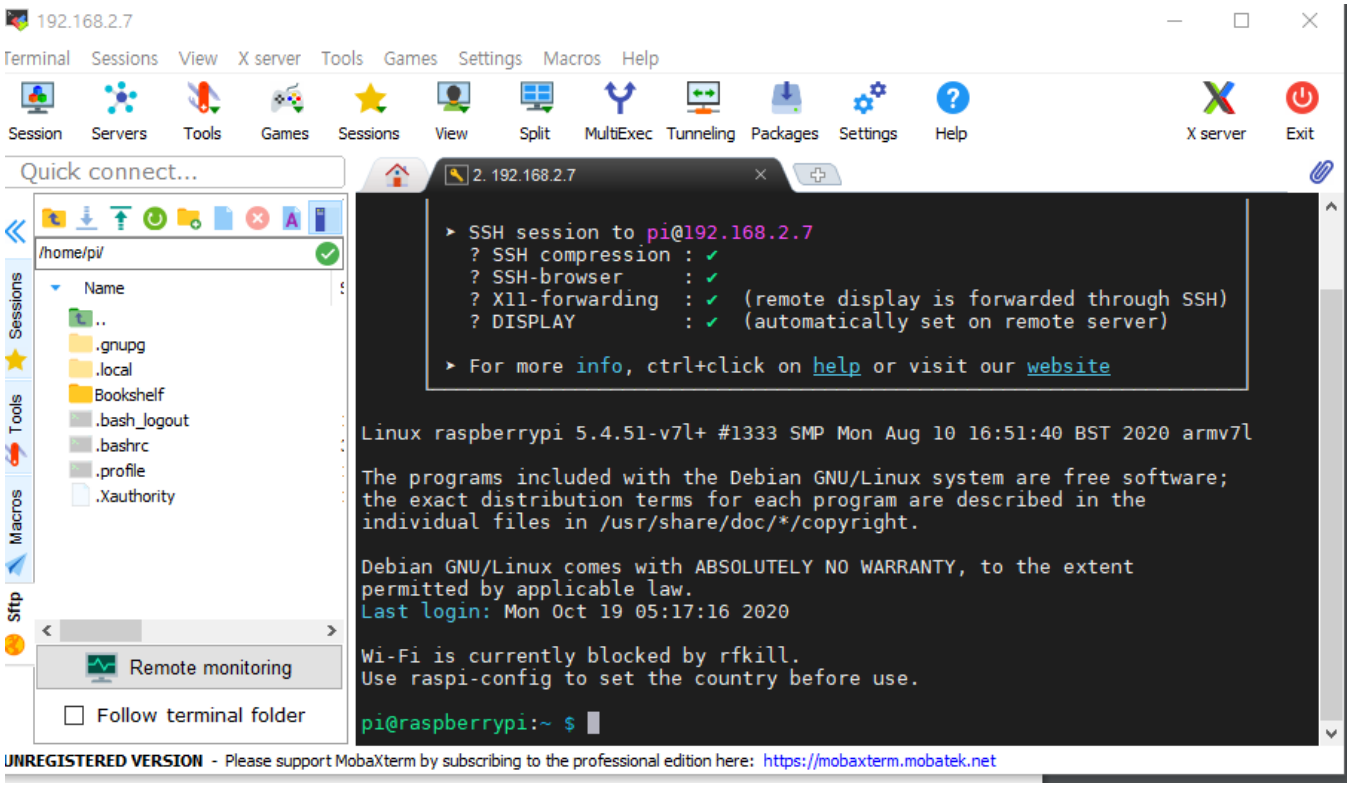
지능형IoT네트워크
라즈베리파이 설정 및 기초 제어

충북대학교

2020. 11. 12.

