

# 디바이스마트

## 아두이노 **WIFI** 리모컨 키트

기획&제작 : 엔지니어 흥프로

기획 & 홍보 : **MD** 윤주임

## 제품 구성 및 목적

아두이노와 적외선 LED 모듈을 이용한 원격 리모컨 키트입니다.

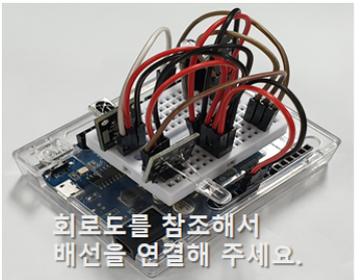
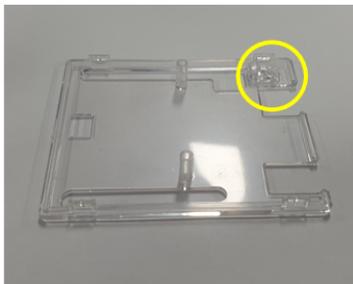
WIFI 로 연결된 아두이노가 적외선 LED를 이용해 리모컨의 역할을 수행합니다.

수광부를 통해 기존의 리모컨 정보를 해킹하고 해당 정보를 발광부가 같은 정보를 내보내는 방식입니다.

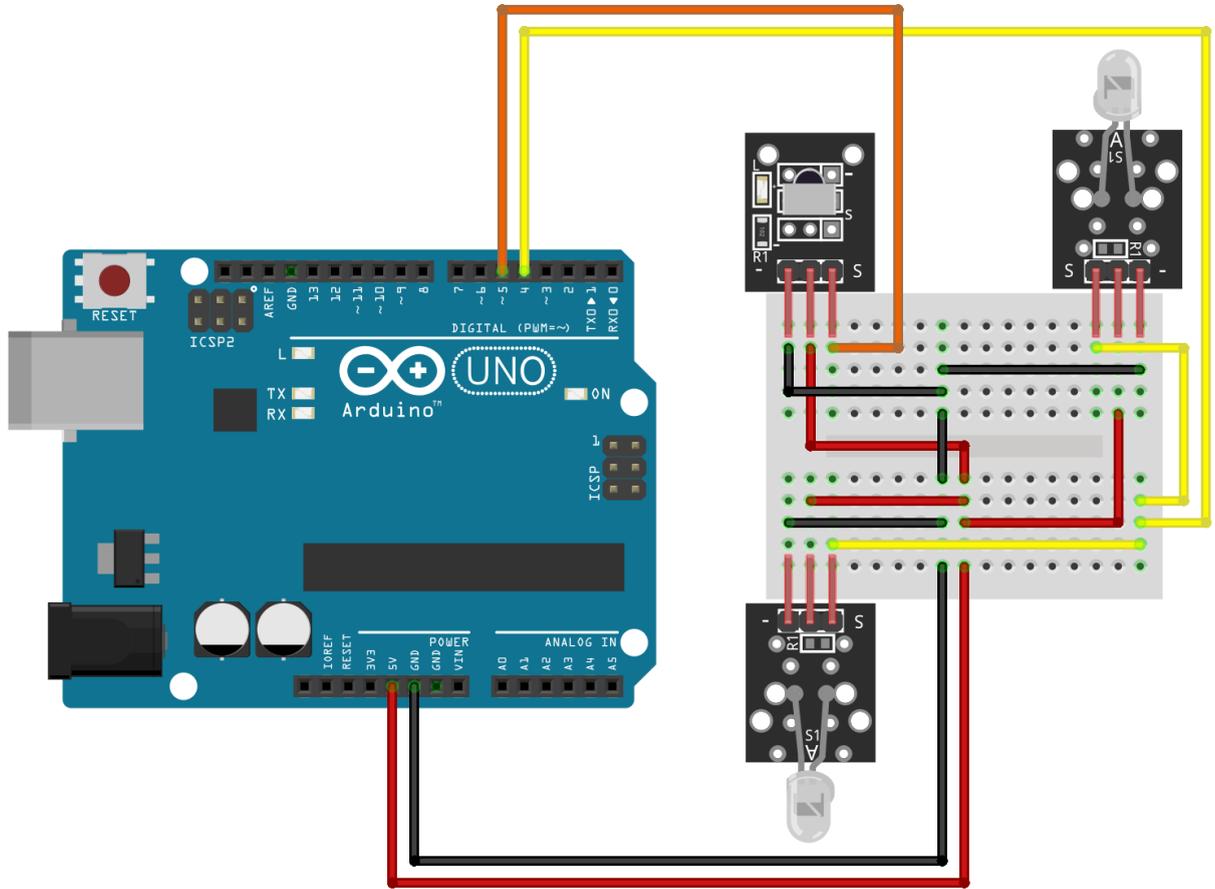
본제품은 교육 & 체험 & 원리 이해를 위해 구성된 제품입니다.

