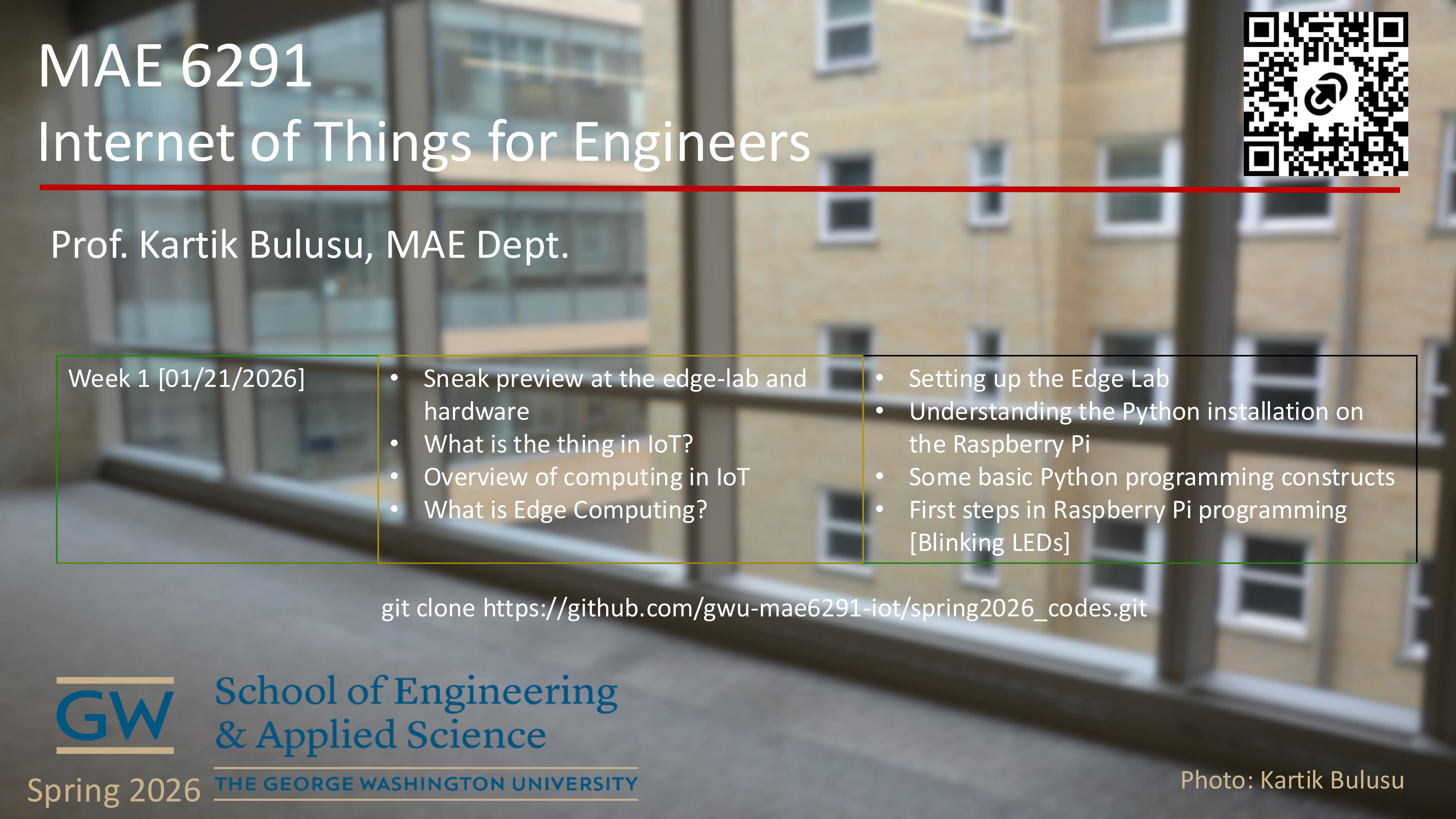


MAE 6291
Internet of Things for Engineers

Prof. Kartik Bulusu, MAE Dept.

Week 1 [01/21/2026]	• Sneak preview at the edge-lab and hardware • What is the thing in IoT? • Overview of computing in IoT • What is Edge Computing?	• Setting up the Edge Lab • Understanding the Python installation on the Raspberry Pi • Some basic Python programming constructs • First steps in Raspberry Pi programming [Blinking LEDs]

```
git clone https://github.com/gwu-mae6291-iot/spring2026_codes.git
```

GW School of Engineering & Applied Science

THE GEORGE WASHINGTON UNIVERSITY

Spring 2026

Photo: Kartik Bulusu

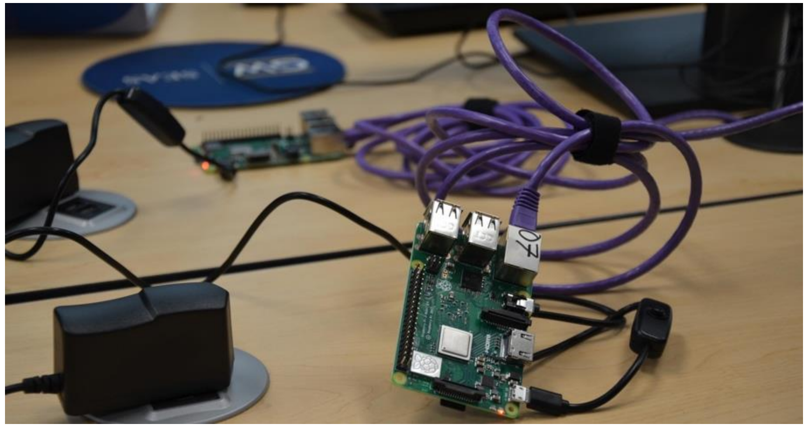
Setting up the Edge-lab each week

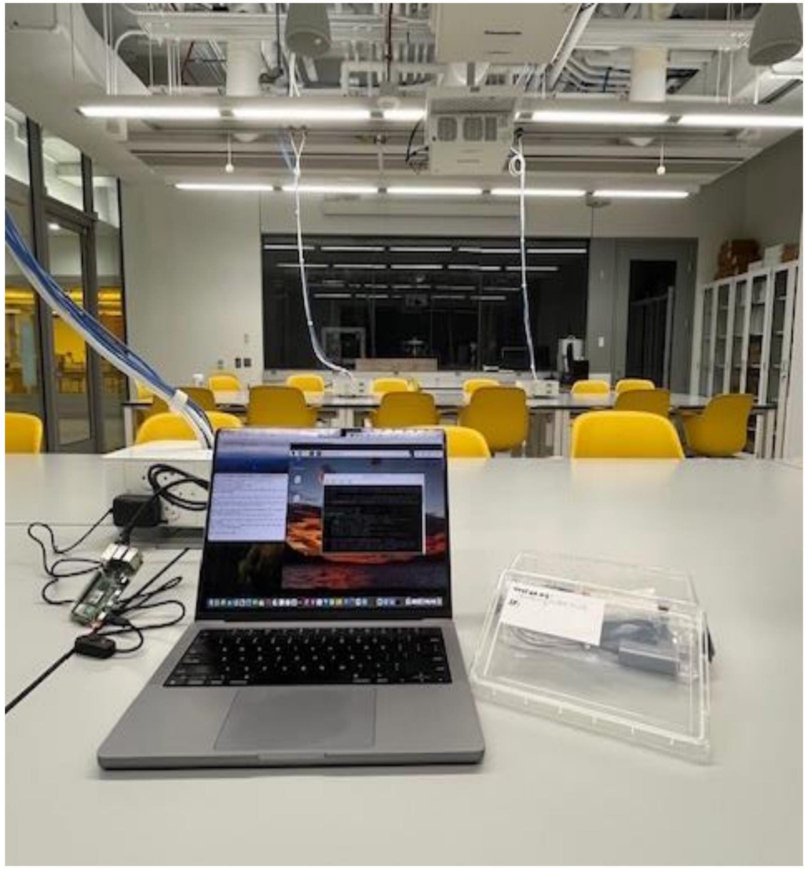
Work together to set it up .. every week!