원격 접속

❒ MobaXterm에서 라즈베리파이 접속



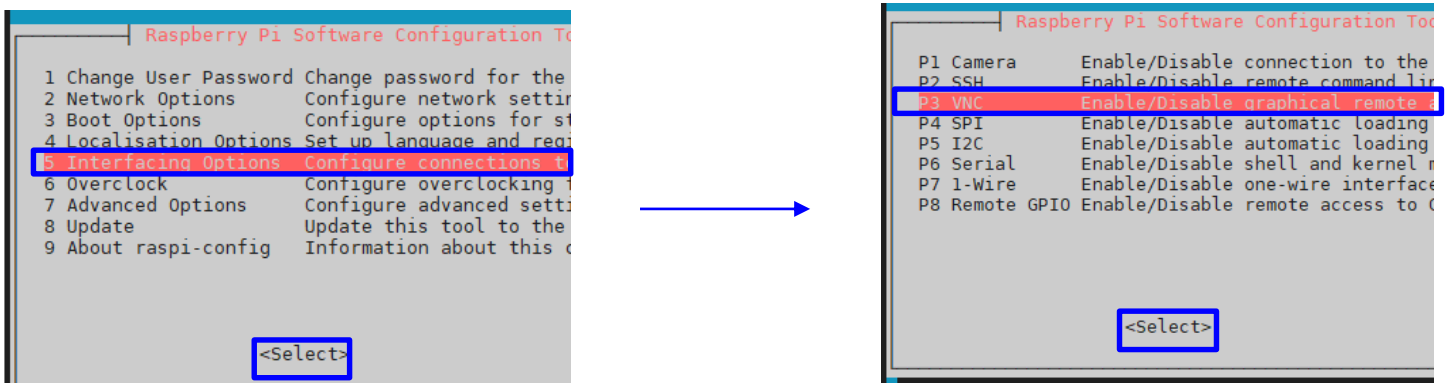
원격 접속

❐ 원격 접속 설정

- 터미널에서 ‘sudo raspi-config’ 입력

```
pi@raspberrypi:~ $ sudo raspi-config
```

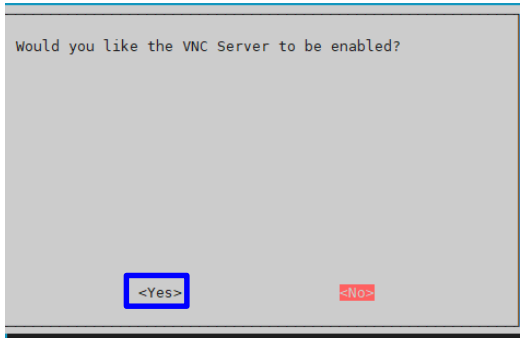
- ‘Interfacing Options’ -> ‘VNC’



