN?

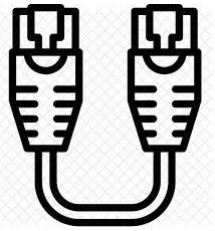


Exercise 1.1 – Understanding some computer terms (cont.)

Q3. What is meant by BLE?

Exercise 1.1 – Understanding some computer terms (cont.)





Q4. What is Ethernet?

Exercise 1.2 – Finding out more about Raspberry Pi

Match these:

Processor ☐

Memory ☐

Connectivity ☐

Physical World Access ☐

Video & sound ☐

Multimedia ☐

SD card support ☐

Input power ☐

Environment ☐

Production lifetime ☐

☐ The Raspberry Pi 3 Model B+ will remain in production until at least January 2023.

☐ Broadcom BCM2837B0, Cortex-A53 64-bit SoC @ 1.4GHz

☐ Extended 40-pin GPIO header

☐ Micro SD format for loading operating system and data storage

☐ 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE, Gigabit Ethernet over USB 2.0 (maximum throughput 300Mbps), 4 × USB 2.0 ports

☐ 1 × full size HDMI, MIPI DSI display port, MIPI CSI camera port, 4 pole stereo output and composite video port

☐ 1GB LPDDR2 SDRAM

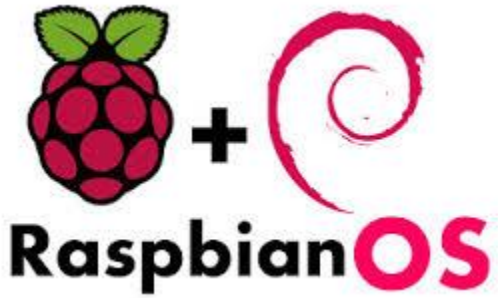
☐ Operating temperature, 0–50°C

☐ 5V/2.5A DC via micro USB connector

☐ H.264, MPEG-4 decode (1080p30); H.264 encode (1080p30); OpenGL ES 1.1, 2.0 graphics

Exercise 1.3 – Setting up the Raspberry Pi OS

By following the demo / instruction given during the lecture, install the Raspbian OS onto an SD card.



After installation, you will be using this SD card, in (almost) every lesson, for the rest of the semester.

So, take good care of this card…

Exercise 1.4 – Writing your first Python program

Try out the BlinkLED.py program. Debug until it works.

BlinkLED.py - /home/pi/BlinkLED.py (3.5.3)

File Edit Format Run Options Window Help

```
import RPi.GPIO as GPIO
from time import sleep
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
GPIO.setup(27,GPIO.OUT)
while(True):
    GPIO.output(27,1)
    print("LED turns on...")
    sleep(1)
    GPIO.output(27,0)
    print("LED turns off...")
    sleep(1)
```

See if you can change the program to blink the LED at a faster rate (e.g. on for 0.5 sec, off for 0.5 sec) 3 times, before turning it off for a further 3 sec and repeat.

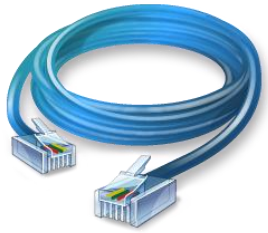

Thank
You

Appendix – Using Raspberry Pi in “Headless Mode”

Using Raspberry Pi in “Headless Mode”

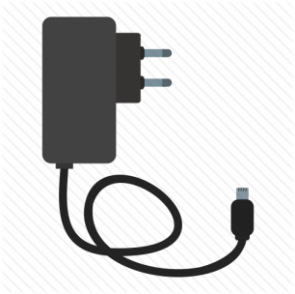
You can use Raspberry Pi in “Headless Mode” i.e. without a monitor, a mouse or a keyboard.

Your Micro SD card (16G)

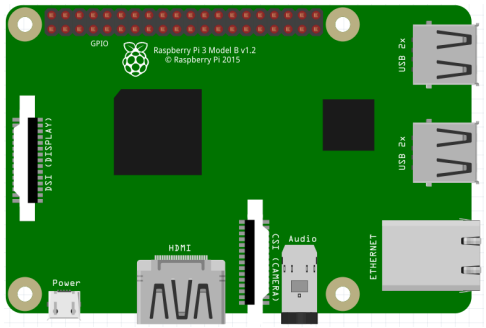


LAN cable

A laptop is connected to the Raspberry Pi to provide the user interface (a screen, a mouse & a keyboard).



5V 2.5A adaptor



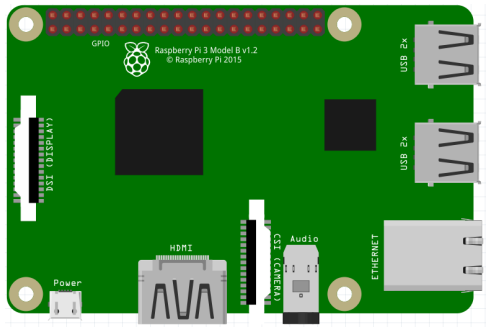
Your laptop

The connection can be wired or wireless. But the Raspberry Pi and the laptop must be in the same LAN (Local Area Network).