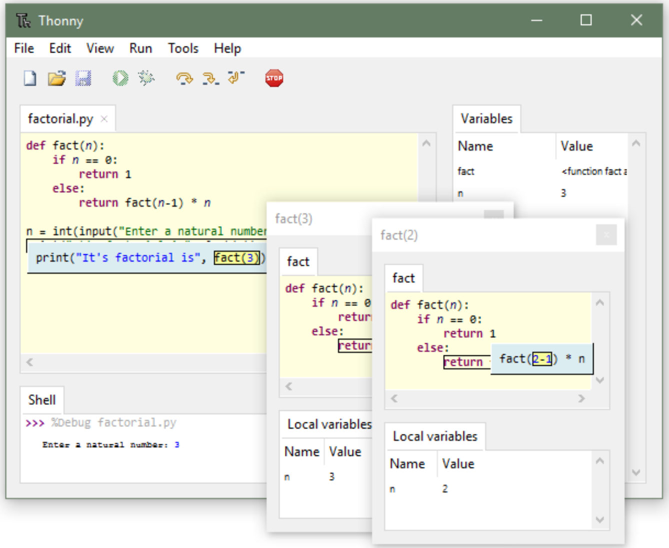
Sources:

Thonny (IDE): https://en.wikipedia.org/wiki/Thonny

Thonny: https://thonny.org







factorial.py

```
def fact(n):
    if n == 0:
        return 1
    else:
        return fact(n-1) * n

n = int(input("Enter a natural number
print("It's factorial is", fact(3))
```

Shell

```
>>> %Debug factorial.py
  Enter a natural number: 3
```

Variables

Name	Value
fact	<function fact a
n	3

fact(3)

```
def fact(n):
    if n == 0
        retur
    else:
        retur
```

Local variables

Name	Value
n	3

fact(2)

```
def fact(n):
    if n == 0:
        return 1
    else:
        return fact(2-1) * n
```

Local variables

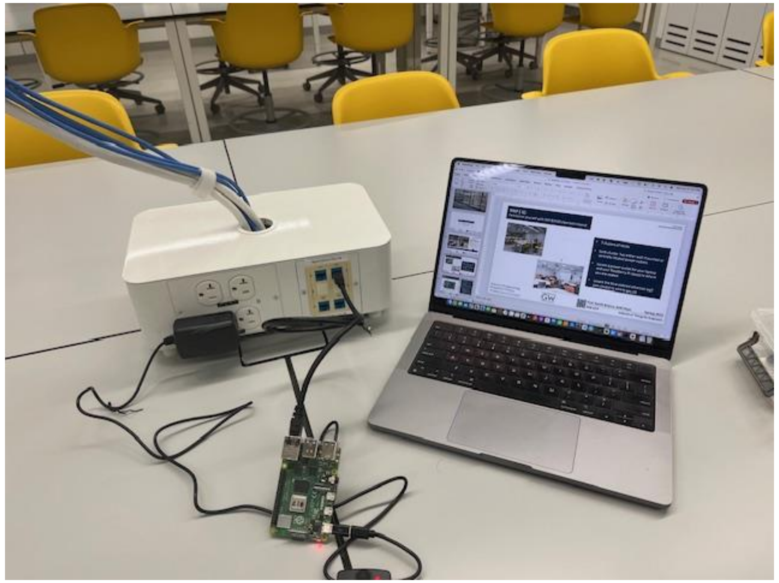
Name	Value
n	2

SEH B3040: Laboratory set up

Image Source:
https://seascf.seas.gwu.edu/th-201

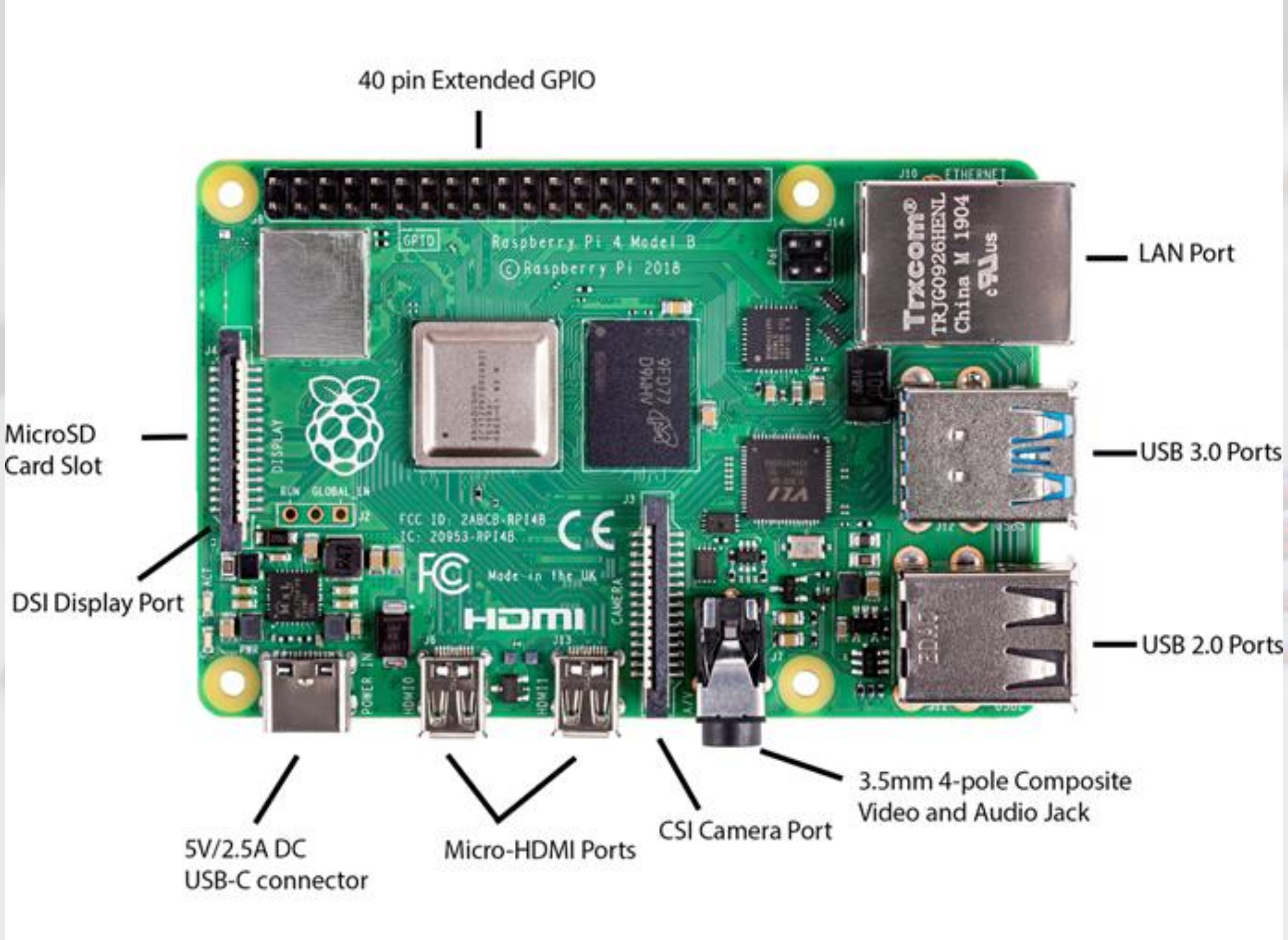
STEP [-1]:
Familiarize yourself with SEH B2020 classroom layout

- 7 clusters of desks
- Each cluster has either wall-mounted or centrally located power outlets
- Locate a power outlet for your laptop and your Raspberry Pi closest to where you are seated
- Locate the blue-colored ethernet wall jack closest to where you are seated



STEP [0]:

