Jetson Nano GPIO

조 영 혁

노다시스템

Jetson Nano GPIO

1. Get GPIO source from Git-Hub

Python 설치

```
$ sudo add-apt-repository ppa:fkrull/deadsnakes
$ sudo apt-get update
$ sudo apt-get install python3.6
```

Python버젼을 변경

```
$  alias python=‘/usr/bin/python3’
```

Python pip install

```
$ sudo apt-get install python3-pip
$ pip3 install numpy
```

pip 설치

numpy lib 설치

Jetson Nano GPIO

2. Python 셋팅 및 환경 설정

Jetson GPIO lib 설치 및 권한 부여

```
$ git clone https://github.com/NVIDIA/jetson-gpio.git
$ cd jetson-gpio
$ sudo python3 setup.py install
```

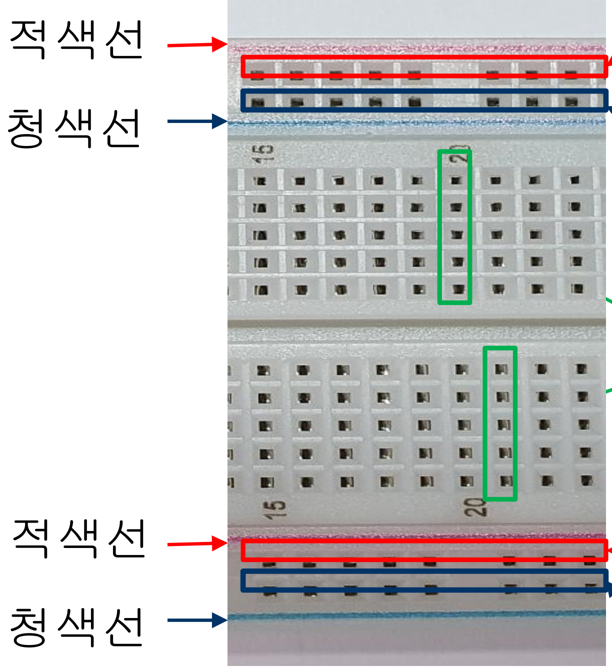
```
$ sudo groupadd –f –r gpio
$ sudo usermod–a –G gpio  <user_id>
```

gpio를 그룹으로 추가
gipo 그룹에 자신의 ID 등록

```
$ sudo cp lib/python/Jetson/GPIO/99-gpio.rules /etc/udev/rules.d/
$ sudo reboot                : 재부팅
  … … ..
$ sudo udevadm control --reload-rules && sudo udevadm trigger
```