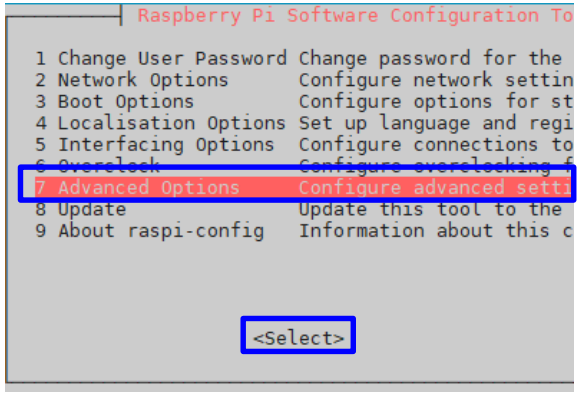
원격 접속

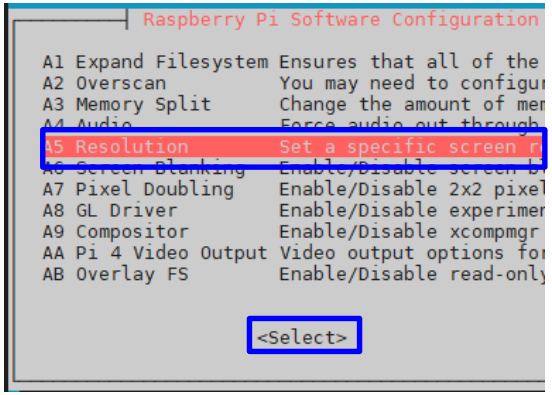
❐ 원격 접속 설정

- <Yes> 선택



- ‘Advanced Options’ -> ‘Resolution’ 선택

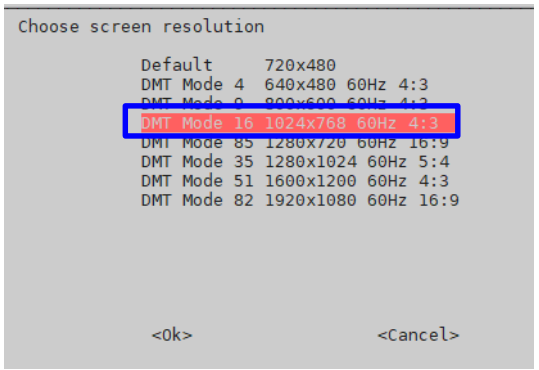




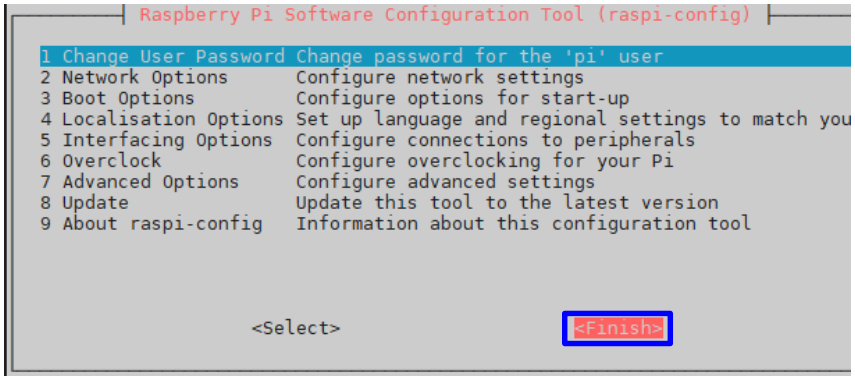
원격 접속

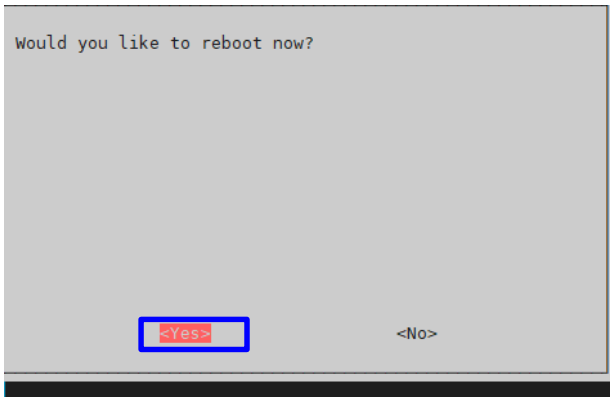
❐ 원격 접속 설정

- ‘DMT Mode 16 1024x768 60Hz 4:3’ 선택



- ‘<Finish>’ 선택 및 재부팅

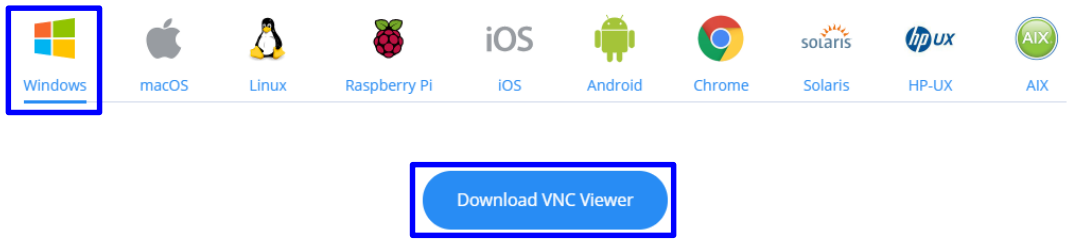