본래의 제품보다 성능이 떨어질 수 있습니다.

### 1. 조립 방법입니다.



## 2. 회로도



fritzing

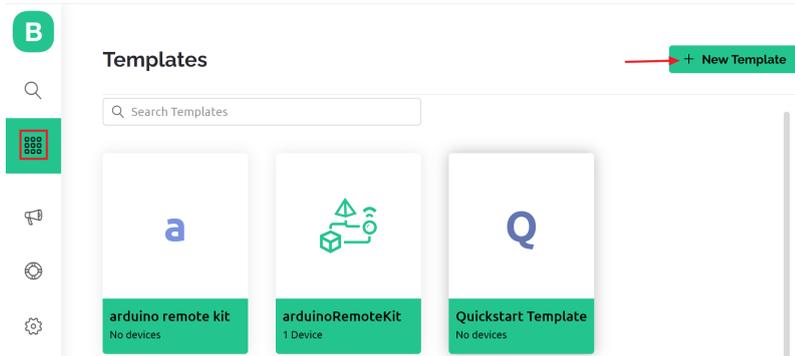
## Blynk 2.0 설정 및 연동

<https://blynk.io/> [Blynk 2.0]

위 링크로 접속해 회원 가입 후 로그인 합니다.



### 1. Blynk 웹 설정 (PC)



템플릿 창으로 이동 후 **New Template**를 클릭해 만들어 줍니다. 템플릿 설정은 다음과 같습니다.

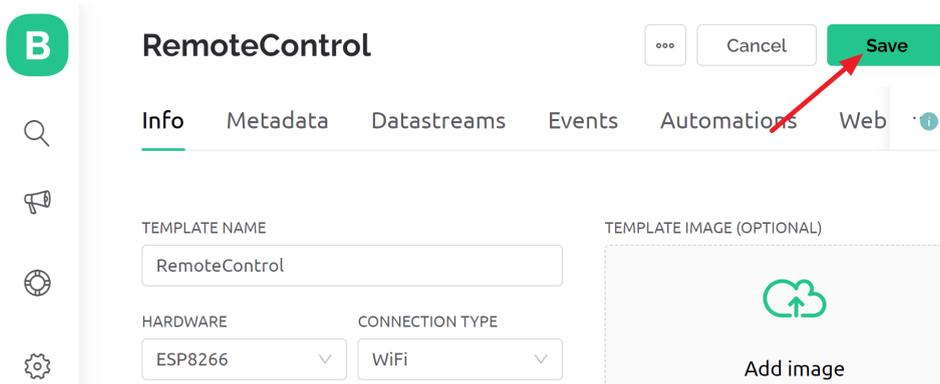
**(NAME : RemoteControl / HARDWARE : ESP8266)**

**(CONNECTION TYPE : WiFi / DESCRIPTION: 아두이노 리모컨 키트 테스트)**

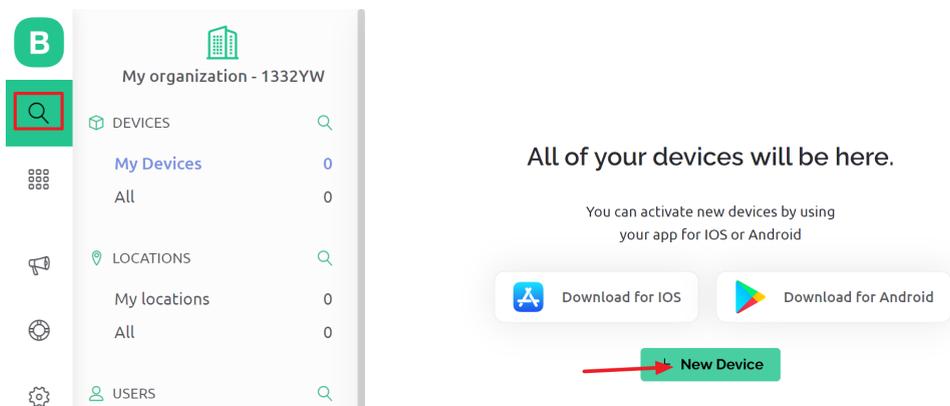
#### Create New Template

NAME	RemoteControl		
HARDWARE	ESP8266	CONNECTION TYPE	WiFi
DESCRIPTION	아두이노 리모컨 키트 테스트		

위와 같이 설정 후 만들어 준 다음에 곧 바로 **Save** 버튼을 눌러 저장 합니다.