3. LED 회로 보드

- 브레드 보드



가로 방향으로 한줄이 모두 연결되어 있음
직류전원의 +전원선(+Vcc)으로 사용

가로 방향으로 한줄이 모두 연결되어 있음
직류전원의 GND선(0V)으로 사용

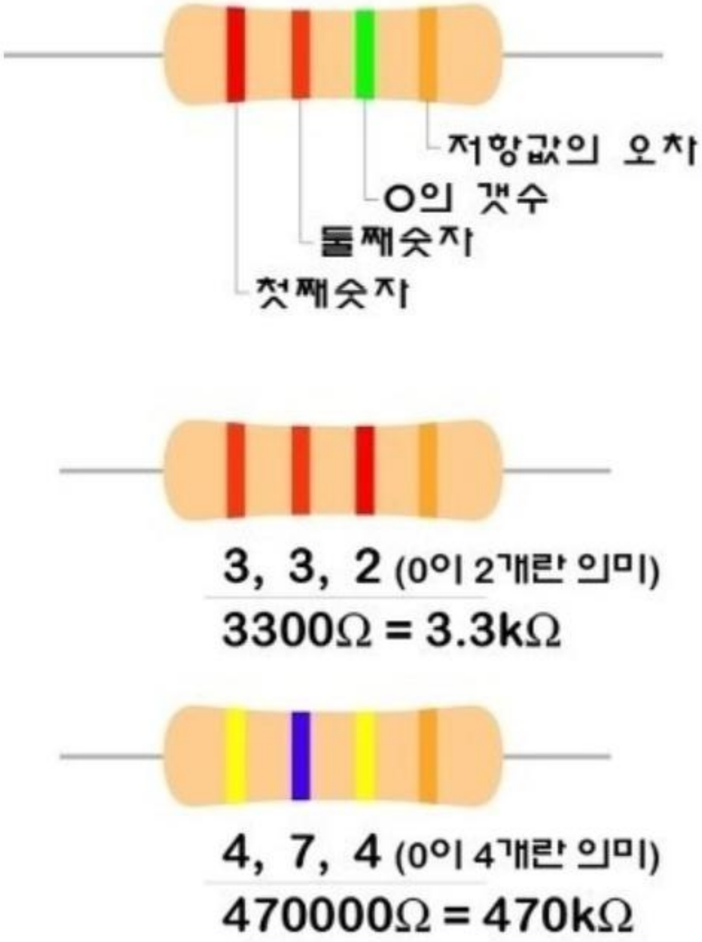
세로 방향으로 5개씩 내부로 연결되어 있음

가로 방향으로 한줄이 모두 연결되어 있음
직류전원의 +전원선(+Vcc)으로 사용