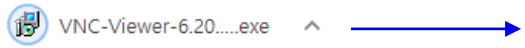
원격 접속

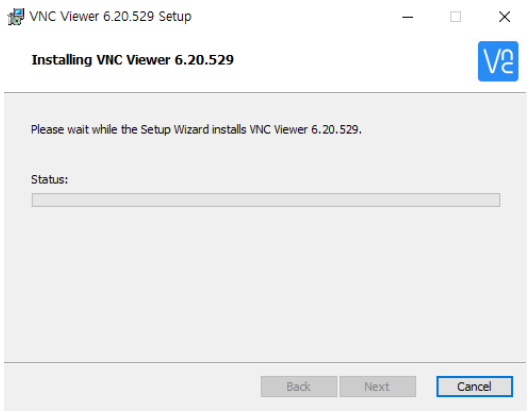
❒ VNC 설치

- 다음 주소에서 VNC 설치
- https://www.realvnc.com/en/connect/download/viewer/



- VNC 설치 진행



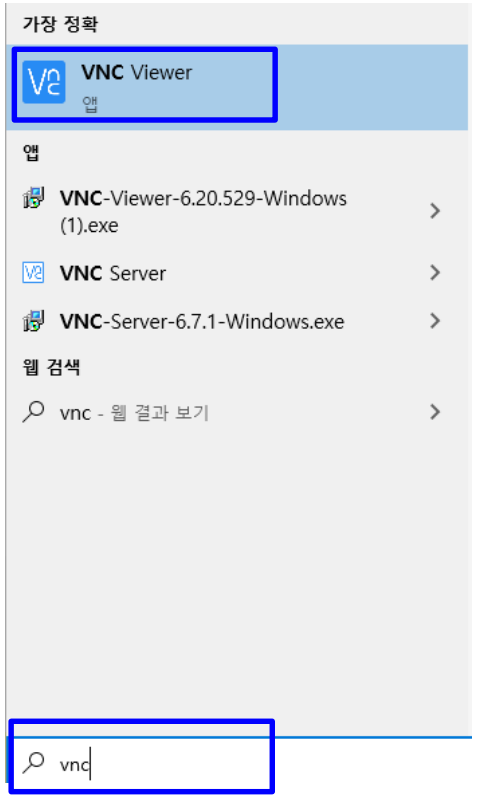


원격 접속

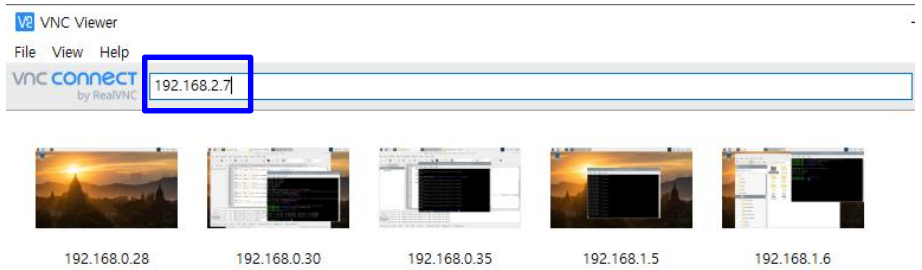
❐ VNC 접속

- VNC Viewer 실행 및 라즈베리파이 접속

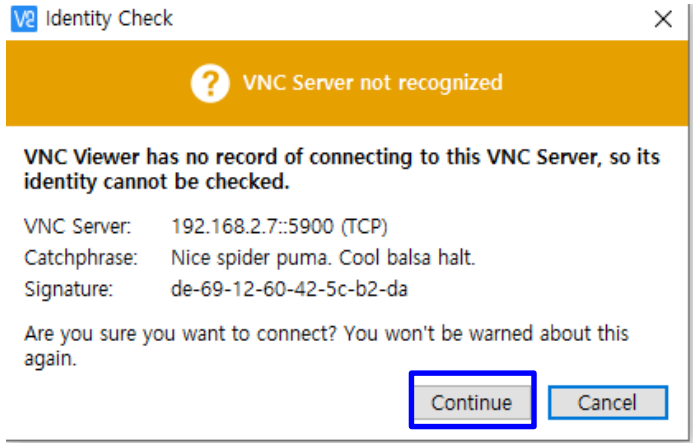
1.



2.