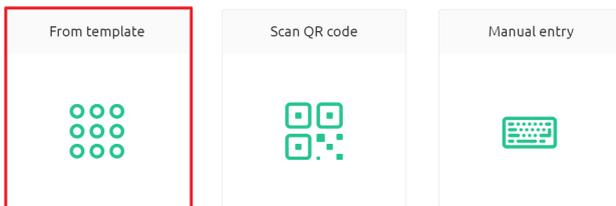
이후 서치(돋보기) 창으로 이동해 새로운 디바이스를 만들어 줍니다.



아래 이미지와 같이 **From template**(템플릿으로 가져오기)를 선택합니다. 이후 지정했던 이름과 같도록 선택해 만들어 줍니다.

### New Device

Choose a way to create new device



### New Device

Create new device by filling in the form below

A screenshot of the 'New Device' form. The 'TEMPLATE' dropdown menu is highlighted with a red box, and the 'RemoteControl' text in the dropdown is also highlighted with a red box. Below it is a 'DEVICE NAME' text input field containing 'RemoteControl'.

이후 **Device info** 에서 템플릿 정보를 불러 올 수 있습니다.

(해당 부분은 추후 예제코드에 추가 작성해야 해주셔야 합니다. 뒤에 자세한 내용이 나옵니다.)

Dashboard   Timeline   **Device Info**   Metadata   Actions Log

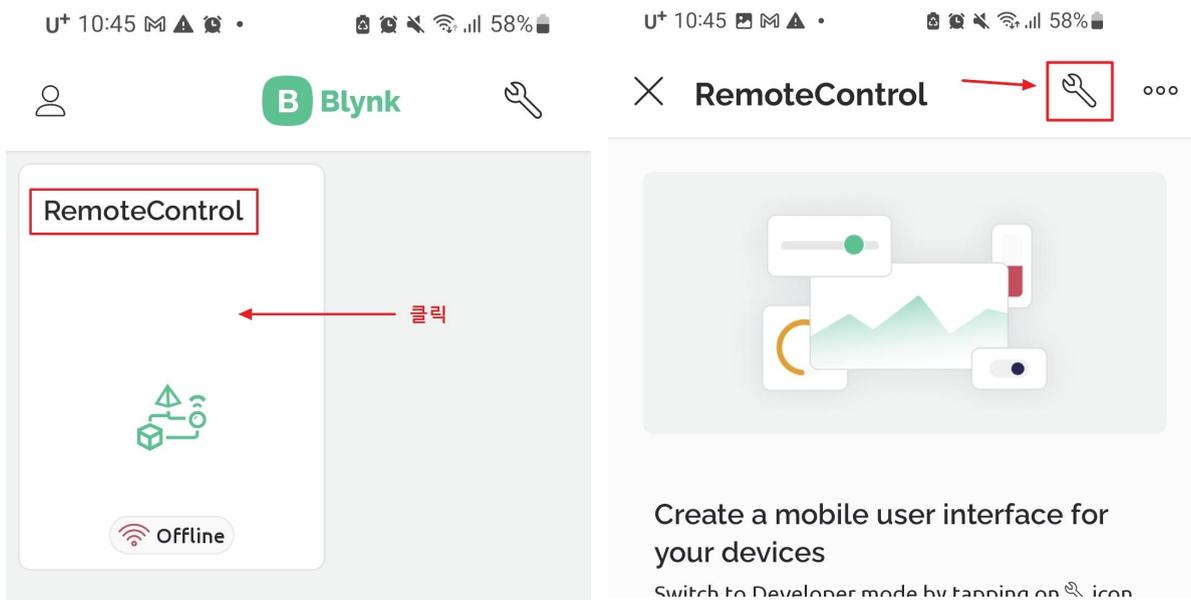
<b>STATUS</b> Offline	<b>LAST UPDATED</b> Not updated yet	<b>FIRMWARE CONFIGURATION</b> <pre>#define BLYNK_TEMPLATE_ID "TMPLtey3SP9q" #define BLYNK_DEVICE_NAME "RemoteControl" #define BLYNK_AUTH_TOKEN "3JmyTdd4BG9jCpYzeJVGyuf_soGGg3su"</pre>
<b>DEVICE ACTIVATED</b> 10:38 AM Today by chlxogus33@gmail.com	<b>ORGANIZATION</b> My organization - 1332YW	Template ID, Device Name, and AuthToken should be declared at the very top of the firmware code.