가로 방향으로 한줄이 모두 연결되어 있음
직류전원의 GND선(0V)으로 사용

Jetson Nano GPIO

- 4색 띠 저항기

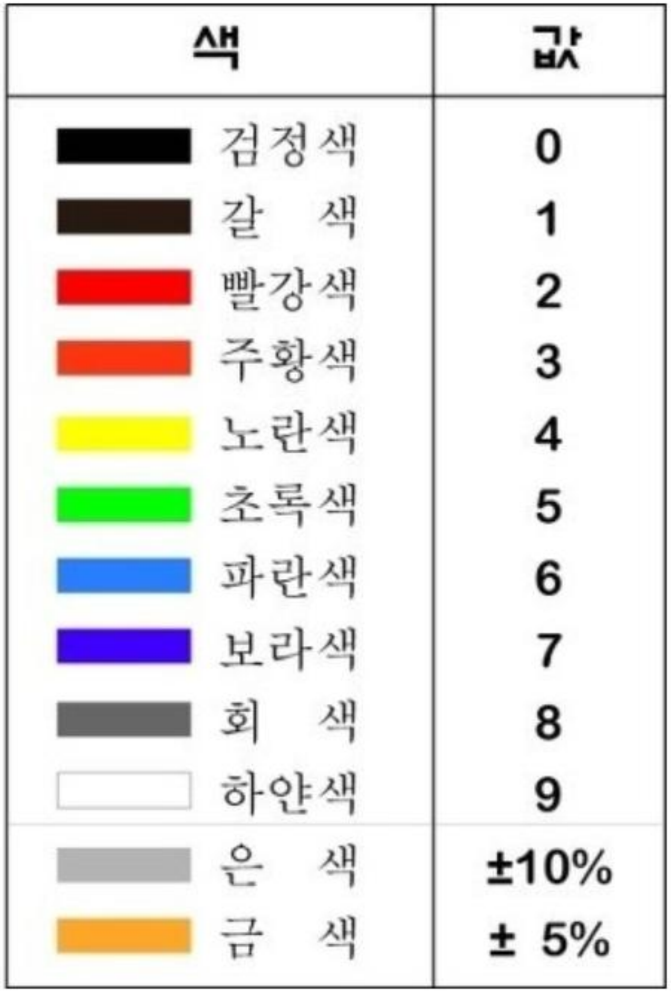


3, 3, 2 (0이 2개란 의미)

3300Ω = 3.3kΩ

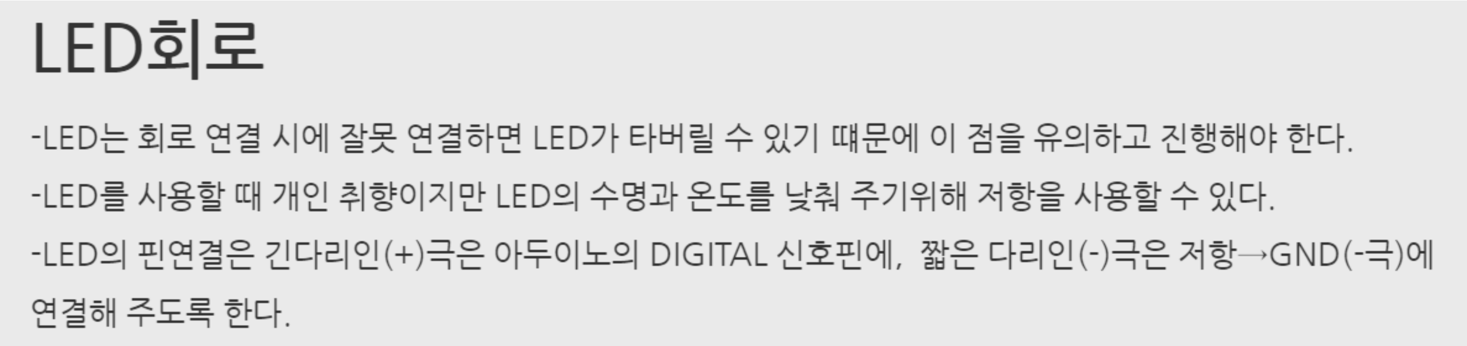
4, 7, 4 (0이 4개란 의미)