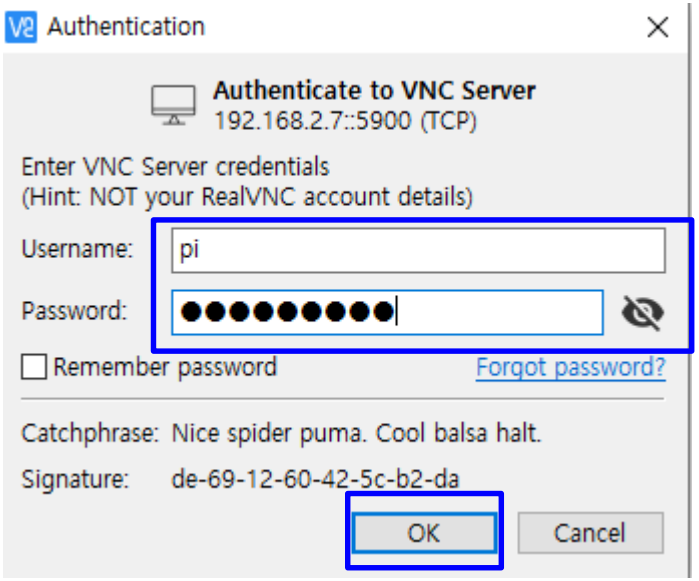


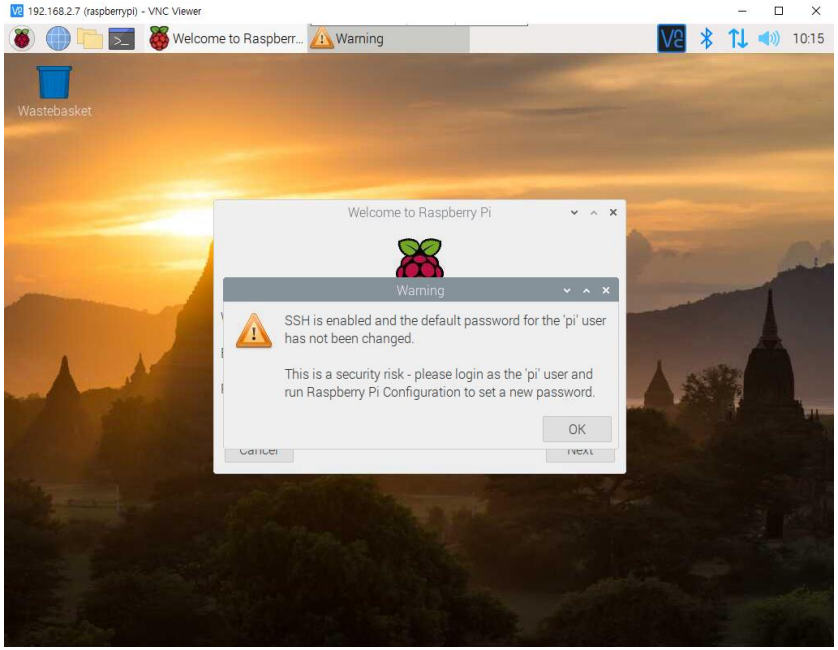
3.



❒ VNC 접속

- 라즈베리파이 접속
  - ID: pi
  - PW: raspberry



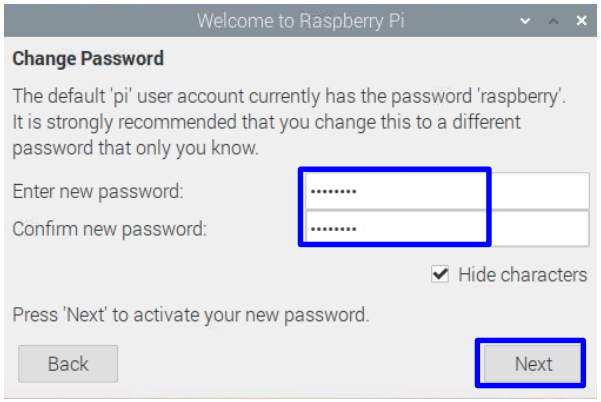


라즈베리파이 설정

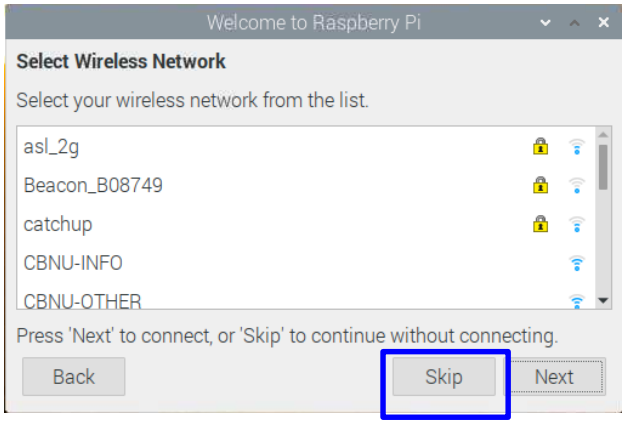
❐ 설정 진행

1.

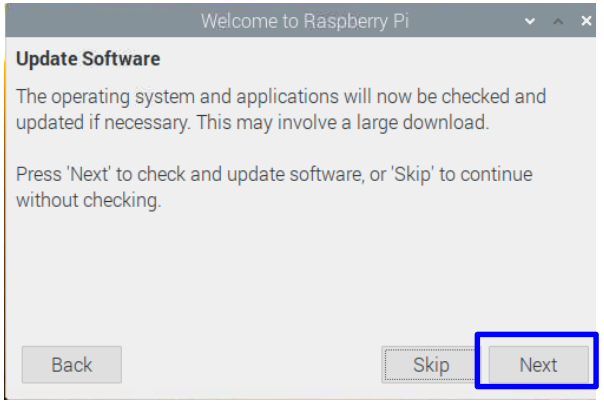
2.