Using Raspberry Pi in "Headless Mode" (cont.)

There are a few steps involved:

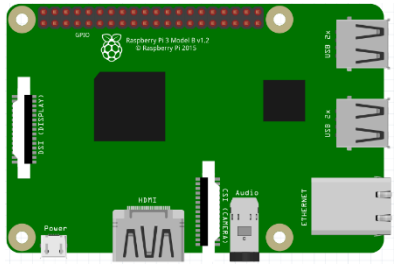
1. Setting a **static IP address** for the **Raspberry Pi.**
2. Enabling **VNC** (Virtual Network Computing) for the Raspberry Pi.
3. **Reboot**ing the Raspberry Pi.



4. Setting a **static IP address** for the **laptop** – this must be in the **same LAN** as the Raspberry Pi.
5. Downloading and installing **VNC Viewer.**
6. Connecting the laptop to the Raspberry Pi **wirelessly**, or by a **LAN cable**.
7. Running VNC Viewer, and entering Raspberry Pi's IP address.

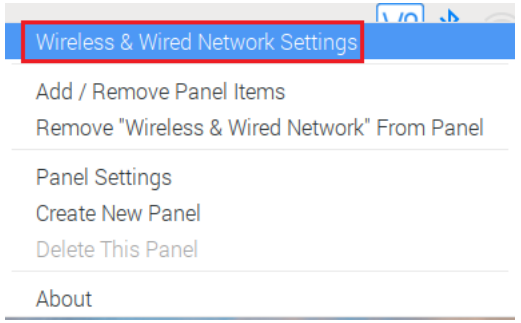
These steps will be explained in more details next.

Using Raspberry Pi in “Headless Mode” (cont.)

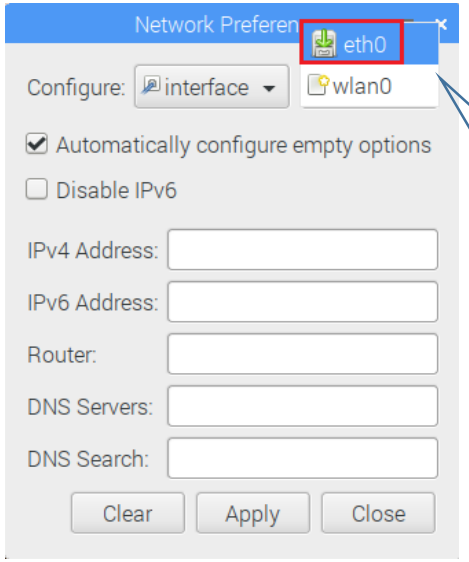


1. Setting a **static IP address** for the **Raspberry Pi**

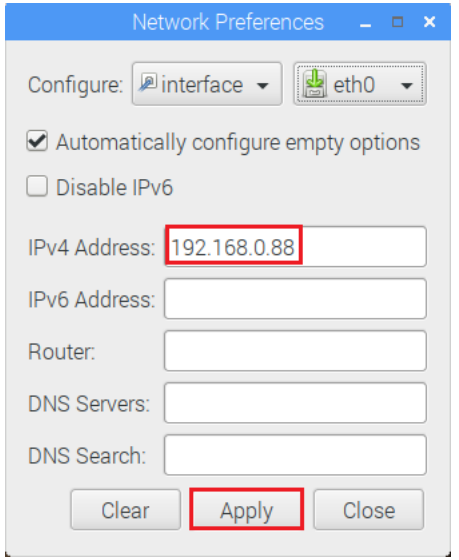
1a. Right-click on the internet access icon.



1b. Select Wireless & Wired Network Settings.

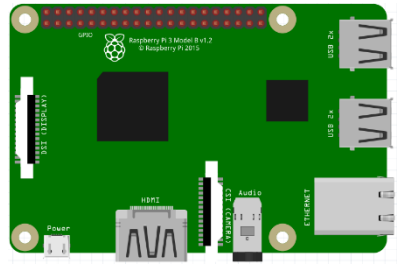


1c. Select eth0 (if you are using LAN cable to connect to a laptop) or wlan0 (if you are using wireless) to configure.