470000Ω = 470kΩ



색	값
⬛ 검정색	0
🟫 갈 색	1
🟥 빨강색	2
🟧 주황색	3
🟨 노란색	4
🟩 초록색	5
🟦 파란색	6
🟪 보라색	7
▩ 회 색	8
⬜ 하얀색	9
▨ 은 색	±10%
🟧 금 색	± 5%

Jetson Nano GPIO

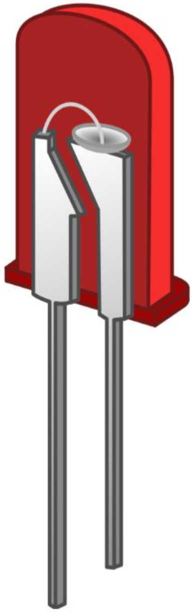


LED회로

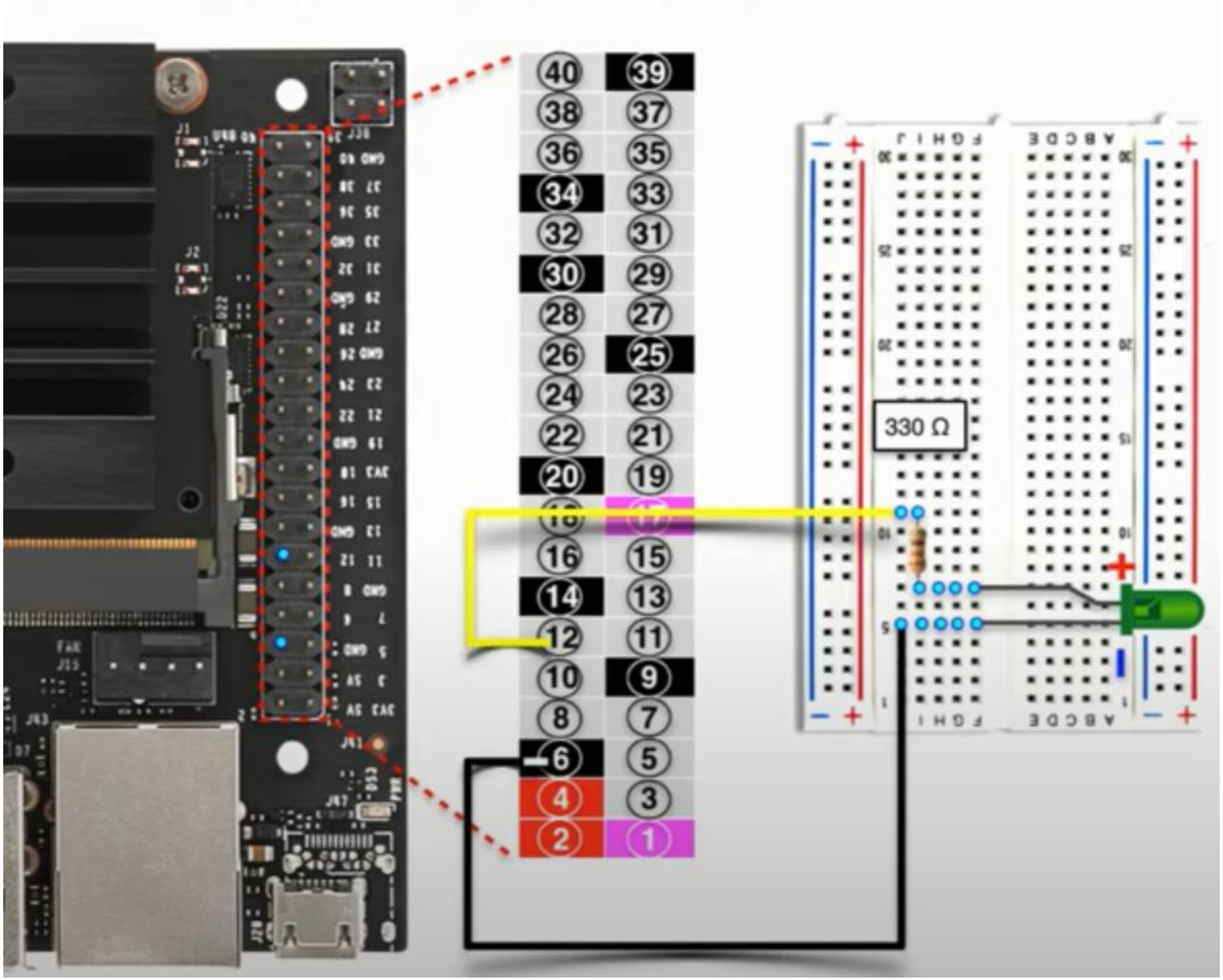
-LED는 회로 연결 시에 잘못 연결하면 LED가 타버릴 수 있기 때문에 이 점을 유의하고 진행해야 한다.