## 2. Blynk 2.0 앱 설정 (스마트폰)

플레이스토어에 접속해서 **Blynk** 를 검색해 아래 이미지와 같은 어플을 설치해 줍니다.

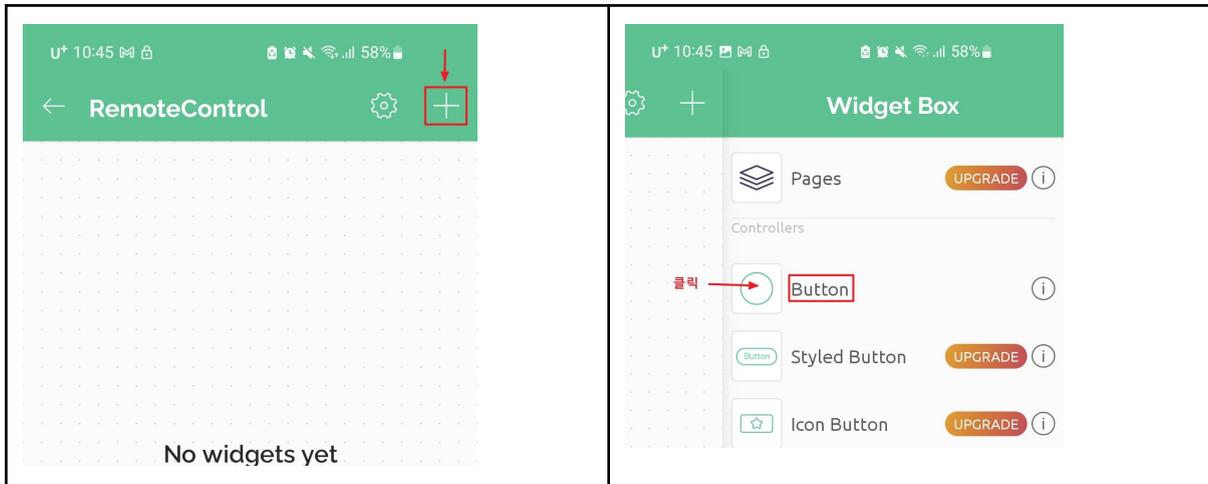


어플을 실행 한 후 로그인 해주시면 사전에 만들어 두었던 템플릿이 있습니다! 클릭해 접속 한 뒤 빨간 네모 칸으로 표시한 버튼을 눌러 줍니다.

