

```
// 라즈베리파이 6일차,

// 스위치를 누를때마다 누른 횟수 누적하면서 엘이디 켜고 끄기.

import RPi.GPIO as GPIO

import time

led_pin = 7

pushButton_pin = 11

GPIO.setmode(GPIO.BCM)

GPIO.setwarnings(False)

GPIO.setup(led_pin, GPIO.OUT)

GPIO.setup(pushButton_pin, GPIO.IN)

cnt= 0

pressed = 0

try:

    while True:

        a = GPIO.input(pushButton_pin)

        if a == 1:

            GPIO.output(led_pin, True)

            if pressed == 0: # 기존 버튼상태가 0

                cnt = cnt + 1

                print(cnt)

            pressed = 1

        else:
```

```
GPIO.output(led_pin, False)

pressed = 0

except KeyboardInterrupt:

    pass

GPIO.cleanup()

// 버튼 하나로 엘이디 상태 토글 시키기.

import RPi.GPIO as GPIO

import time

led_pin = 7

pushButton_pin = 11

GPIO.setwarnings(False)

GPIO.setmode(GPIO.BCM)

GPIO.setup(led_pin, GPIO.OUT)

GPIO.setup(pushButton_pin, GPIO.IN)

led_on = False

pressed = 0

try:

    while True:

        a = GPIO.input(pushButton_pin)

        if a == 1:
```

```
        if pressed == 0:

            led_on = not led_on

            print(led_on)

            time.sleep(1)

            pressed = 1

        else:

            pressed = 0

        GPIO.output(led_pin, led_on)

except KeyboardInterrupt:

    pass

GPIO.cleanup()

// 버튼을 누를때마다 2초씩 엘이디 구동 시간 늘리기.

import RPi.GPIO as GPIO

import time

led_pin = 7

pushButton_pin = 11

GPIO.setwarnings(False)

GPIO.setmode(GPIO.BCM)

GPIO.setup(led_pin, GPIO.OUT)

GPIO.setup(pushButton_pin, GPIO.IN)
```

```
led_on = False
```

```
pressed = 0
```

```
start_time = 0
```

```
remain_time = 0
```

```
try:
```

```
    while True:
```

```
        a = GPIO.input(pushButton_pin)
```

```
        #print(a)
```

```
        #time.sleep(1)
```

```
        if(a==1) :
```

```
            GPIO.output(led_pin,1)
```

```
            remain_time=remain_time+2
```

```
            print('remain_time :',remain_time)
```

```
            time.sleep(remain_time)
```

```
        else :
```

```
            GPIO.output(led_pin,0)
```

```
except KeyboardInterrupt:
```

```
    pass
```

```
GPIO.cleanup()
```

```
// 피에조 스피커 사용,
```

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

```
import time
```

```
piezo_pin = 11

GPIO.setwarnings(False)

GPIO.setmode(GPIO.BCM)

GPIO.setup(piezo_pin, GPIO.OUT, initial=GPIO.LOW)
```

```
pi = GPIO.PWM(piezo_pin, 500)

pi.start(50)
```

```
try:
```

```
    while True:
```

```
        for i in range(100, 1000, 5):
```

```
            pi.ChangeFrequency(i)
```

```
            time.sleep(0.1)
```

```
        for i in range(1000, 100, -5):
```

```
            pi.ChangeFrequency(i)
```

```
            time.sleep(0.1)
```

```
except KeyboardInterrupt:
```

```
    pass
```

```
pi.stop()
```

```
GPIO.cleanup()
```

```
// 초음파 센서 사용. (아두이노의 계산 수식과는 다름)
```

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

```
import time
```

```
trig = 13
```

```
echo = 11
```

```
GPIO.setwarnings(False)
```

```
GPIO.setmode(GPIO.BCM)
```

```
GPIO.setup(echo, GPIO.IN)
```

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

```
GPIO.output(trig, GPIO.LOW)
```

```
def distance_check():
```

```
    GPIO.output(trig, GPIO.HIGH)
```

```
    time.sleep(0.00001)
```

```
    GPIO.output(trig, GPIO.LOW)
```

```
    stop = 0
```

```
    start = 0
```

```
    while GPIO.input(echo) == GPIO.LOW:
```

```
        start = time.time()
```

```
    while GPIO.input(echo) == GPIO.HIGH:
```

```
        stop = time.time()
```

```
    duration = stop - start
```

```
    distance = (duration*34300)/2
```

```
    return distance
```

try:

while True:

result_distance = round(distance_check(),2)

print("distance=%2f cm" % (result_distance))

time.sleep(1)

except KeyboardInterrupt:

GPIO.cleanup()