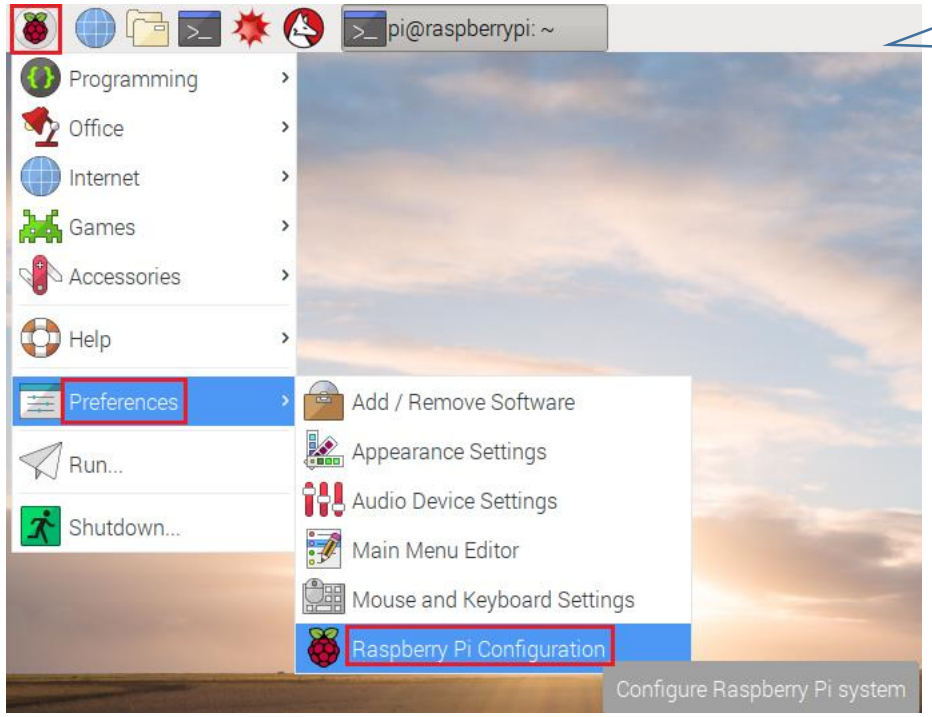


1d. Enter a static IP address e.g. 192.168.0.88 and click Apply.

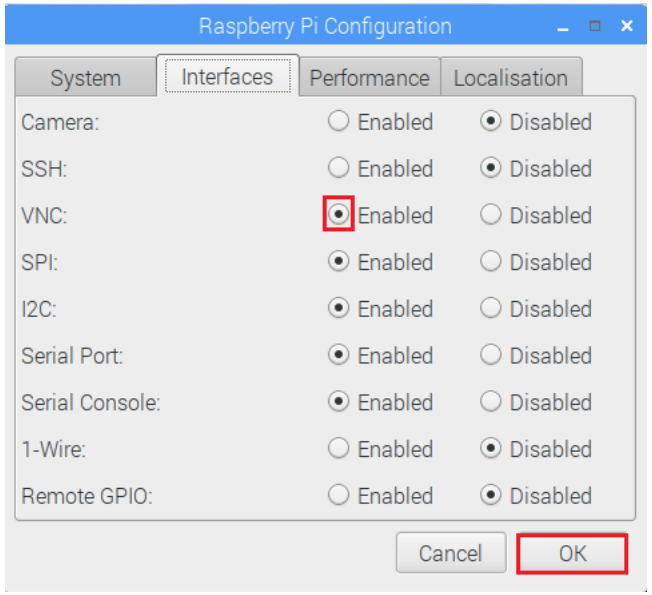
Using Raspberry Pi in “Headless Mode” (cont.)



2. Enabling **VNC** (Virtual Network Computing) for the Raspberry Pi.



2a. Click Raspberry Pi ⇨ Preferences ⇨ Raspberry Pi Configuration.

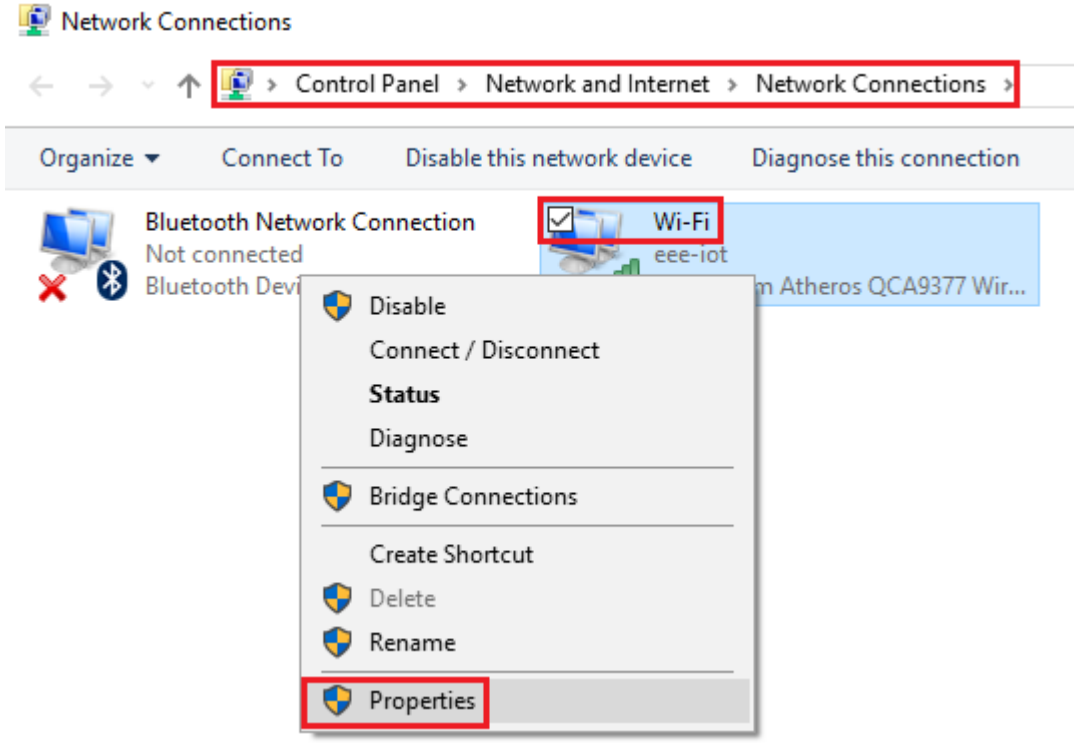


2b. Enable VNC and click OK.