-LED를 사용할 때 개인 취향이지만 LED의 수명과 온도를 낮춰 주기위해 저항을 사용할 수 있다.

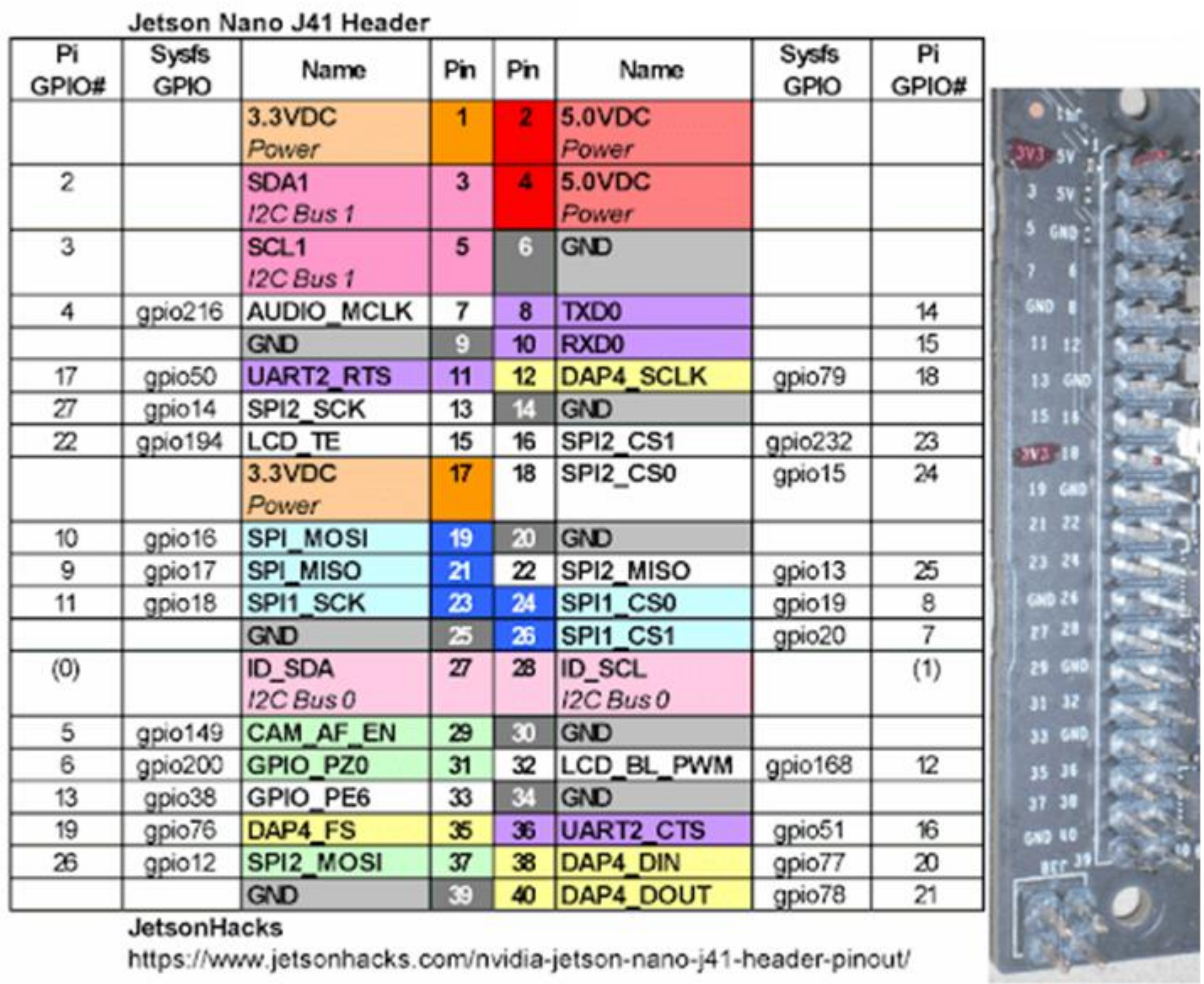
-LED의 핀연결은 긴다리인(+)극은 아두이노의 DIGITAL 신호핀에, 짧은 다리인(-)극은 저항→GND(-극)에 연결해 주도록 한다.



