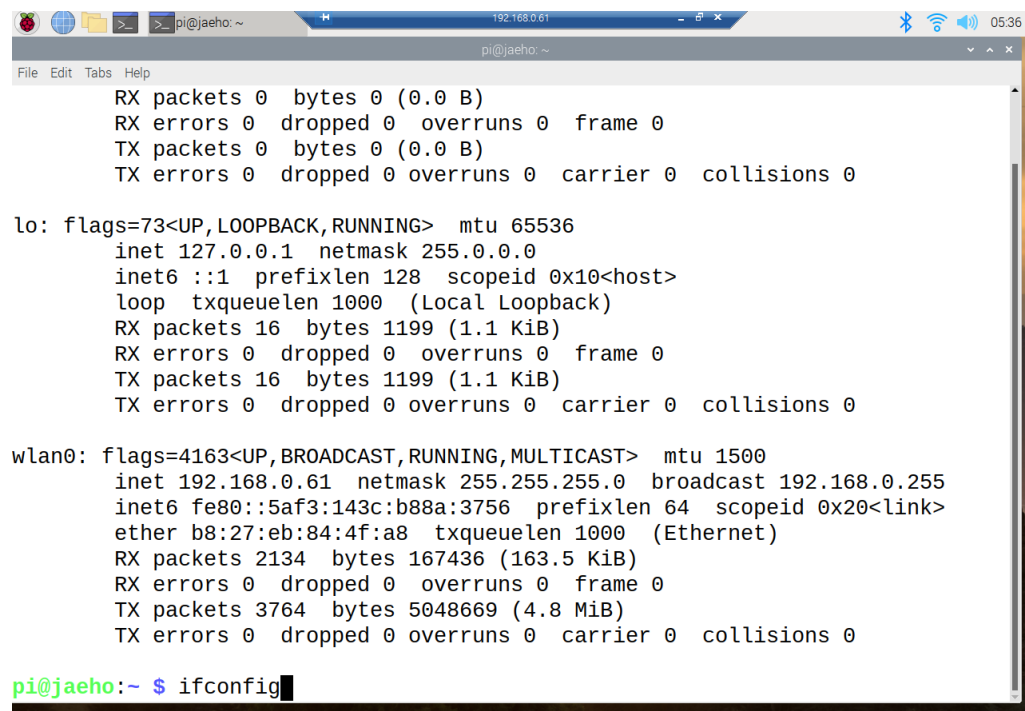


// 라즈베리파이 4일차,

플레스크를 이용한 wifi 자동차 리뷰하고,

리눅스 명령어와, 파이썬 기본 사용법 학습하기.

윈도우에서 **mstsc** 로 원격 접속.



```
pi@jaeho: ~  
File Edit Tabs Help  
RX packets 0 bytes 0 (0.0 B)  
RX errors 0 dropped 0 overruns 0 frame 0  
TX packets 0 bytes 0 (0.0 B)  
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
inet 127.0.0.1 netmask 255.0.0.0  
inet6 ::1 prefixlen 128 scopeid 0x10<host>  
loop txqueuelen 1000 (Local Loopback)  
RX packets 16 bytes 1199 (1.1 KiB)  
RX errors 0 dropped 0 overruns 0 frame 0  
TX packets 16 bytes 1199 (1.1 KiB)  
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
inet 192.168.0.61 netmask 255.255.255.0 broadcast 192.168.0.255  
inet6 fe80::5af3:143c:b88a:3756 prefixlen 64 scopeid 0x20<link>  
ether b8:27:eb:84:4f:a8 txqueuelen 1000 (Ethernet)  
RX packets 2134 bytes 167436 (163.5 KiB)  
RX errors 0 dropped 0 overruns 0 frame 0  
TX packets 3764 bytes 5048669 (4.8 MiB)  
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
pi@jaeho:~ $ ifconfig
```

clear 화면 정리.

// 접속 계정명 변경 :

sudo su

접속 계정이 최고관리자로 변경.

raspi-config

1 >> 3 hostName 변경.

// 파일 복사하기 cp

cd Doc 하고 탭

```
touch 111.txt
```

```
ls
```

```
vi 111.txt
```

```
cat 111.txt
```

```
cp 111.txt 222.txt
```

```
ls
```

```
cat 222.txt
```

```
mkdir
```

```
rm -rf 폴더명.
```

```
// 파이썬 문법
```

```
a=10
```

```
b=3.5
```

```
c='mike'
```

```
print(a)
```

```
print(b)
```

```
print(c)
```

```
// 파이썬 문법2
```

```
a=10
```

```
b=40
```

```
sum=a+b
```

```
print(sum)
```

```
// 파이썬 자습서
```