3. **Reboot** the Raspberry Pi now.

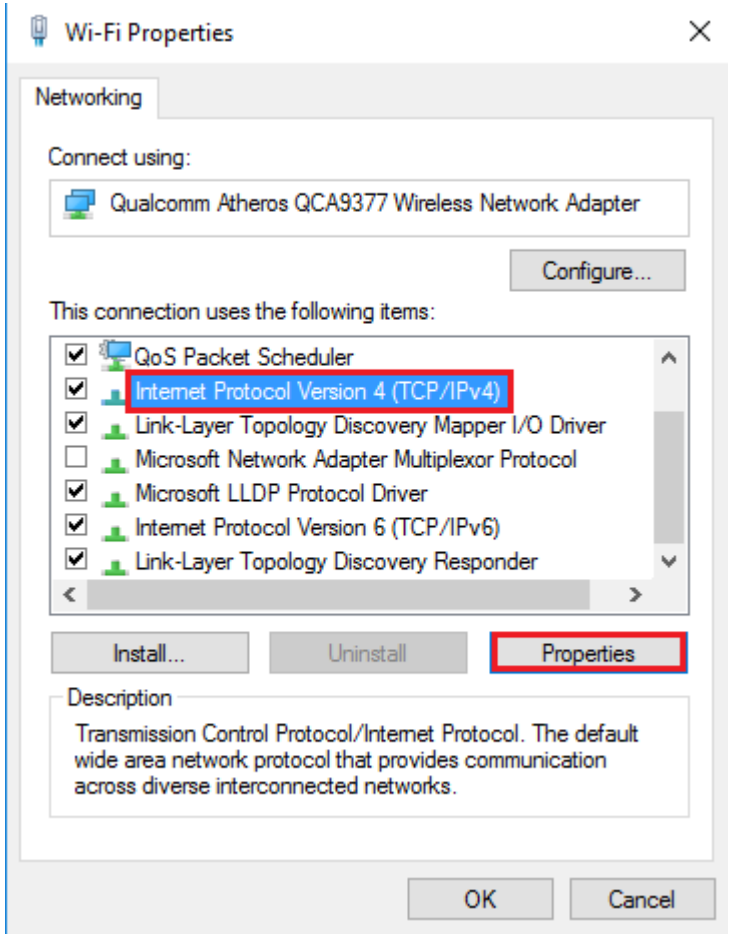
Using Raspberry Pi in “Headless Mode” (cont.)

4. Setting a **static IP address** for the **laptop** – this must be in the **same LAN** as the Raspberry Pi.

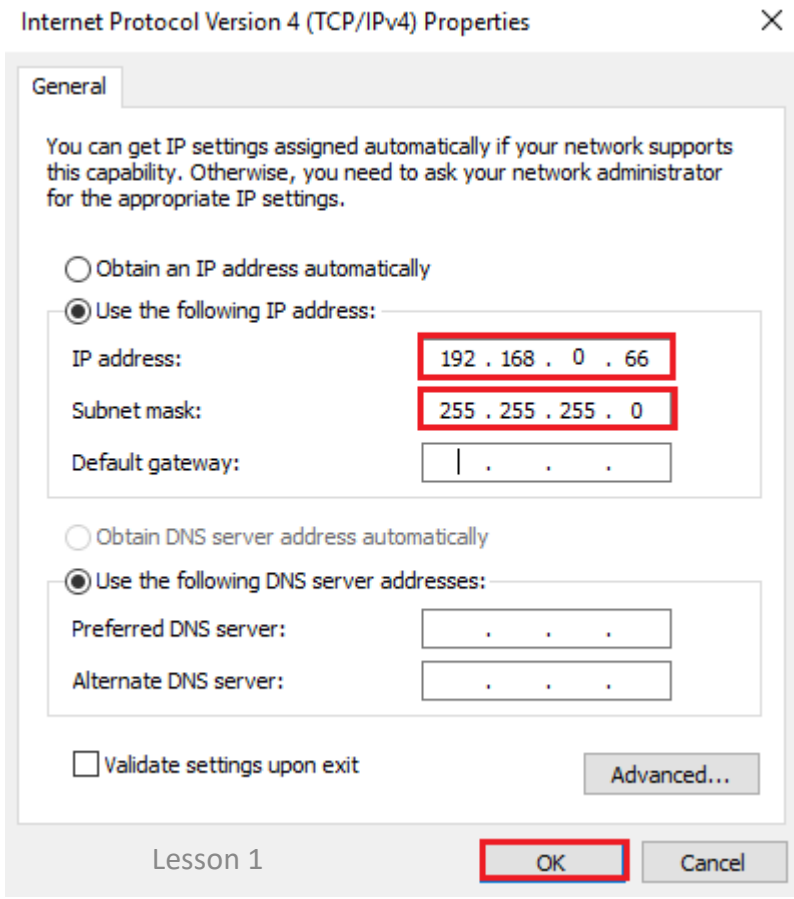


4a. Go to the Network Connections. Right-click on Wi-Fi (if you are using wireless to connect to the Raspberry Pi) or Ethernet (if you are using LAN cable) and select Properties.