Jetson Nano GPIO



Jetson Nano GPIO



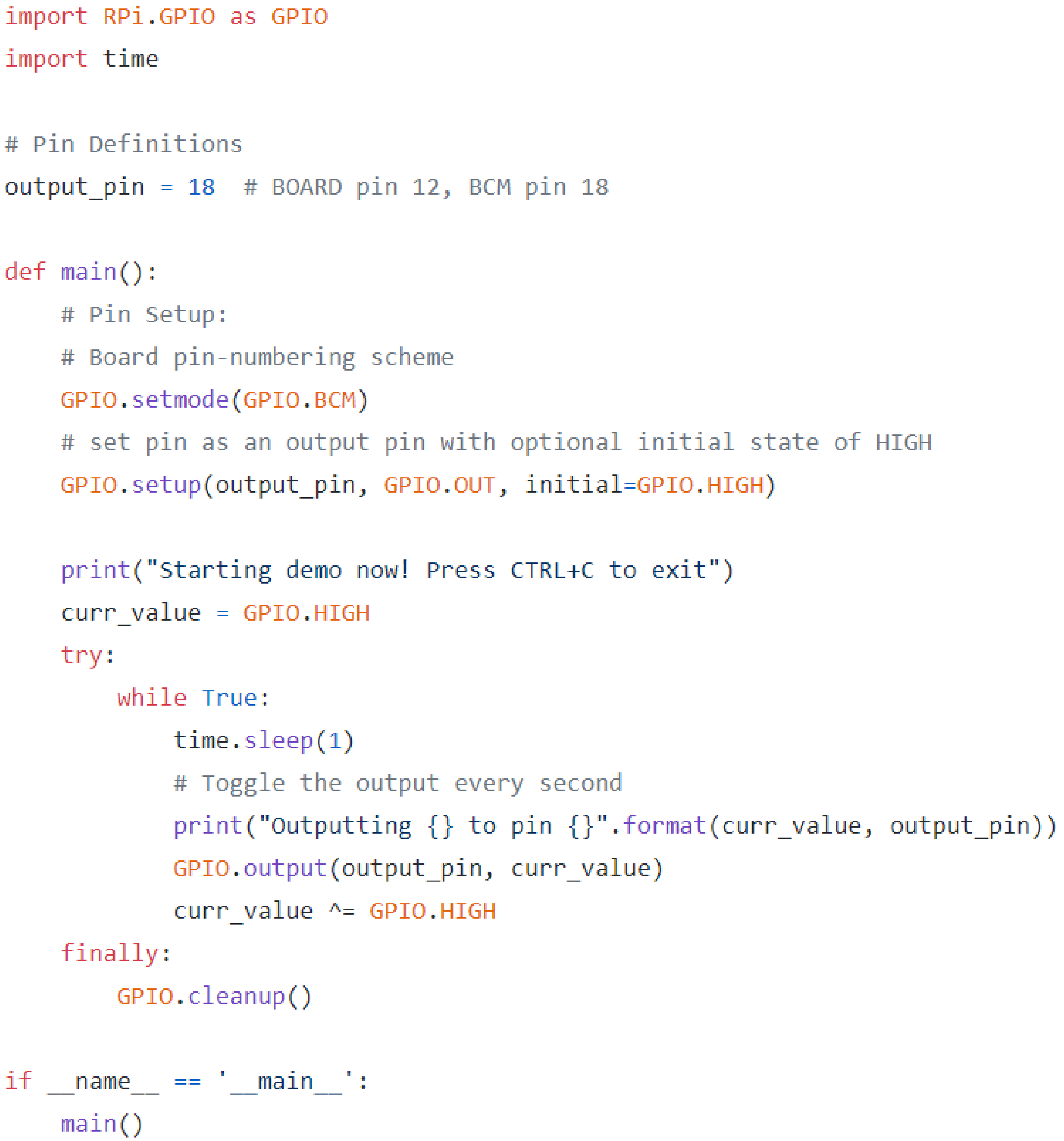
Jetson Nano J41 Header

Pi GPIO#	Sysfs GPIO	Name	Pin	Pin	Name	Sysfs GPIO	Pi GPIO#
		3.3VDC *Power*	1	2	5.0VDC *Power*		
2		SDA1 *I2C Bus 1*	3	4	5.0VDC *Power*		
3		SCL1 *I2C Bus 1*	5	6	GND		
4	gpio216	AUDIO_MCLK	7	8	TXD0		14
		GND	9	10	RXD0		15
17	gpio50	UART2_RTS	11	12	DAP4_SCLK	gpio79	18
27	gpio14	SPI2_SCK	13	14	GND		
22	gpio194	LCD_TE	15	16	SPI2_CS1	gpio232	23
		3.3VDC *Power*	17	18	SPI2_CS0	gpio15	24
10	gpio16	SPI_MOSI	19	20	GND		
9	gpio17	SPI_MISO	21	22	SPI2_MISO	gpio13	25
11	gpio18	SPI1_SCK	23	24	SPI1_CS0	gpio19	8
		GND	25	26	SPI1_CS1	gpio20	7
(0)		ID_SDA *I2C Bus 0*	27	28	ID_SCL *I2C Bus 0*		(1)
5	gpio149	CAM_AF_EN	29	30	GND		
6	gpio200	GPIO_PZ0	31	32	LCD_BL_PWM	gpio168	12
13	gpio38	GPIO_PE6	33	34	GND		
19	gpio76	DAP4_FS	35	36	UART2_CTS	gpio51	16
26	gpio12	SPI2_MOSI	37	38	DAP4_DIN	gpio77	20
		GND	39	40	DAP4_DOUT	gpio78	21