<https://docs.python.org/ko/3/tutorial/index.html>

```
// 나누기 처리.
```

```
a=40
```

```
b=10
```

```
sum=a//b
```

```
print(sum)
```

```
// 파이썬 문법4 (리스트)
```

```
names=['mke','jane','tom'] #인덱스 시작은 0 부터.
```

```
print(names)
```

```
print(names[0])
```

```
print(names[2])
```

```
// 문법5
```

```
user_data=[20,'jane',True]
```

```
print(user_data)
```

```
print(user_data[0])
```

```
print(user_data[2])
```

```
// 딕셔너리 (맵이나 json 타입과 비슷)
```

```
user={'age':25, 'address':'gwangju'}
```

```
user['age']=25
```

```
print(user)
```

```
0228.py x
1 a='Good morning'
2 a.upper()
3 print(a)
4 print(a.upper())

Shell
>>> %Run 0228.py
Good morning
GOOD MORNING
>>>
```

// 내장 함수

```
print(int('123'))
```

```
print(float('354.1'))
```

```
print(str(123))
```

```
print(len('hi'))
```

// 성적 출력 (들여쓰기 중요)

```
jumsu=input('input number:')
```

```
jumsu=int(jumsu)//10
```

```
if jumsu>=9:
```

```
    print('A')
```

```
elif jumsu>=8:
```

```
    print('B')
```

```
elif jumsu>=7:
```

```
    print('C')
```

```
elif jumsu>=6:
```

```
print('D')
```

else :

```
print('F')
```

```
1 jumsu=input('input number:')
2 jumsu=int(jumsu)
3
4
5 if jumsu>=90:
6     print('A',end='')
7     if jumsu>=95:
8         print('+')
9         print('~')
10    print('^^')
```

Shell

```
>>> %Run 0228.py
input number:90
A^^
```

```
jumsu=input('input number:')
```

```
jumsu=int(jumsu)
```

```
if jumsu>=90:
```

```
    print('A',end='')
```

```
    if jumsu>=95:
```

```
        print('+')
```

```
        print('~')
```

```
    print('^^')
```

// while 을 이용한 1~10 출력.

```
1 a=0
2 while a<10:
3     a=a+1
4     print(a,end=' ')
```

Shell

```
>>> %Run 0228.py
1, 2, 3, 4, 5, 6, 7, 8, 9, 10,
^^^
```

a=0

while a<10:

a=a+1

print(a,end=' ')

// for 를 이용하여 짝수만 출력.

number=[1,2,3,4,5,6,7]

for num in number:

if num%2==0:

print(num, end=' ')

// 렌지 (범위 이용, 0부터)

for i in range(10):

print(i, end=' ')

```
1 for i in range(10):
2     print(i, end=' ')
```

Shell

```
>>> %Run 0228.py
0 1 2 3 4 5 6 7 8 9
```

// 구구단 2단에서 9단까지 이중 for 를 이용하여 출력.

```
for x in range(2, 10):
```

```
    print("---[" + str(x) + "dan]---")
```

```
        for y in range(1, 10):
```

```
            print(x, "X", y, "=", x*y)
```

```
print("-----")
```

// 버튼을 누를때만 led 켜지게 하기.

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

```
import time
```

```
led_pin = 7
```

```
pushButton_pin = 11
```

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

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

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

```
try:
```

```
    while True:
```

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

```
        if a == 0:
```

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

```
        else:
```

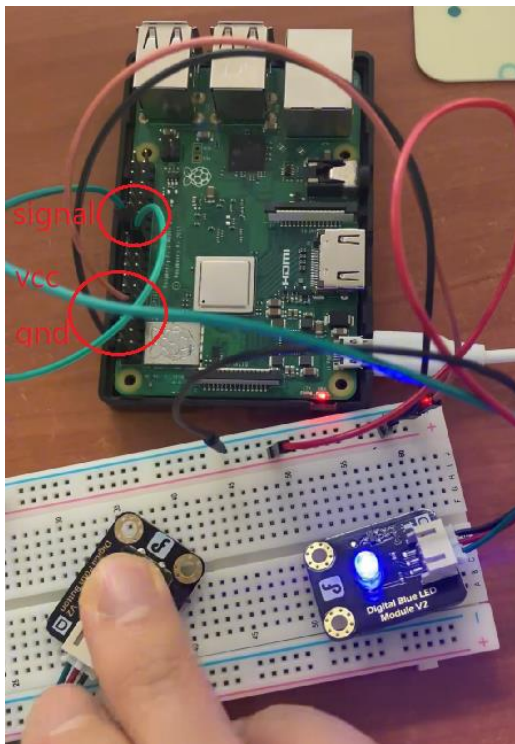
```
            GPIO.output(led_pin, True)
```

```
        print(a)
```

```
except KeyboardInterrupt:
```

```
    pass
```

```
GPIO.cleanup()
```



```
// 도전
```

```
버튼 눌린 횟수 표시하기.
```

```
토글 버튼으로 상태 유지하기.
```

```
버튼을 누를때마다 켜지는 시간을 10초씩 늘리기.
```

```
// 3월 1일 잘 쉬시고요, 3월 2일은 제가 하루 쉽니다.
```

```
// 다른 강사님과 잘 공부하시기 바래요.
```

```
// 금요일에 다시 만나요~
```