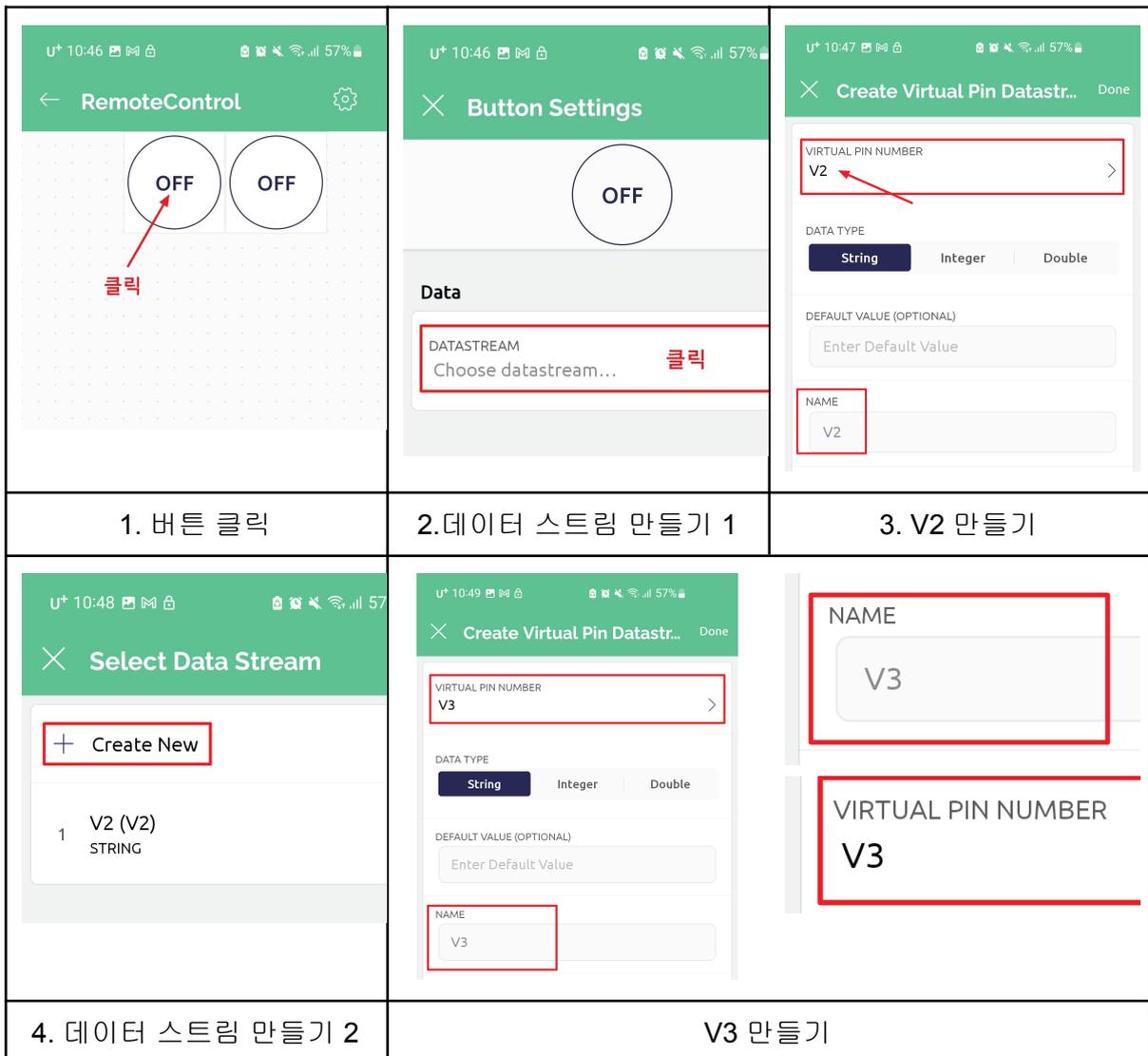


이후 + 버튼을 눌러 대시보드를 만들 수 있습니다.