JetsonHacks
https://www.jetsonhacks.com/nvidia-jetson-nano-j41-header-pinout/

Jetson Nano GPIO

#!/usr/bin/env python



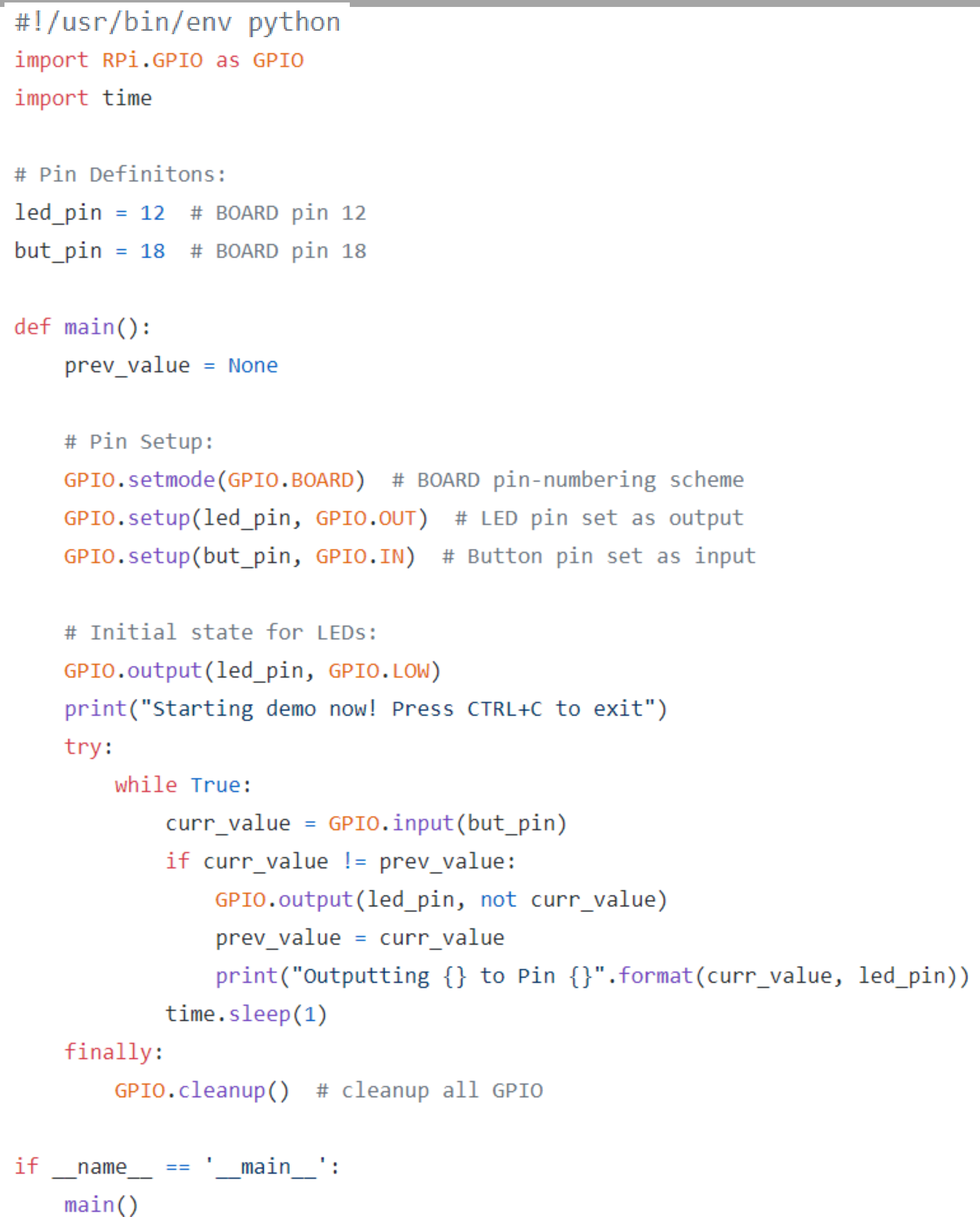
```python
import RPi.GPIO as GPIO
import time

# Pin Definitions
output_pin = 18  # BOARD pin 12, BCM pin 18

def main():
    # Pin Setup:
    # Board pin-numbering scheme
    GPIO.setmode(GPIO.BCM)
    # set pin as an output pin with optional initial state of HIGH
    GPIO.setup(output_pin, GPIO.OUT, initial=GPIO.HIGH)

    print("Starting demo now! Press CTRL+C to exit")
    curr_value = GPIO.HIGH
    try:
        while True:
            time.sleep(1)
            # Toggle the output every second
            print("Outputting {} to pin {}".format(curr_value, output_pin))
            GPIO.output(output_pin, curr_value)
            curr_value ^= GPIO.HIGH
    finally:
        GPIO.cleanup()

if __name__ == '__main__':
    main()
```