// 온습도 센서 사용.

pwd

/home/pi

git clone https://github.com/adafruit/Adafruit_Python_DHT.git

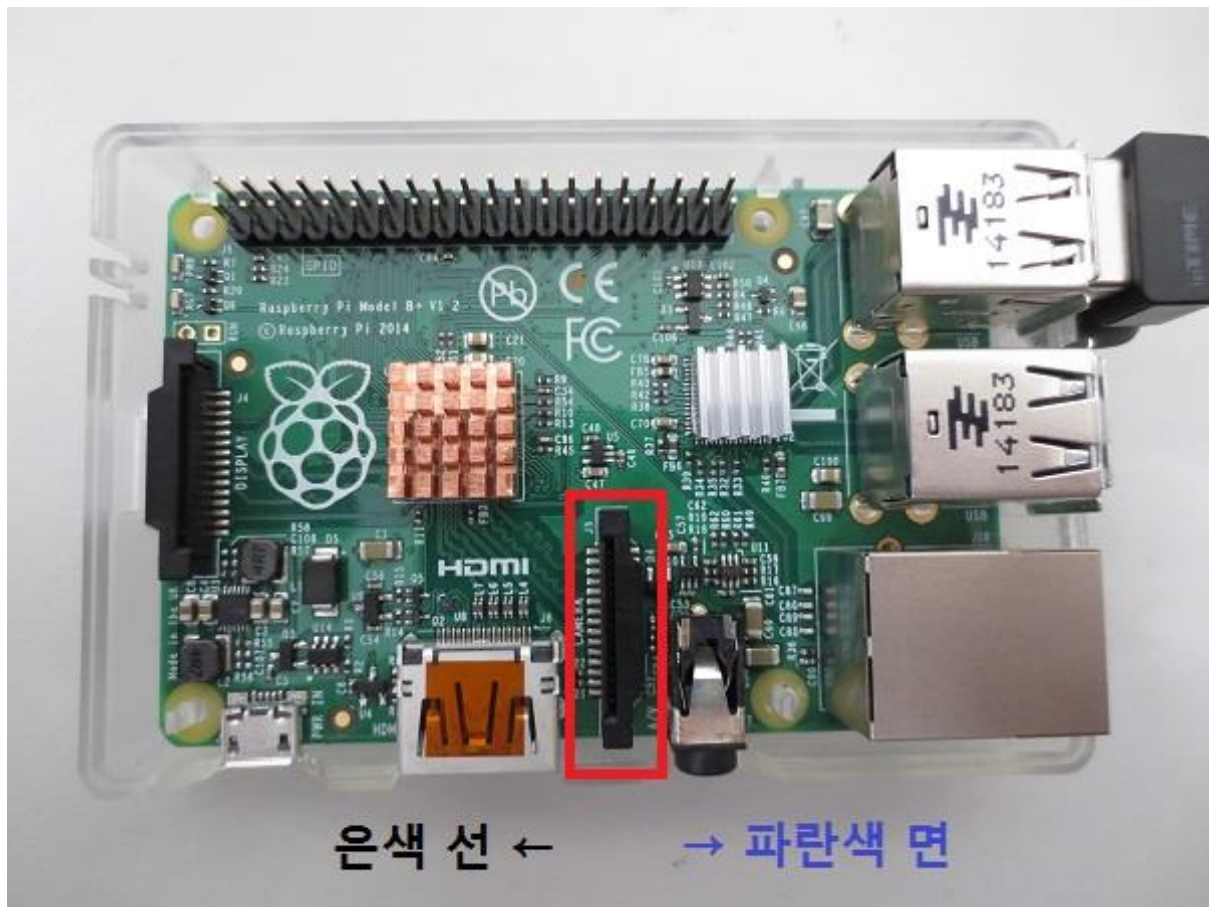
ls

cd Adafruit

sudo python3 setup.py install

```
Processing dependencies for Adafruit-DHT==1.4.0
Finished processing dependencies for Adafruit-DHT==1.4.0
pi@jaeho:~/Adafruit_Python_DHT $
```

// 카메라 모듈 사용 해보기.



```
from picamera import PiCamera
```

```
from time import sleep
```

```
camera = PiCamera()
```

```
camera.start_preview()
```

```
sleep(5)
```

```
camera.capture('/home/pi/Pictures/capture.jpg')
```

```
camera.stop_preview()
```

```
camera.close()
```

```
// 오류가 발생 된다면, 아래 수행.
```

```
sudo raspi-config
```


3 interface option 에서 카메라 활성화.

reboot

오류 없이 정상 구동.

// 주말 잘 보내시고요. 다음주에 이어서 살펴 볼게요.