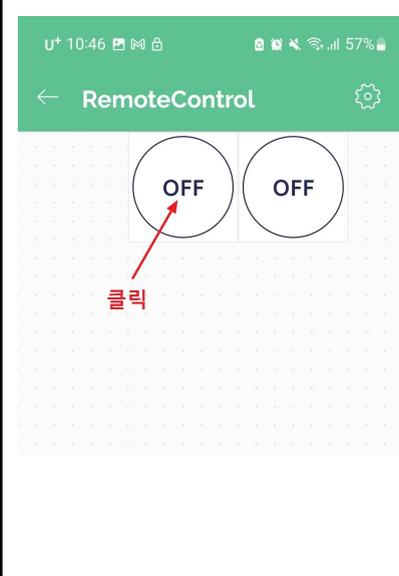
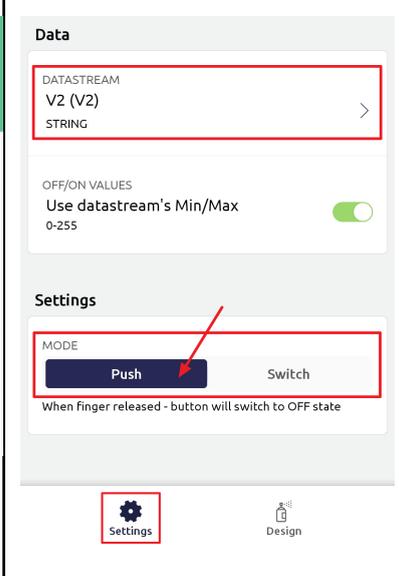
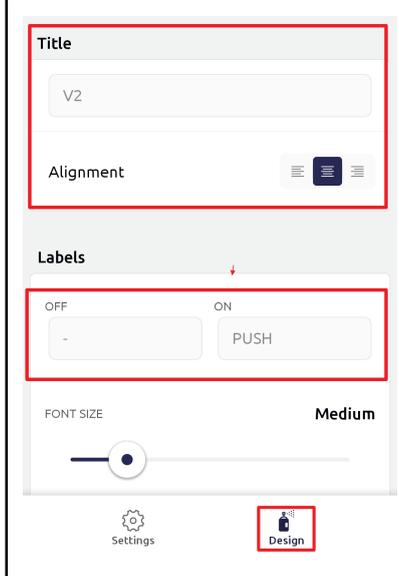
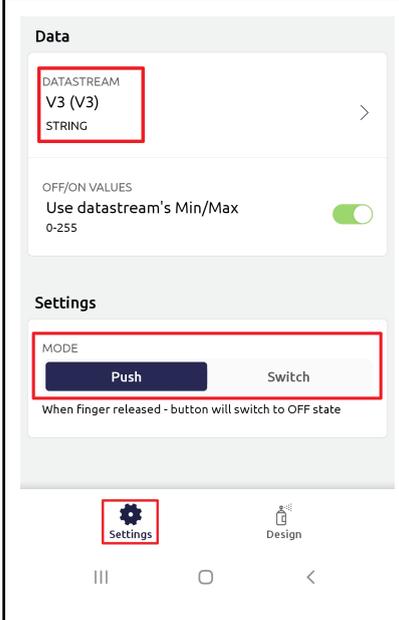
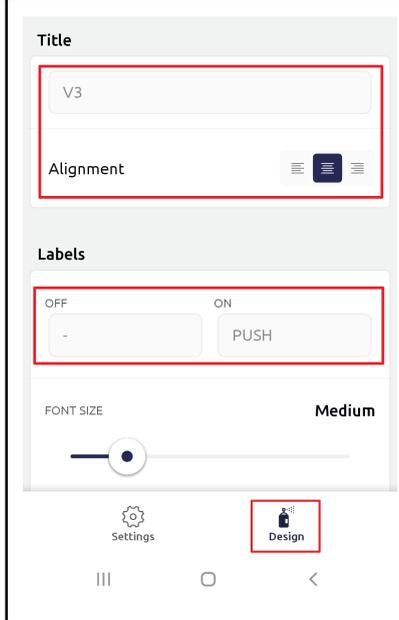
이번에 사용될 제어박스는 버튼 2개 입니다 클릭해 만들어 줍니다.



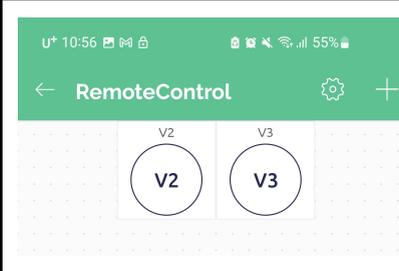
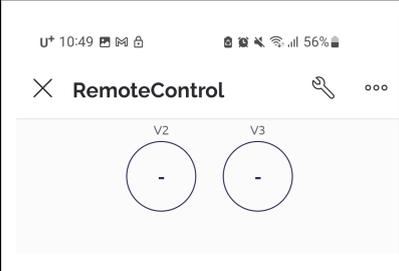
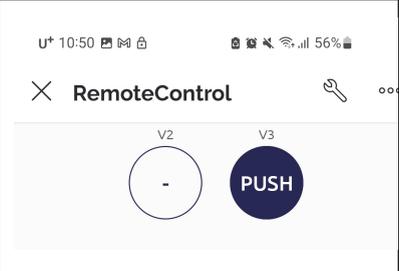
만들어준 버튼을 클릭해 다음과 같은 순서로 데이터 스트림을 만들어 줍니다.



데이터 스트림을 모두 만들어 준 후 각 버튼에 다음과 같은 순서로 설정합니다.

		
<p>1. V2 버튼 설정</p>	<p>2. V2 셋팅 설정</p>	<p>3. V2 디자인 설정</p>
		
<p>4. V3 버튼 설정</p>	<p>5. V3 셋팅 설정</p>	<p>6. V3 디자인 설정</p>

모두 설정을 완료하시면 다음과 같이 버튼이 활성화 됩니다.