Using Raspberry Pi in “Headless Mode” (cont.)



4b. Click on Internet Protocol Version 4 (TCP/IPv4) and then Properties.

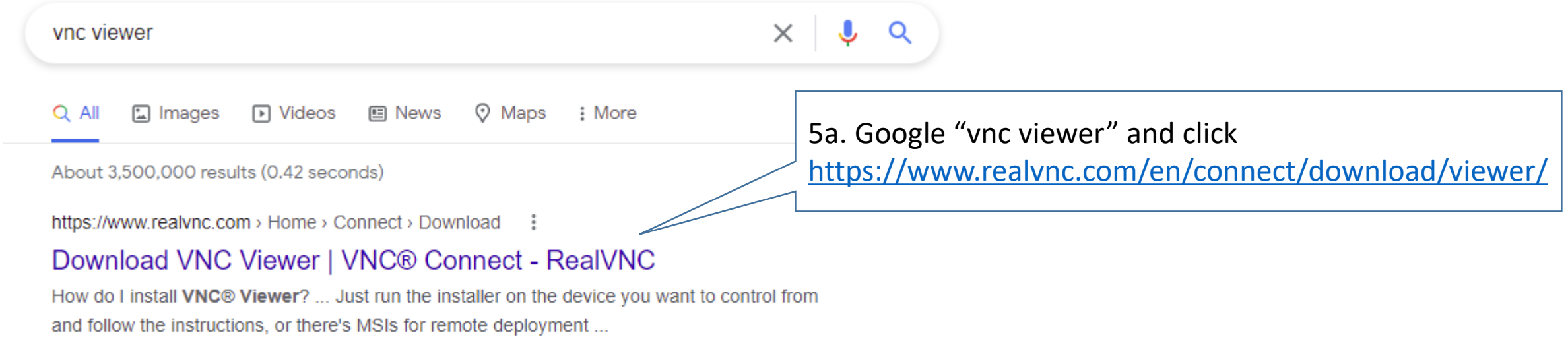


4c. Enter a static IP address e.g. 192.168.0.66 and a /24 mask and click OK.

4d. Back to the previous window, click Close.

Using Raspberry Pi in “Headless Mode” (cont.)

5. Downloading and installing **VNC Viewer**



5a. Google “vnc viewer” and click https://www.realvnc.com/en/connect/download/viewer/

Using Raspberry Pi in “Headless Mode” (cont.)

5b. Note that a VNC Server is running on the Raspberry Pi (a license fee is payable for Server). What we need is a VNC Client to run on the Windows PC (no license fee needed).

VNC® Conne… …ts of VNC® Viewer and VNC® Server

Download VNC® Viewer to t… …e you want to control from, below. Make sure you've installed VNC® Server on the computer you want to control.

Windows | macOS | Linux | Raspberry Pi | iOS | Android | Solaris | HP-UX | AIX

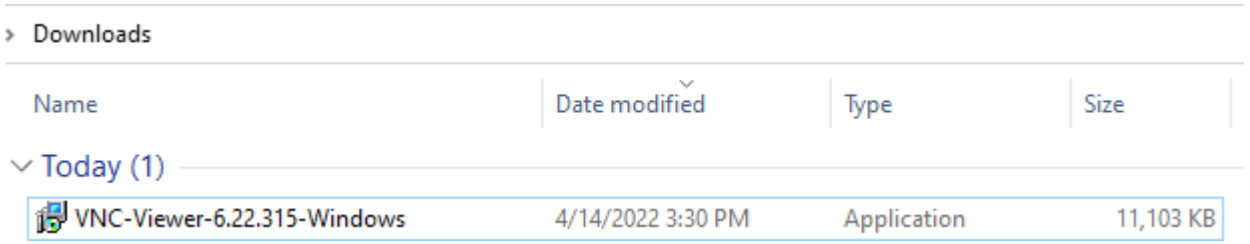
Download VNC Viewer

SHA256: 788b4817e1686491924f9e3bd166265bc2b477e73a7703103f512fb02f15b5f5

EXE x86/x64

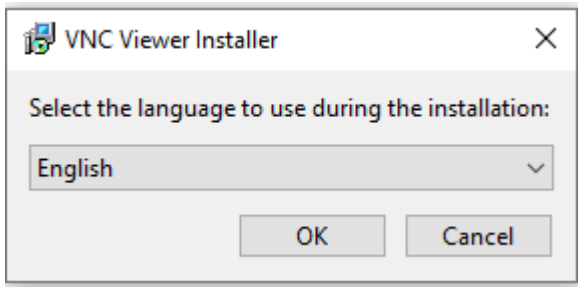
5c. Click Download VNC Viewer.