3.



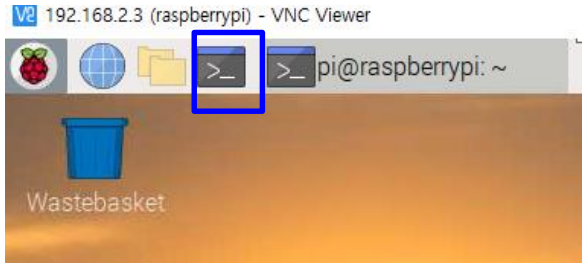
4.



라즈베리파이 설정

❐ 수동 업데이트 진행

- 터미널 접속
  - 우측 상단에서 접속 가능



  - ‘sudo apt-get update’ 입력

```
pi@raspberrypi:~ $ sudo apt-get update
```

  - ‘sudo apt-get upgrade’ 입력

```
pi@raspberrypi:~ $ sudo apt-get upgrade
```

LED 테스트

❐ 소스코드 작성

- 터미널 실행 및 다음 명령어 입력

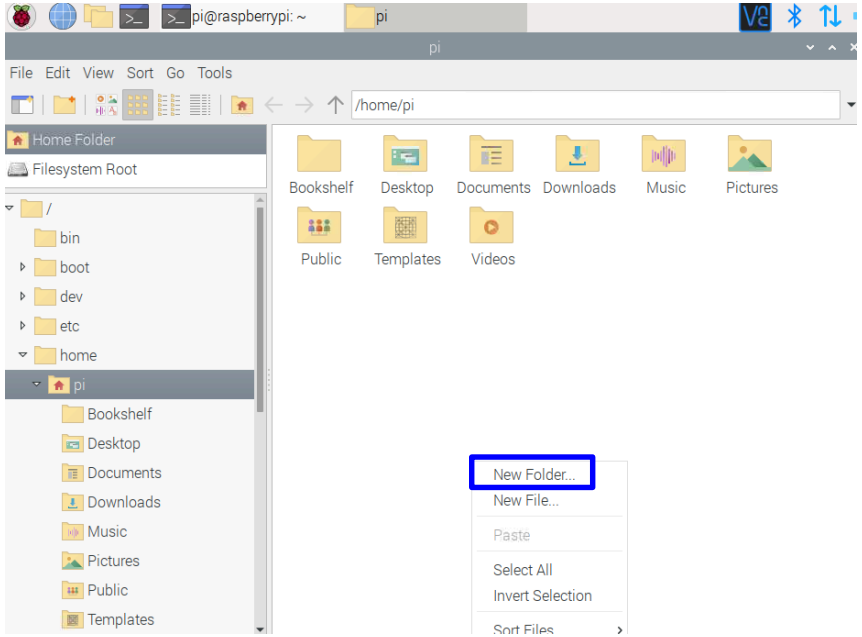
```
pi@raspberrypi:~ $ sudo wget https://project-downloads.drogon.net/wiringpi-lates
t.deb
pi@raspberrypi:~ $ sudo dpkg -i wiringpi-latest.deb
```