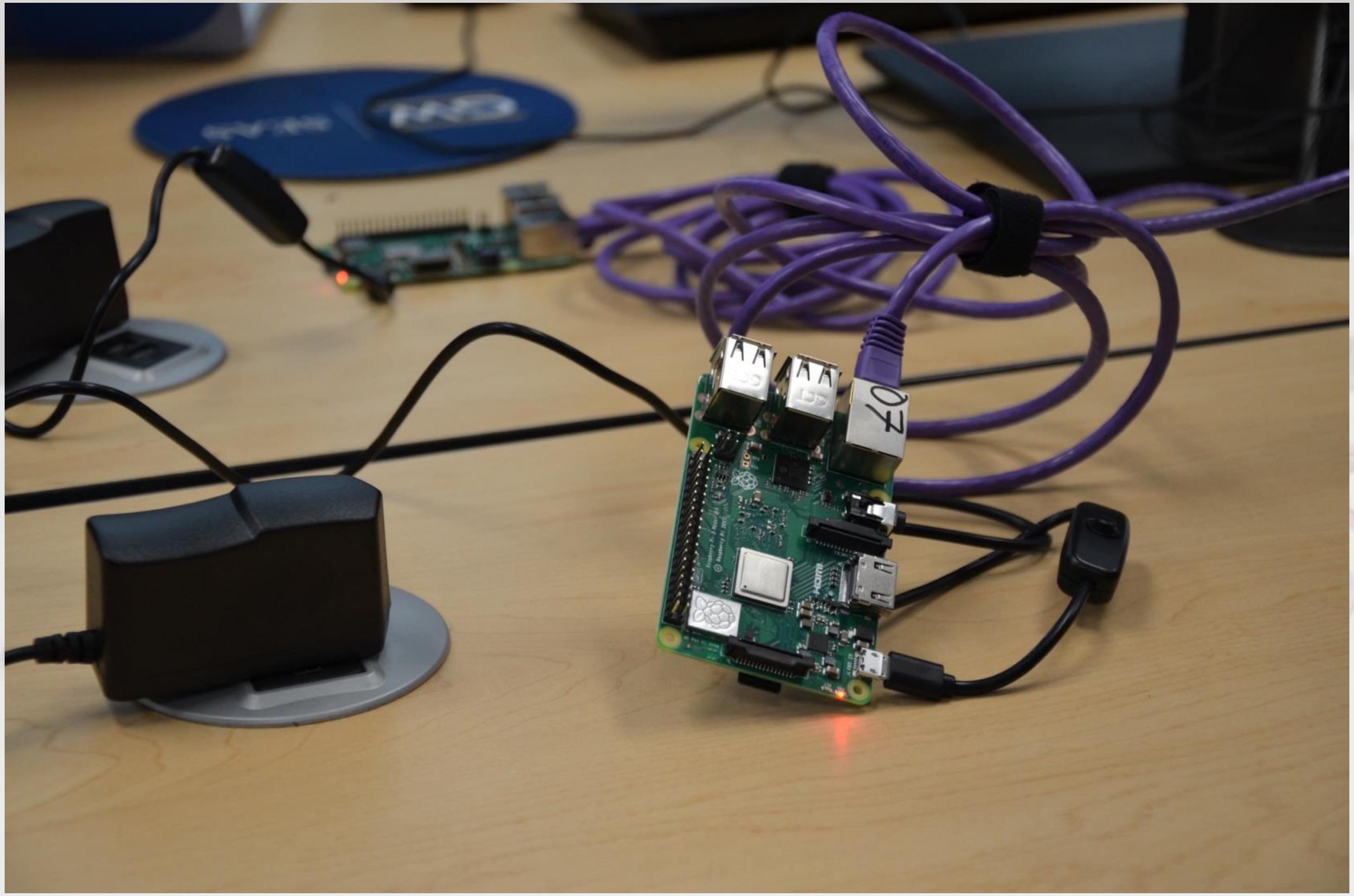
Familiarize yourself with the Raspberry Pi 4B (RPi) layout



- Each RPi will come with a preconfigured OS in a microSD card
- Each RPi will have a microSD card already installed
- Each RPi is assigned a unique alpha-numeric name (e.g., 007, 015 etc)
- To power, boot up and test the RPI you will need
  - USB-C power cord
  - Ethernet cord for 10/100 LAN port
  - IP address
  - Login and password for the RPi
  - Remote desktop feature enabled in your computer

STEP [1]:

Connect the RPis to each desk power outlet as shown



- Make sure there is a microSD card installed in the RPi
- Connect the RPi using the microUSB power cable provided
- Connect the ethernet cables specifically for RPi connections
- LEDs on the RPi will start blinking indicating that it is booting up

STEP [2]:
Access the RPi in the Edge-lab

2.1 Log in to GWU Virtual private network (GWU VPN)
Downloads: https://it.gwu.edu/vpn-global-protect

2.2 Open up remote desktop connection (using the VNC server)

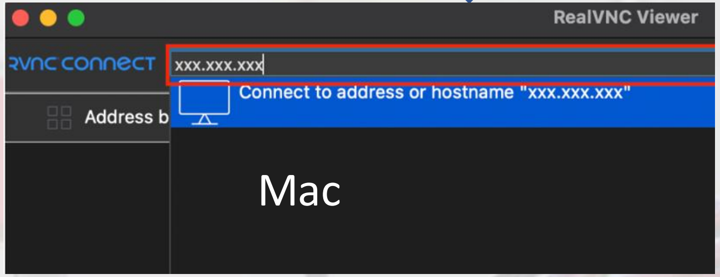
Each RPi has unique alpha-numeric name (e.g., 007, 015 etc.)

- Locate the Pi-name and the IP address on the <128.164.139.xx>

OR

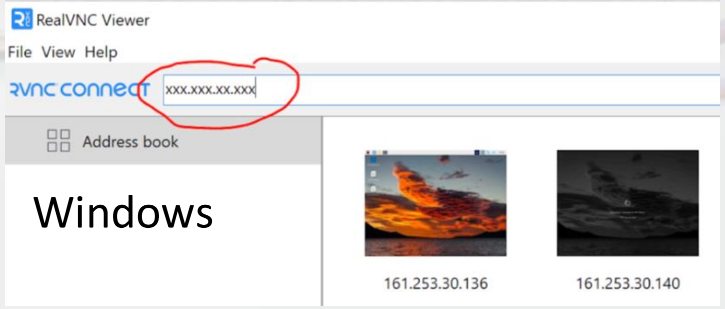
Each RPi connected using an ethernet cable directly to your laptops

- raspberrypi###.local



xxx.xxx.xxx
Connect to address or hostname "xxx.xxx.xxx"

Mac



xxx.xxx.xx.xxx

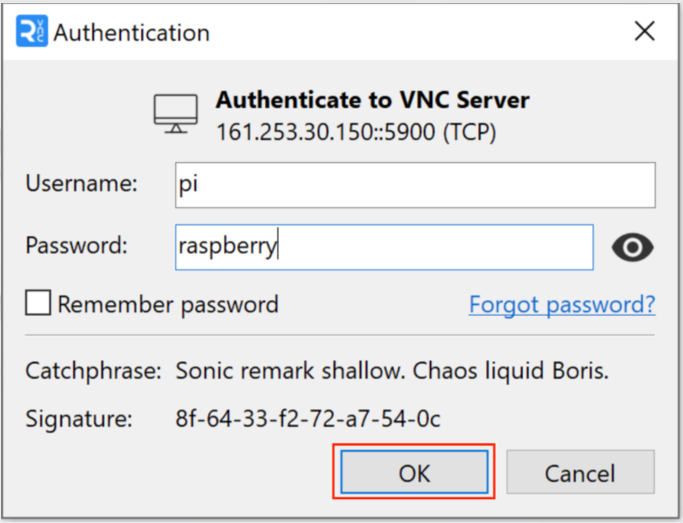
Windows

161.253.30.136

161.253.30.140

2.3 Once you are connected you will

- See Authentication box below
- Type in the Username and Password



Authenticate to VNC Server
161.253.30.150::5900 (TCP)