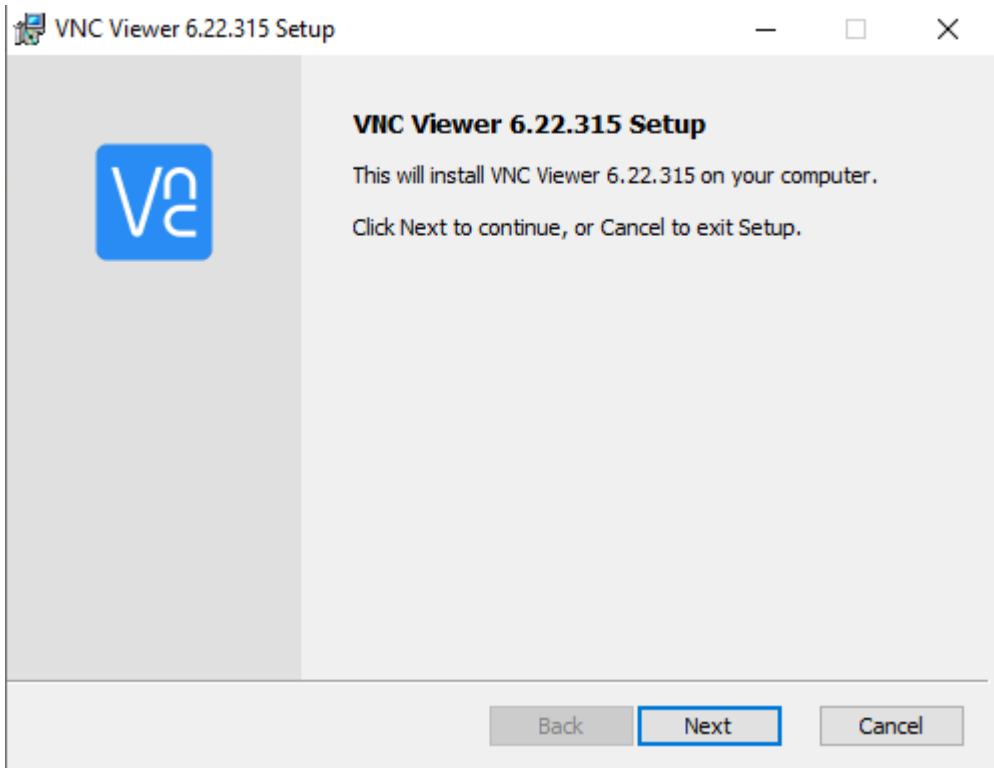
Using Raspberry Pi in “Headless Mode” (cont.)



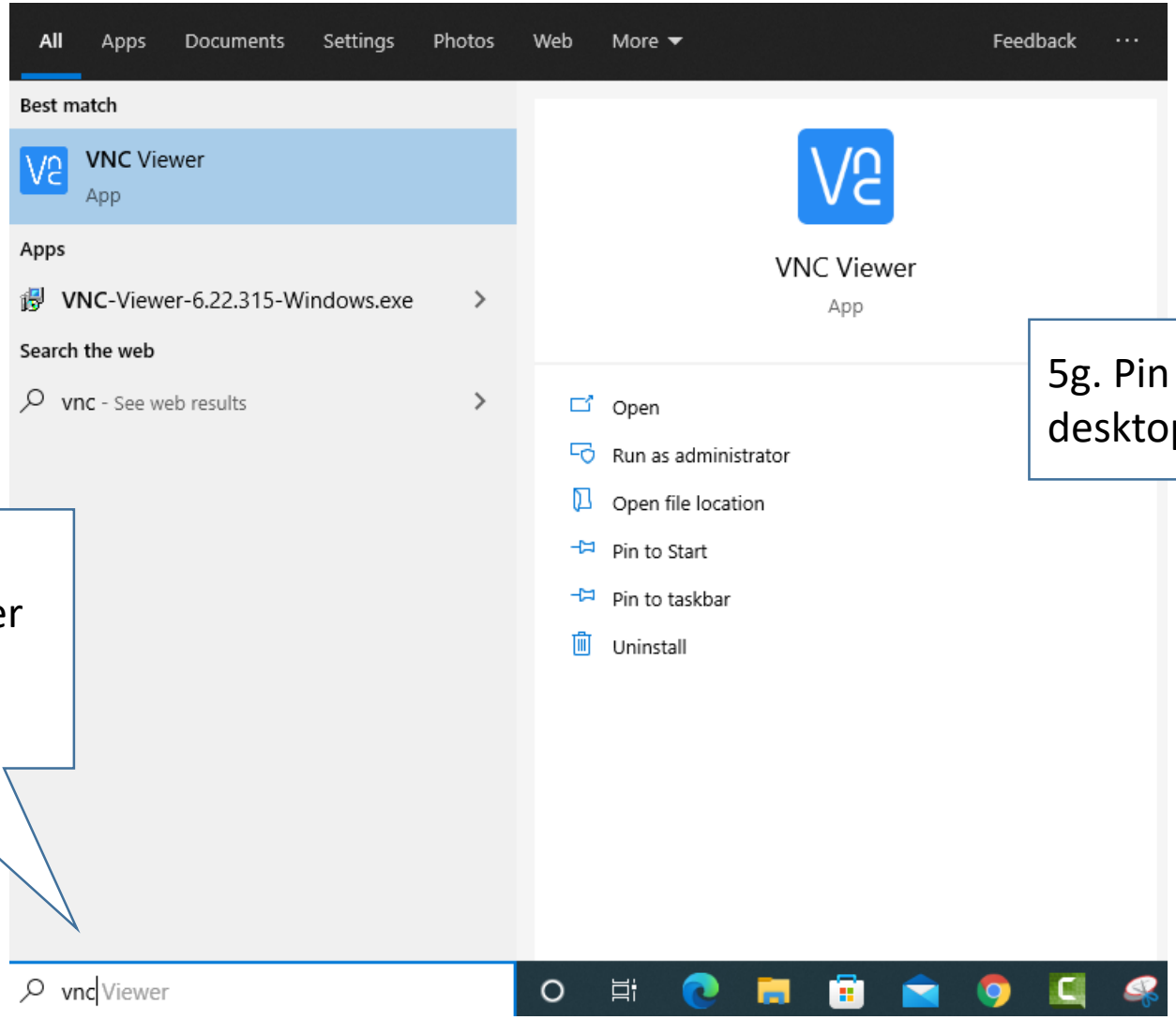
5d. Double click on the installer to start installing.