Jetson Nano GPIO

- button Pin = Board 12
- LED Pin = Board 18

-버튼을 누르는 동안 LED On

Jetson Nano GPIO



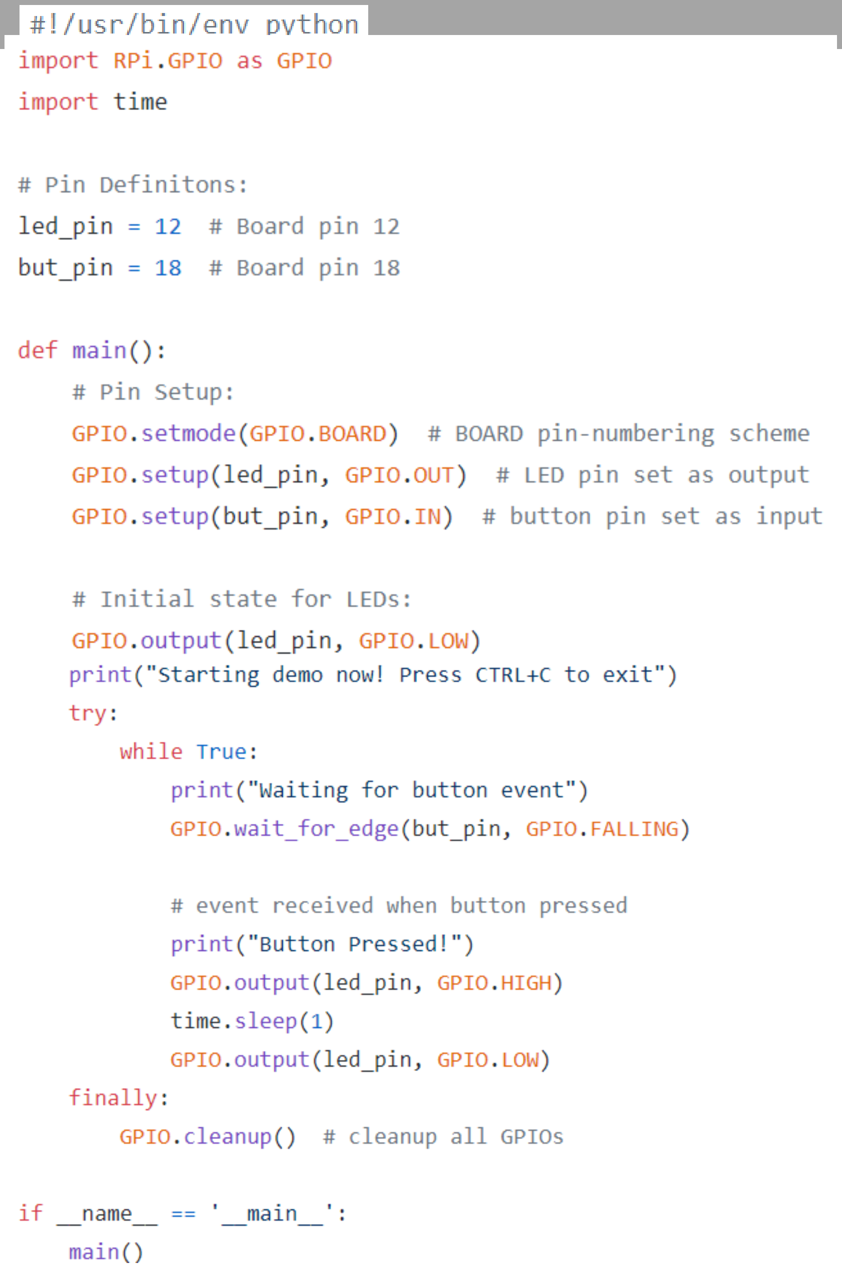
```
#!/usr/bin/env python
import RPi.GPIO as GPIO
import time

# Pin Definitons:
led_pin = 12  # BOARD pin 12
but_pin = 18  # BOARD pin 18

def main():
    prev_value = None

    # Pin Setup:
    GPIO.setmode(GPIO.BOARD)  # BOARD pin-numbering scheme
    GPIO.setup(led_pin, GPIO.OUT)  # LED pin set as output
    GPIO.setup(but_pin, GPIO.IN)  # Button pin set as input

    # Initial state for LEDs:
    GPIO.output(led_pin, GPIO.LOW)
    print("Starting demo now! Press CTRL+C to exit")
    try:
        while True:
            curr_value = GPIO.input(but_pin)
            if curr_value != prev_value:
                GPIO.output(led_pin, not curr_value)
                prev_value = curr_value
                print("Outputting {} to Pin {}".format(curr_value, led_pin))
            time.sleep(1)
    finally:
        GPIO.cleanup()  # cleanup all GPIO

if __name__ == '__main__':
    main()
```

Jetson Nano GPIO

- button Pin = Board 12
- LED Pin = Board 18

-첫번째 버튼에 LED 1번 깜빡임
-두번재 버튼에 LED 3번 깜빡임
-세번째 버튼에 LED 5번 깜박임
-……..
- …….
- 길게 누르면 초기 리셋

Jetson Nano GPIO



```python
#!/usr/bin/env python
import RPi.GPIO as GPIO
import time

# Pin Definitons:
led_pin = 12  # Board pin 12
but_pin = 18  # Board pin 18

def main():
    # Pin Setup:
    GPIO.setmode(GPIO.BOARD)  # BOARD pin-numbering scheme
    GPIO.setup(led_pin, GPIO.OUT)  # LED pin set as output
    GPIO.setup(but_pin, GPIO.IN)  # button pin set as input

    # Initial state for LEDs:
    GPIO.output(led_pin, GPIO.LOW)
    print("Starting demo now! Press CTRL+C to exit")
    try:
        while True:
            print("Waiting for button event")
            GPIO.wait_for_edge(but_pin, GPIO.FALLING)

            # event received when button pressed
            print("Button Pressed!")
            GPIO.output(led_pin, GPIO.HIGH)
            time.sleep(1)
            GPIO.output(led_pin, GPIO.LOW)
    finally:
        GPIO.cleanup()  # cleanup all GPIOs

if __name__ == '__main__':
    main()
```

THANK YOU

Suggestions Questions