Username: pi

Password: raspberry

☐ Remember password

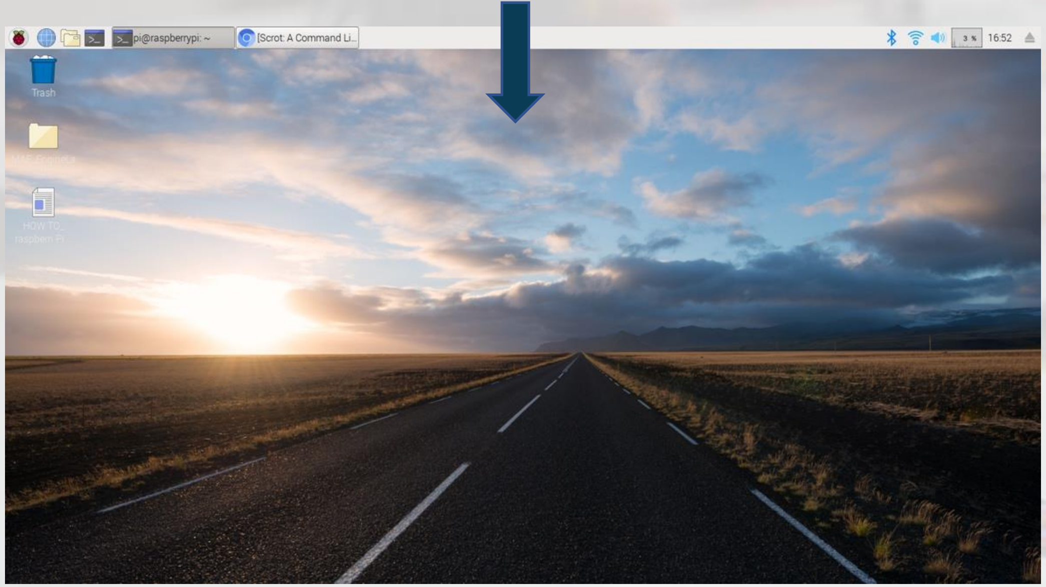
Catchphrase: Sonic remark shallow. Chaos liquid Boris.

Signature: 8f-64-33-f2-72-a7-54-0c

STEP [3]:

Now that you accessed the RPi…

You will see a screen like the one shown below

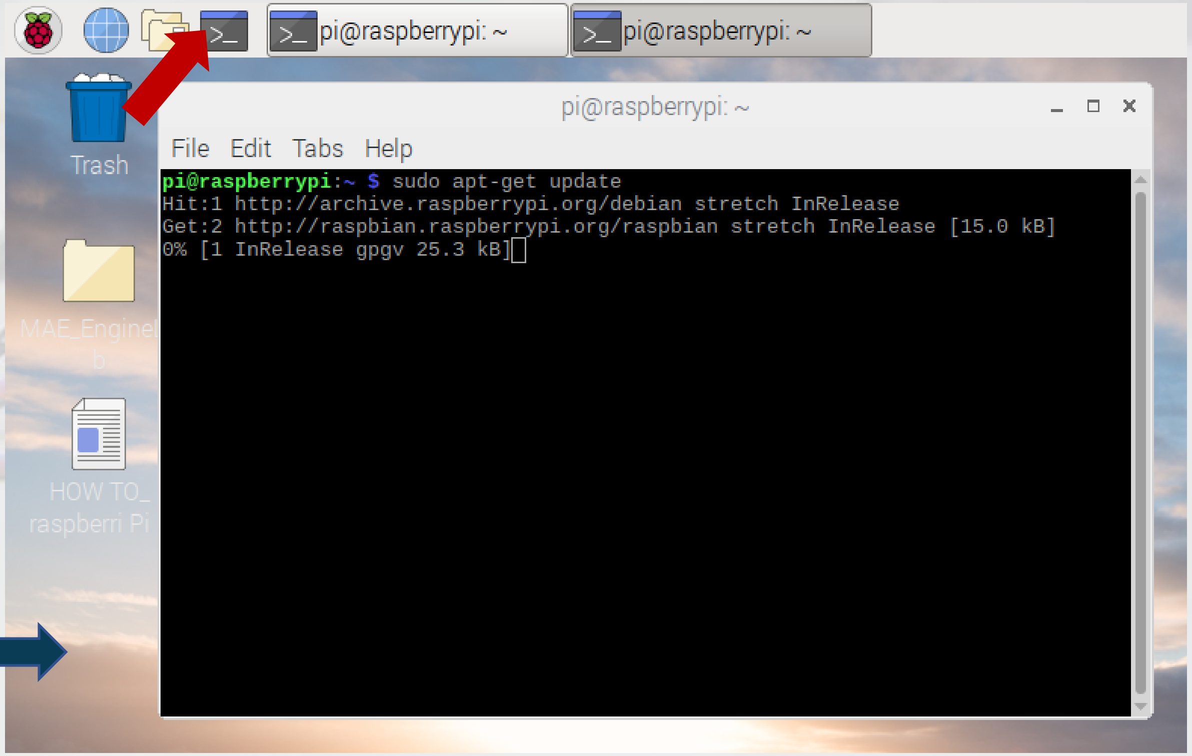


3.2 Testing is complete when you get to this step.

- Students should get the RPis to this step before the laboratory modules begin.

3.1 Click on terminal(shown with a red arrow below)

- At the prompt type: sudo apt-get update
- Wait for the updates to complete
- Then type: sudo apt-get upgrade
- If you get the following prompt
  - Do you want to continue [Y/n]
  - Type: y
  - And hit “Enter” on your keyboard and let the upgrades complete



```
pi@raspberrypi:~ $ sudo apt-get update
Hit:1 http://archive.raspberrypi.org/debian stretch InRelease
Get:2 http://raspbian.raspberrypi.org/raspbian stretch InRelease [15.0 kB]
0% [1 InRelease gpgv 25.3 kB]
```

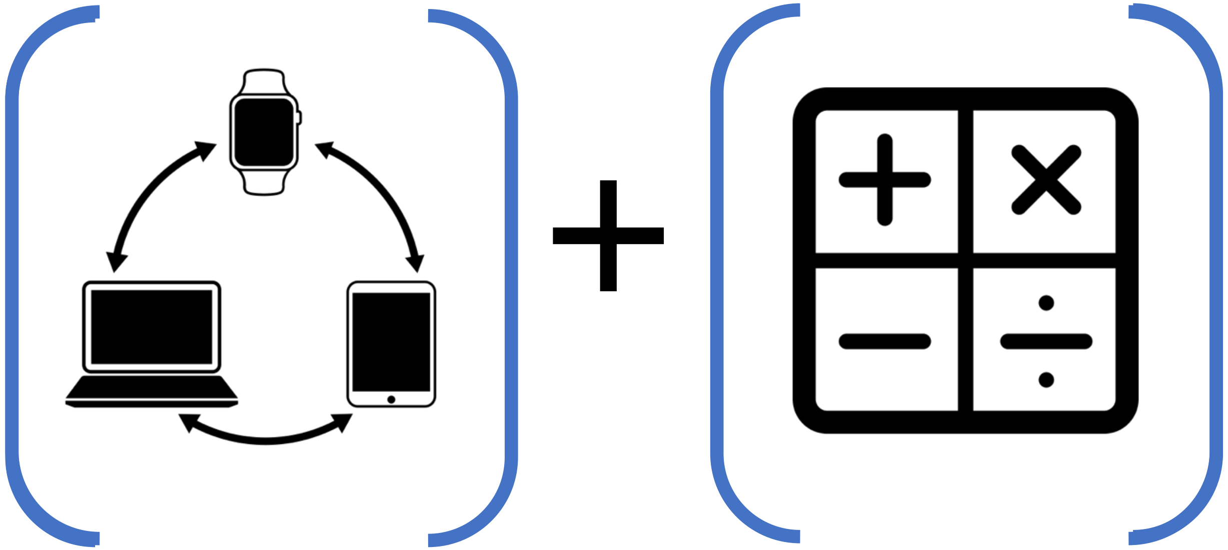
Sources:
Calculator by Markus from https://thenounproject.com/browse/icons/term/calculator/
internet of things by Davo Sime from https://thenounproject.com/browse/icons/term/internet-of-things/
L. Bernardi, S. Sarma and K. R. Traub, The Inversion Factor: How to Thrive in the IoT Economy

What's the "thing"?