5e. Follow the on-screen instructions to complete the installation.





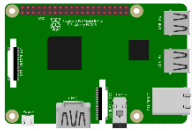
Using Raspberry Pi in “Headless Mode” (cont.)



5f. You can search for the VNC Viewer if it is not on your desktop.

5g. Pin it to your taskbar or add a desktop icon for easy access in future.

Using Raspberry Pi in “Headless Mode” (cont.)



6. Connecting the laptop to the Raspberry Pi **wirelessly**, or by a **LAN cable**.

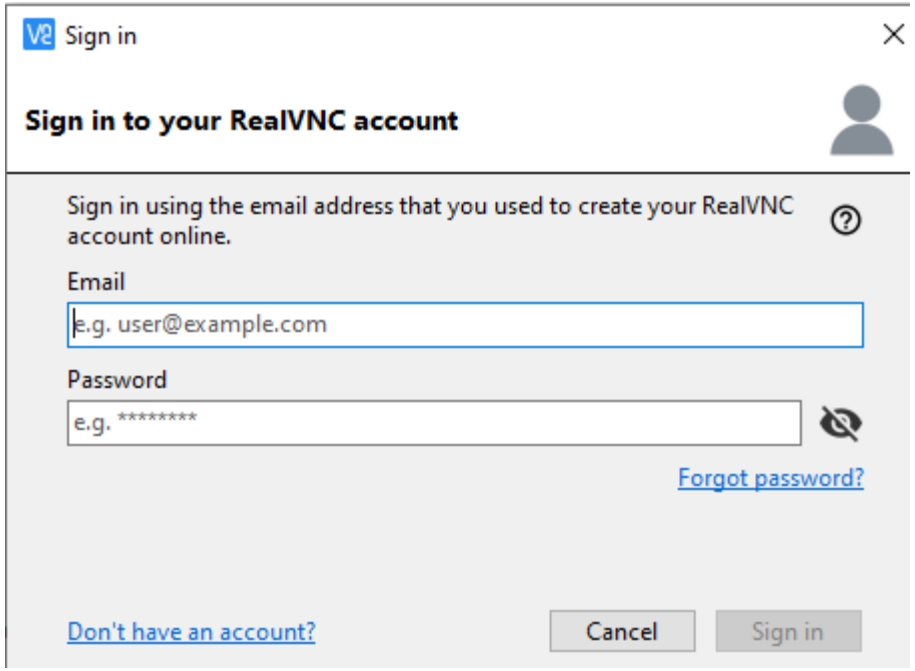
6a. For wireless connection, ensure that both the laptop and the Raspberry Pi are connected to the same wireless LAN.

6b. For wired connection, use a LAN cable to connect the laptop and the Raspberry Pi.

Using Raspberry Pi in “Headless Mode” (cont.)

7. Running VNC Viewer, and entering Raspberry Pi’s IP address.

7a. The first time you launch VNC Viewer, you may be prompted to sign in.



7b. Just close this Sign in window. As mentioned, VNC Client / Viewer does not need a license to run.