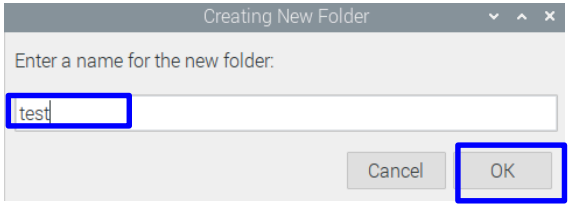
LED 테스트

❐ 테스트 디렉터리 생성

- 좌측 상단 디렉터리 아이콘 클릭

- 우클릭 후 폴더 생성

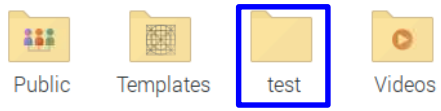




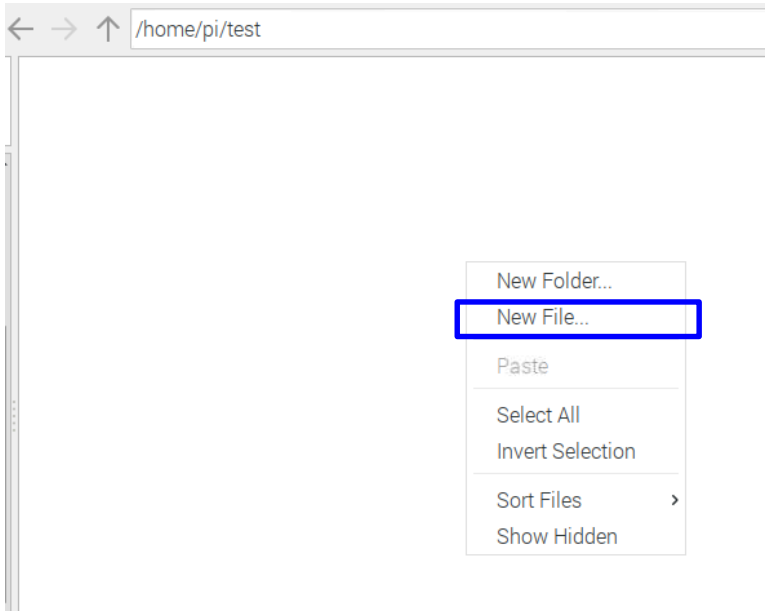
LED 테스트

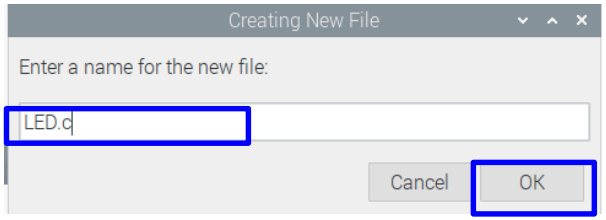
❐ 소스 파일 생성

- 생성한 디렉터리로 이동



- 파일 생성

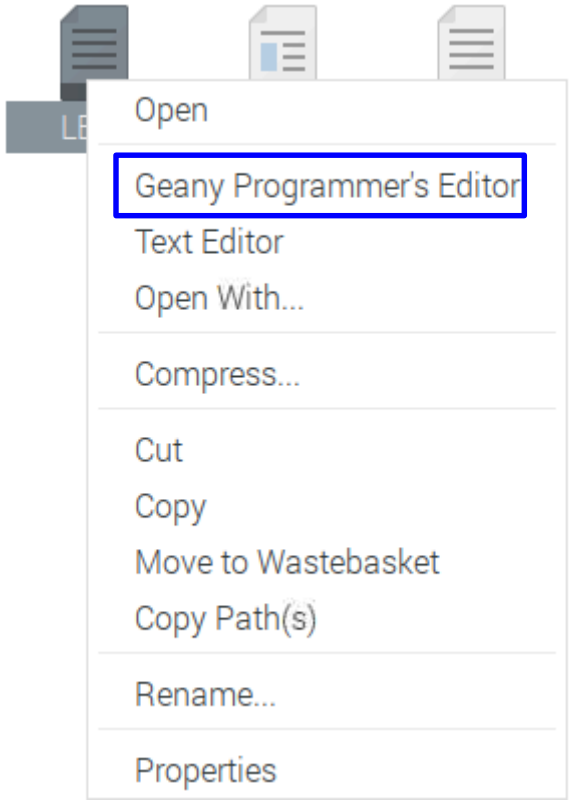




LED 테스트

❐ 소스 코드 작성

- 생성된 파일 실행



LED 테스트

❐ 소스 코드 작성

- 다음 소스 코드 입력

LED.c

```
1  #include <wiringPi.h>
2  #include <stdio.h>
3
4  #define LED1 6
5
6  int main()
7  {
8      if(wiringPiSetup() == -1) return -
9
10     pinMode(LED1, OUTPUT);
11     while(1)
12     {
13         digitalWrite(LED1, 1);
14         delay(1000);
15         digitalWrite(LED1, 0);
16         delay(1000);
17     }
18
19 }
```

LED 테스트

❐ 소스 코드 컴파일 및 실행

- 터미널 실행
- 디렉터리 이동
- 컴파일

```
pi@raspberrypi:~ $ cd test/
pi@raspberrypi:~/test $ gcc LED.c -o LED.o -lwiringPi
```

- 실행

```
pi@raspberrypi:~ $ cd test/
pi@raspberrypi:~/test $ gcc LED.c -o LED.o -lwiringPi
```