Paradigm #1

- A **thing** is self-contained and only operates within the confines of its physical shell.
  - **Thing** carries out only those functions that its designer envisioned when it was fabricated.
- The **thing** contains a powerful computer inside but is completely hidden from the user.
- The **thing** has firmware (not called software).

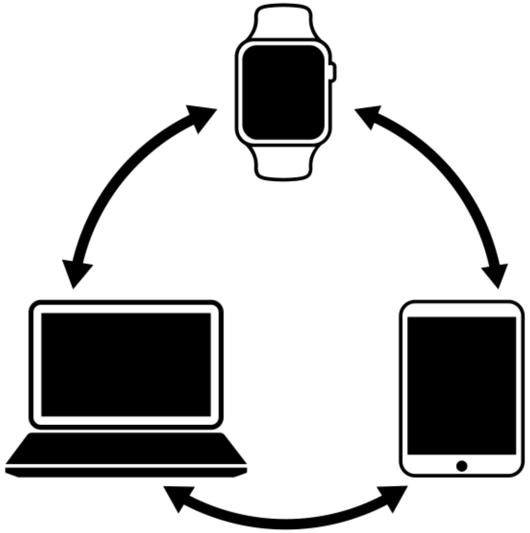
Paradigm #2



Paradigm #3

Connected “things”

Icon Sources:
Bluetooth by J703 from https://thenounproject.com/browse/icons/term/Bluetooth
wifi by Cetha Studio from https://thenounproject.com/browse/icons/term/wifi/
By Connectivity Standards Alliance - https://zigbeealliance.org/solutions/ , Public Domain: https://commons.wikimedia.org/w/index.php?curid=106668572
By Connectivity Standards Alliance - buildwithmatter.com, Public Domain, https://commons.wikimedia.org/w/index.php?curid=114216376
internet of things by Davo Sime from https://thenounproject.com/browse/icons/term/internet-of-things/



zigbee

matter

5G

Use terminal commands to

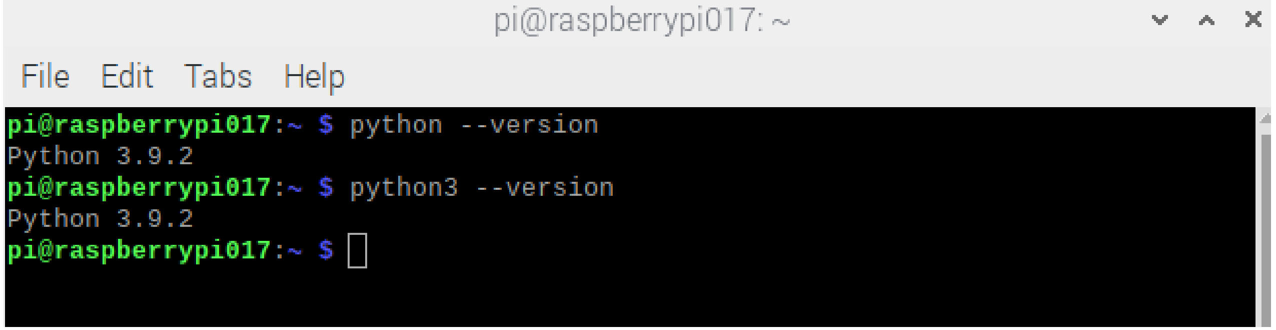
- check Raspberry Pi OS version
- check Python installation version
- update and upgrade Raspberry Pi OS

Installed Python version Raspberry Pi: 3.9.2

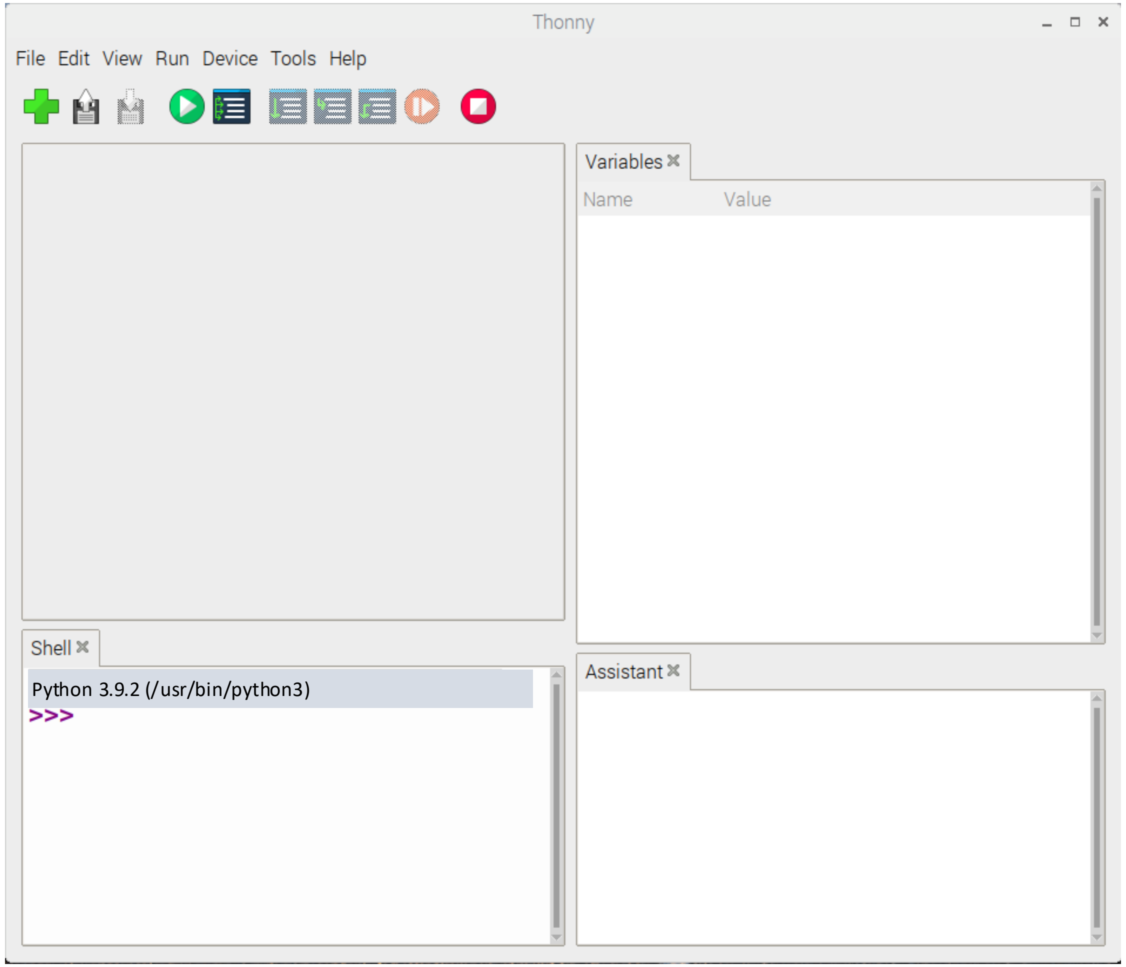
The shell window on your Thonny IDE screen should show the same version as the Terminal

Type the following one after the other:

```
$ python --version
$ python3 --version
```



```
pi@raspberrypi017:~ $ python --version
Python 3.9.2
pi@raspberrypi017:~ $ python3 --version
Python 3.9.2
pi@raspberrypi017:~ $ 
```



```
Python 3.9.2 (/usr/bin/python3)
>>>
```

“python –version” and ”python3 --version” will return the same output i.e., Python 3.9.2.