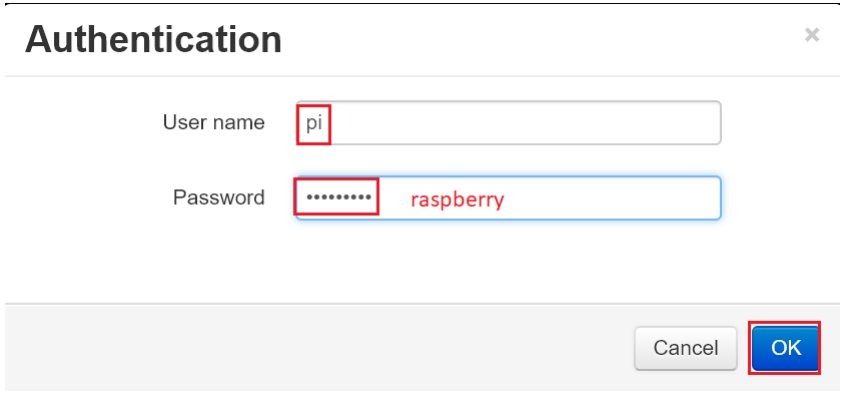
Using Raspberry Pi in “Headless Mode” (cont.)

7c. Enter the IP address of the Raspberry Pi in the url.

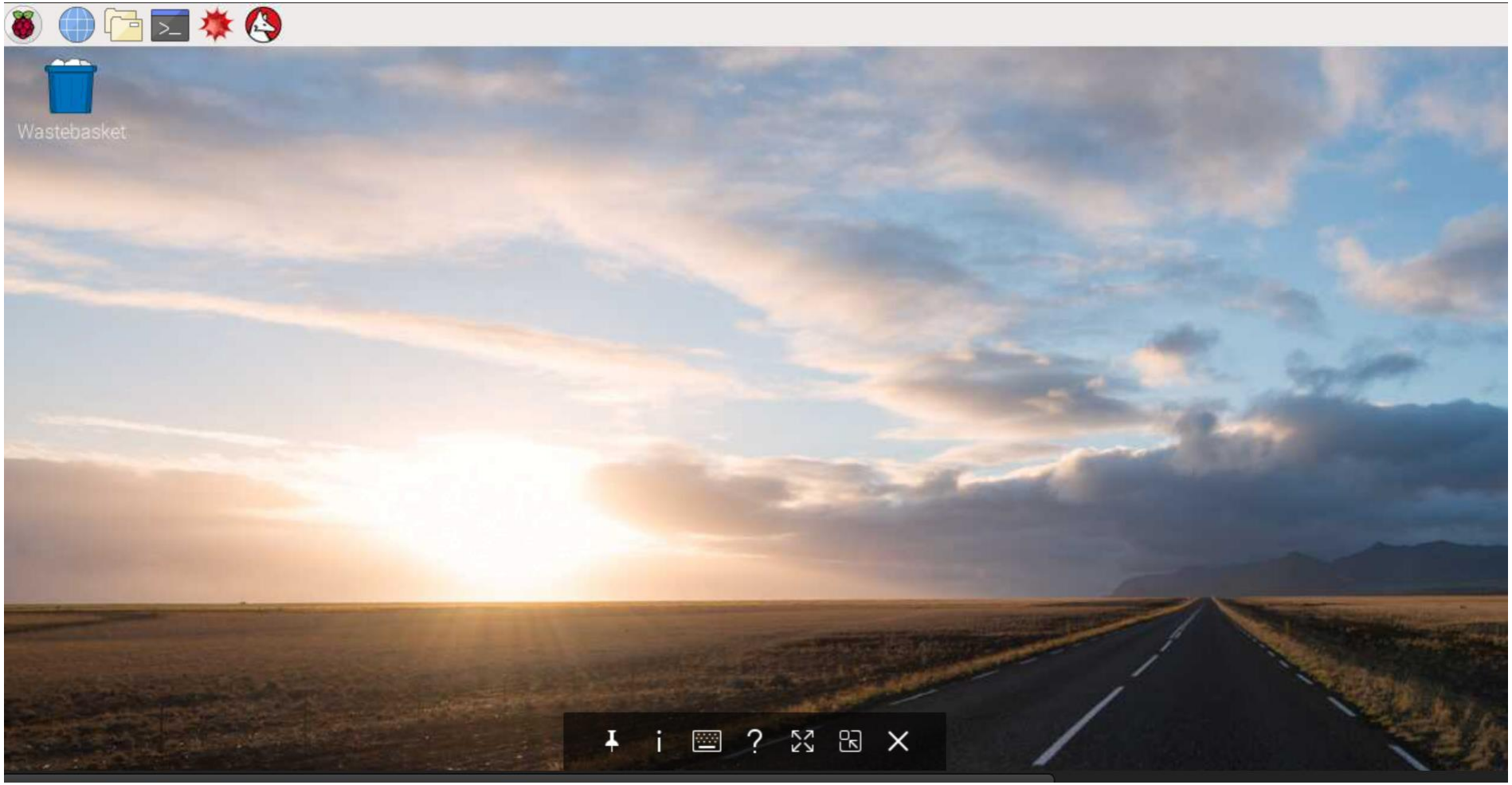
7d. Note that different IP addresses may be used for wireless access and wired access. Make sure you use the correct IP address.

Using Raspberry Pi in “Headless Mode” (cont.)

7e. Enter “pi” as the User name and “raspberry” as the Password and click “OK”.



Using Raspberry Pi in “Headless Mode” (cont.)



7f. You will be able to access the Raspberry Pi’s GUI from your laptop.