		
설정완료	기본	V3 버튼을 눌렀을 때

### 3. 보드 설정 및 라이브러리 추가

#### 아두이노 IDE 설정

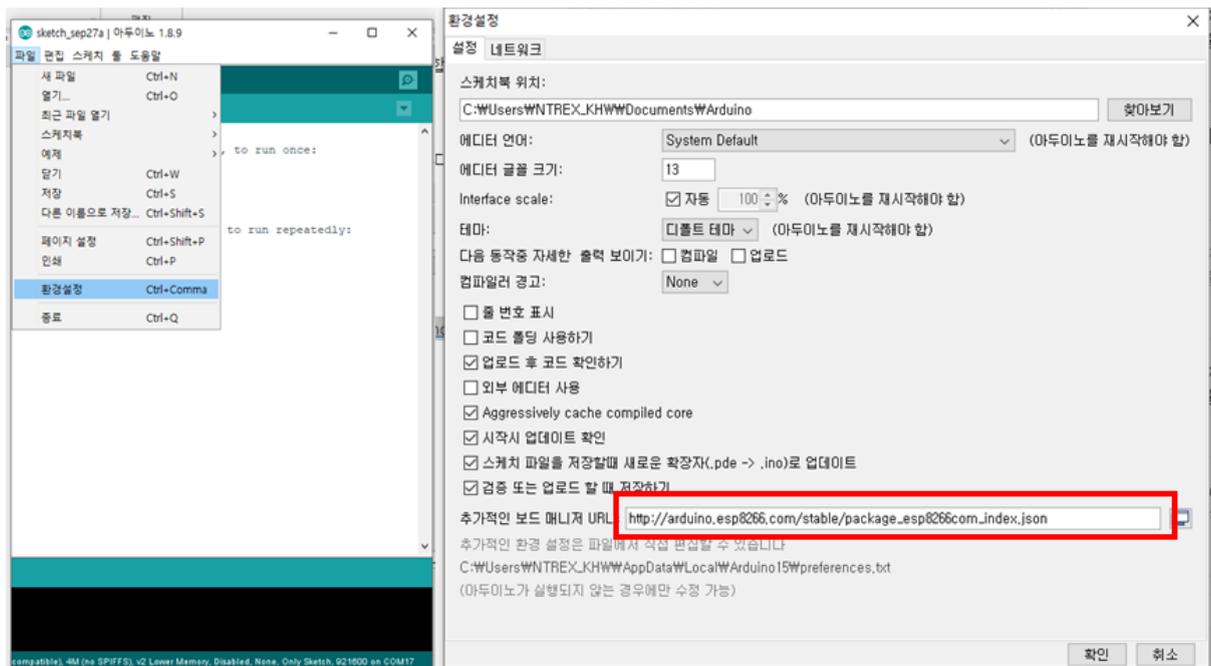
아두이노 IDE 를 실행 합니다.