That is because earlier versions of Raspberry Pi OS had two versions of Python (python 2.x.x and python 3.x.x) installed. On January 1, 2020, Python 2 was sunset or deprecated and all legacy Python 2 users had to move to Python 3. Earlier versions of Raspberry Pi OS carried the last version of Python 2 but it has now been completely phased out.

Installed version of Raspberry Pi OS

Type the following line in your **Terminal**:

```
$ cat /etc/os-release
```

Output on your **Terminal should look like the following:**

```
PRETTY_NAME="Raspbian GNU/Linux 11 (bullseye)" NAME="Raspbian
GNU/Linux"
VERSION_ID="11"
VERSION="11 (bullseye)"
VERSION_CODENAME=bullseye
ID=raspbian
ID_LIKE=debian
HOME_URL="http://www.raspbian.org/"
SUPPORT_URL="http://www.raspbian.org/RaspbianForums"
```

More information on the bullseye OS version and it's kernel version can be found in the following wikipedia page: https://en.wikipedia.org/wiki/Raspberry_Pi_OS

update and upgrade your Raspberry Pi OS

Type the following one after the other:

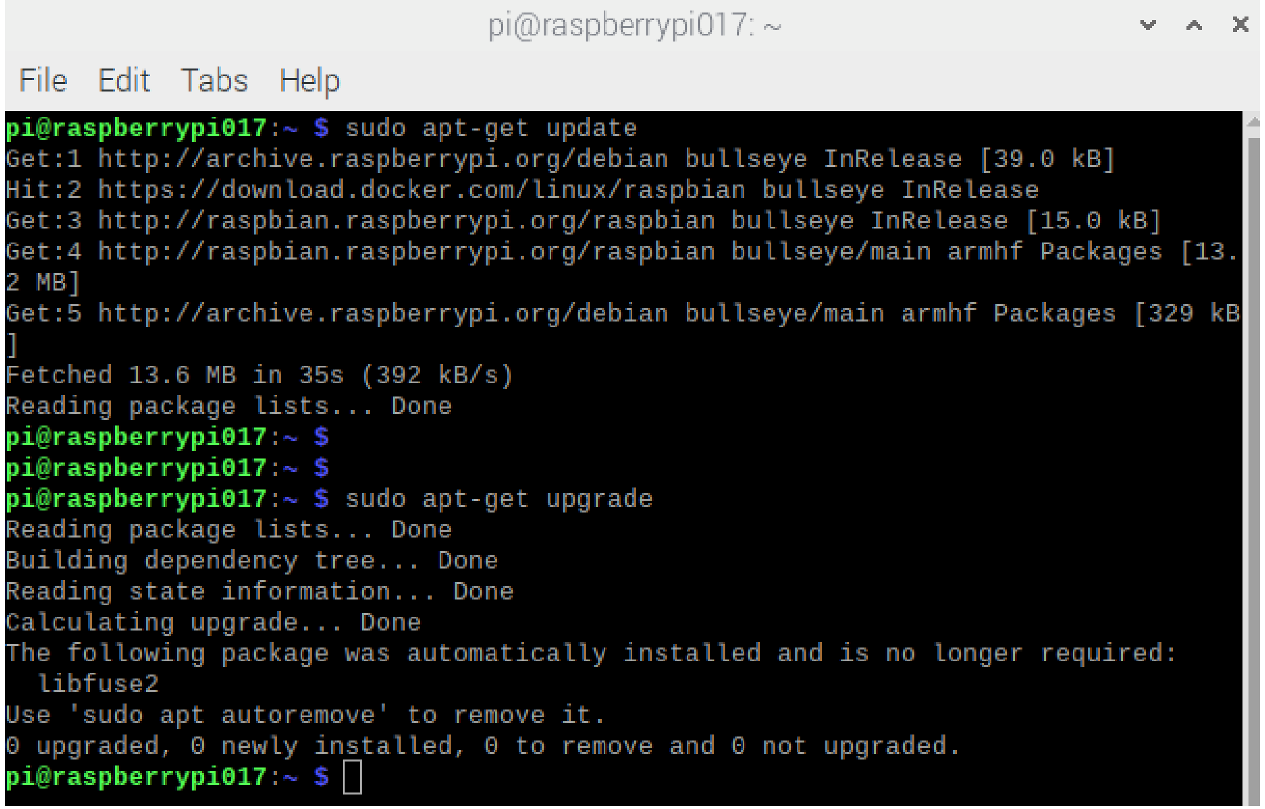
```
$ sudo apt-get update
$ sudo apt-get upgrade
```

Type the following in one line:

```
$ sudo apt-get update && sudo apt-get upgrade
```

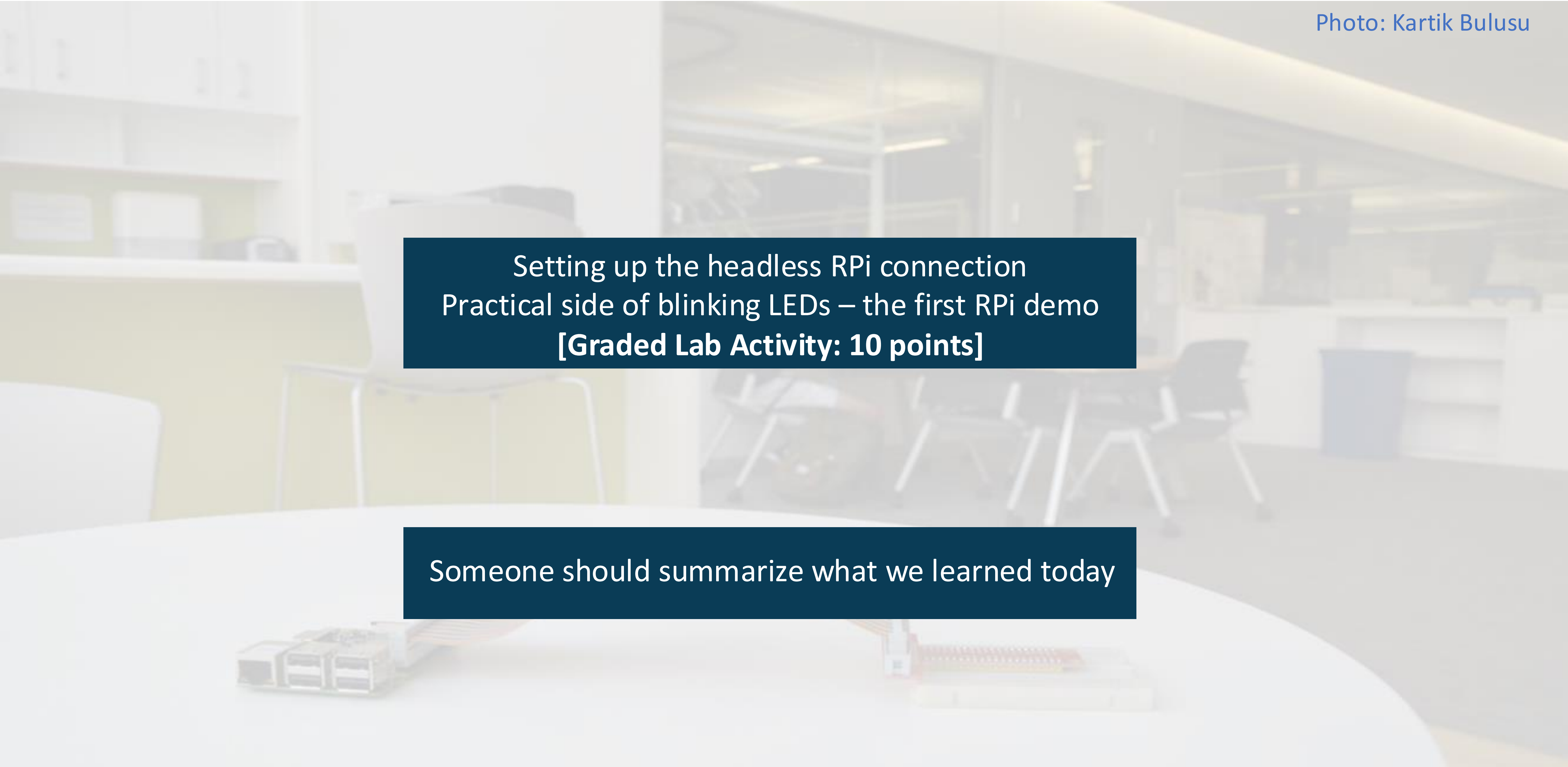
sudo apt-get update:
Updates the list of available packages for a Linux system

sudo apt-get update:
Installs the latest available updates for all currently installed packages on your system



```
pi@raspberrypi017:~ $ sudo apt-get update
Get:1 http://archive.raspberrypi.org/debian bullseye InRelease [39.0 kB]
Hit:2 https://download.docker.com/linux/raspbian bullseye InRelease
Get:3 http://raspbian.raspberrypi.org/raspbian bullseye InRelease [15.0 kB]
Get:4 http://raspbian.raspberrypi.org/raspbian bullseye/main armhf Packages [13.
2 MB]
Get:5 http://archive.raspberrypi.org/debian bullseye/main armhf Packages [329 kB
]
Fetched 13.6 MB in 35s (392 kB/s)
Reading package lists... Done
pi@raspberrypi017:~ $
pi@raspberrypi017:~ $
pi@raspberrypi017:~ $ sudo apt-get upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following package was automatically installed and is no longer required:
  libfuse2
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
pi@raspberrypi017:~ $
```

Photo: Kartik Bulusu

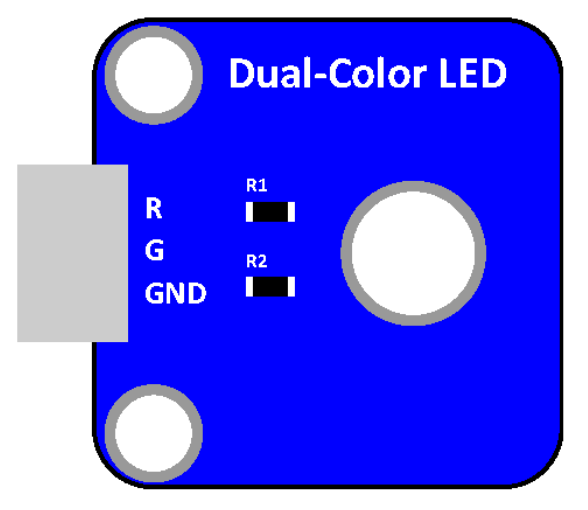


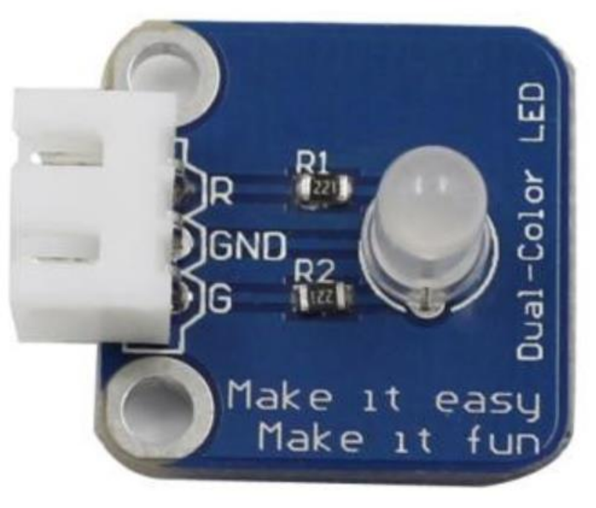
Setting up the headless RPi connection
Practical side of blinking LEDs – the first RPi demo
[Graded Lab Activity: 10 points]

Someone should summarize what we learned today

Photo: Kartik Bulusu

Know your Light Emitting Diode (LED)





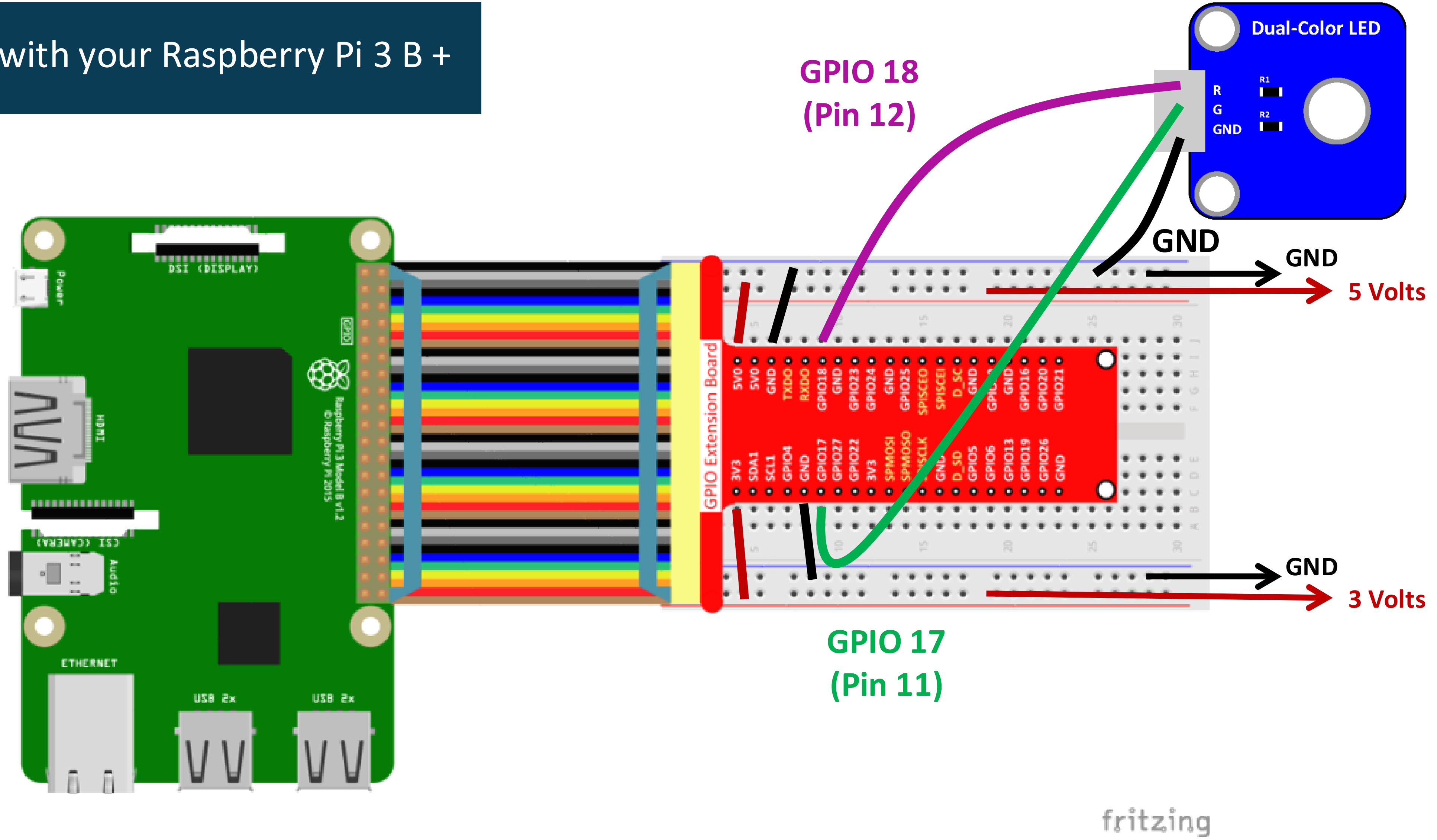
Source:
https://www.sunfounder.com/learn/lesson-1-dual-color-led-sensor-kit-v2-0-for-b.html

A dual-color light emitting diode (LED) is capable of emitting two different colors of light, typically red and green.

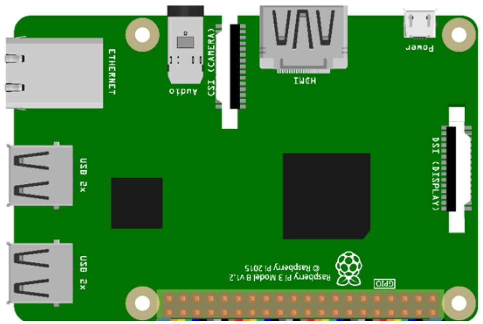
Application:
Variety of devices, such as televisions, digital cameras, and remote controls deploy these type LEDs.

Connector:
3-pin anti-reverse cable

Light up an LED with your Raspberry Pi 3 B +

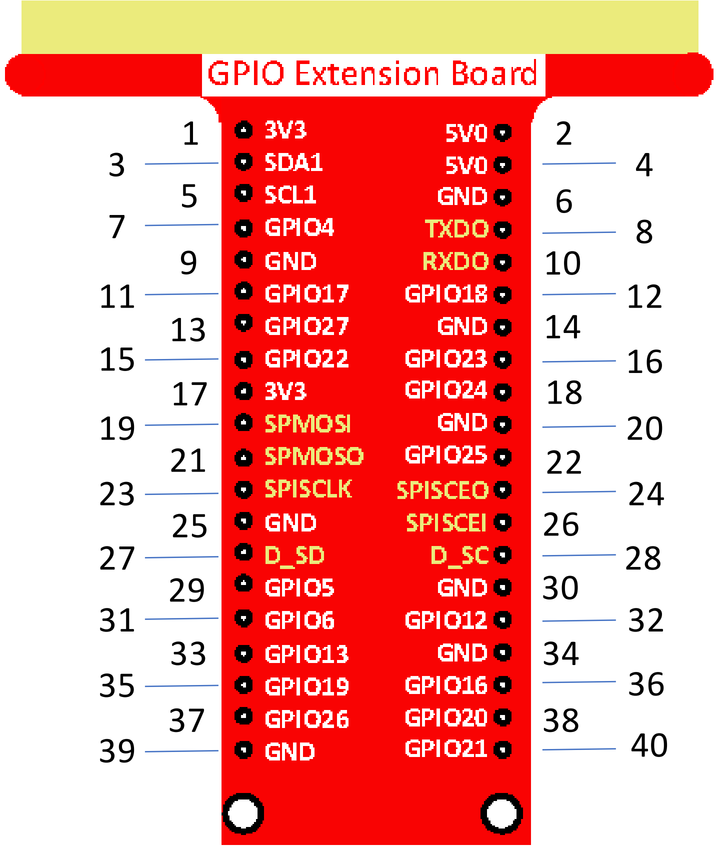


A simple python code to kick start your Raspberry Pi Model 3 B+ (RPi)



```
import RPi.GPIO as GPIO
import time
```

```
GPIO.setmode(GPIO.BOARD)
```

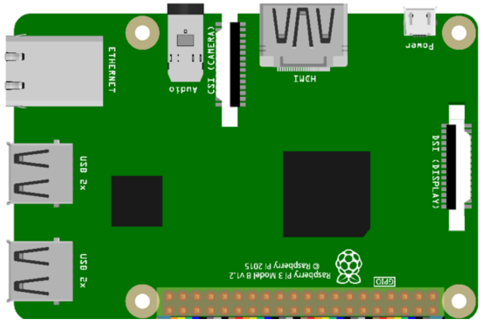


1	3V3	5V0	2
3	SDA1	5V0	4
5	SCL1	GND	6
7	GPIO4	TXDO	8
9	GND	RXDO	10
11	GPIO17	GPIO18	12
13	GPIO27	GND	14
15	GPIO22	GPIO23	16
17	3V3	GPIO24	18
19	SPMOSI	GND	20
21	SPMOSO	GPIO25	22
23	SPISCLK	SPISCEO	24
25	GND	SPISCEI	26
27	D_SD	D_SC	28
29	GPIO5	GND	30
31	GPIO6	GPIO12	32
33	GPIO13	GND	34
35	GPIO19	GPIO16	36
37	GPIO26	GPIO20	38
39	GND	GPIO21	40

```
GPIO.setup(12, GPIO.OUT)
```

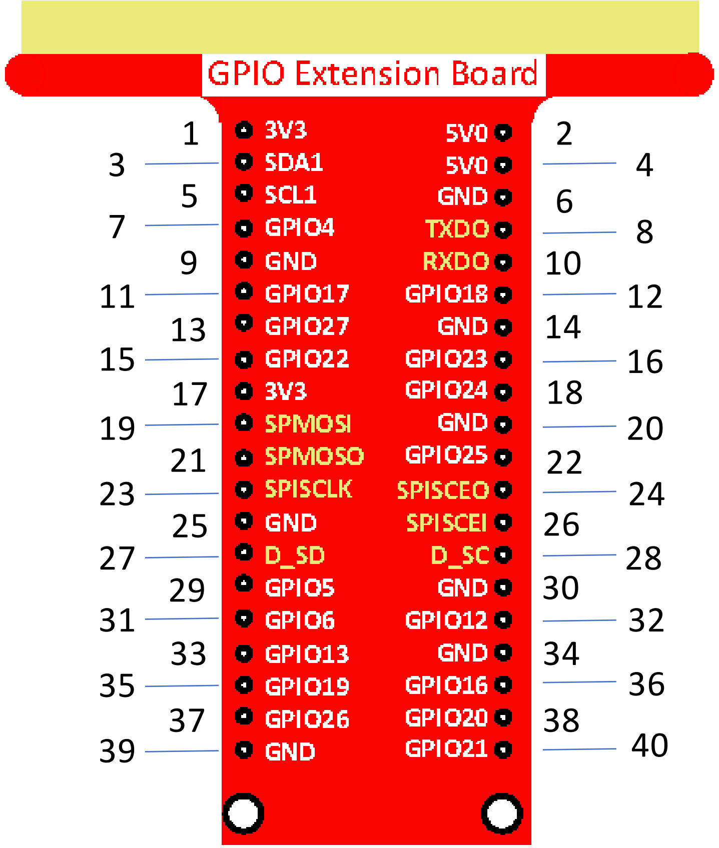
```
for i in range(0,15):

    GPIO.output(12, GPIO.HIGH)
    time.sleep(0.5)
    GPIO.output(12, GPIO.LOW)
    time.sleep(0.5)
    print(i)

GPIO.cleanup()
```

A simple python code to kick start your Raspberry Pi Model 3 B+ (RPi)



```
import RPi.GPIO as GPIO
import time
```

```
GPIO.setmode(GPIO.BOARD)
```



GPIO Extension Board

1	3V3	5V0	2
3	SDA1	5V0	4
5	SCL1	GND	6
7	GPIO4	TXDO	8
9	GND	RXDO	10
11	GPIO17	GPIO18	12
13	GPIO27	GND	14
15	GPIO22	GPIO23	16
17	3V3	GPIO24	18
19	SPMOSI	GND	20
21	SPMOSO	GPIO25	22
23	SPISCLK	SPISCEO	24
25	GND	SPISCEI	26
27	D_SD	D_SC	28
29	GPIO5	GND	30
31	GPIO6	GPIO12	32
33	GPIO13	GND	34
35	GPIO19	GPIO16	36
37	GPIO26	GPIO20	38
39	GND	GPIO21	40

```
GPIO.setup(12, GPIO.OUT)
```

```
def loop():
    while True:
        GPIO.output(12, GPIO.HIGH)
        time.sleep(0.5)
        GPIO.output(12, GPIO.LOW)
        time.sleep(0.5)
```

```
def destroy():
    GPIO.output(12, GPIO.LOW)
        # Turn off all leds
    GPIO.cleanup()
```

```
if __name__ == "__main__":
    try:
        loop()
    except KeyboardInterrupt:
        destroy()
```