파일 -> 환경 설정

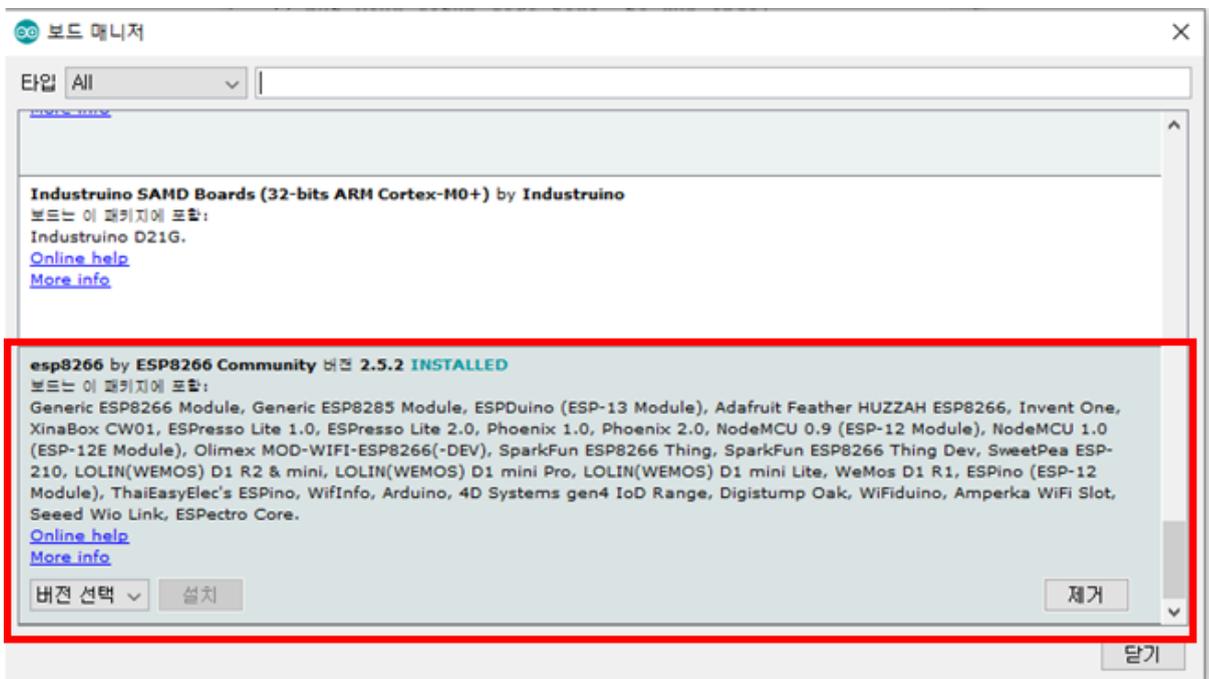
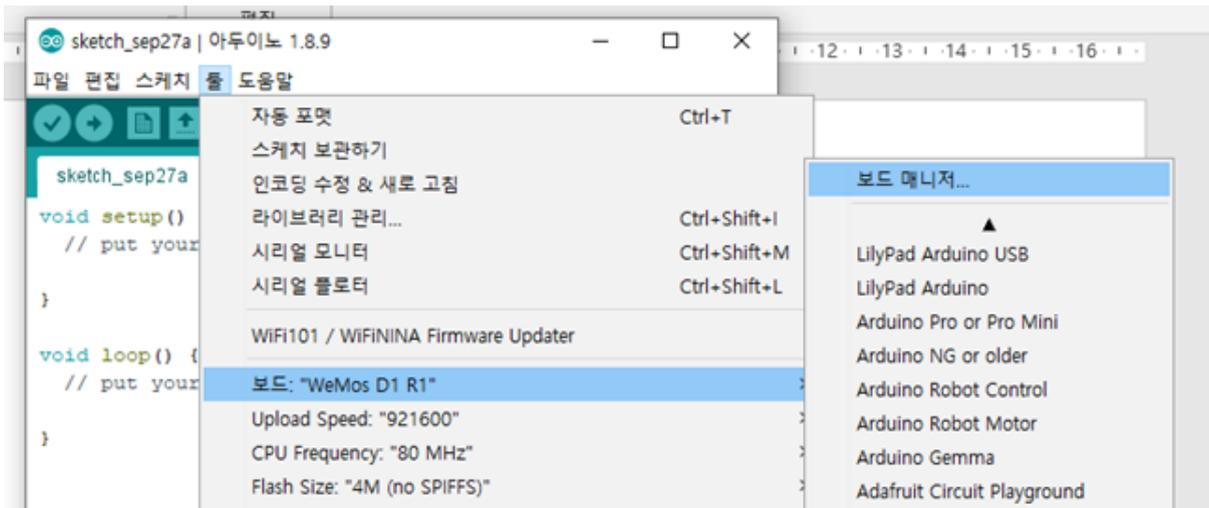
추가적인 보드 매니어 URLs 입력란에

[http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json)

입력 하신 후 확인



메뉴에서 툴 -> 보드: xxxx -> 보드 매니저... 클릭  
아래로 내려가면 esp8266 by esp8266 community  
를 설치해 줍니다!



Blynk 라이브러리 추가하기.

<https://github.com/blynkkk/blynk-library>

에서 오른쪽 Clone or download(초록색) 를 선택하고  
Download ZIP로 받아 줍니다.

**Join GitHub today** Dismiss

GitHub is home to over 36 million developers working together to host and review code, manage projects, and build software together.

[Sign up](#)

Blynk library for embedded hardware. Works with Arduino, ESP8266, Raspberry Pi, Intel Edison/Galileo, LinkIt ONE, Particle Core/Photon, Energia, ARM mbed, etc. <https://www.blynk.cc/>

- arduino
- esp8266
- esp32
- particle-photon
- embedded
- hardware
- mbed
- raspberry-pi
- bluetooth
- bluetooth-low-energy
- serialport
- ethernet-shield
- gsm
- ota
- iot
- internet-of-things
- espressif
- iot-platform
- wifi

1,747 commits | 1 branch | 31 releases | 18 contributors | MIT

Branch: master | [New pull request](#)

[Find File](#) [Clone or download](#)

**Clone with HTTPS**

Use Git or checkout with SVN using the web URL.

<https://github.com/blynkk/blynk-library>

[Open in Desktop](#) [Download ZIP](#)

<b>B</b> blynkk Update README.md	
└─ .github	update template [ci skip]
└─ examples	Fix typo
└─ extras	Update README.md
└─ linux	Update info message format
└─ scripts	Switch SSL port to 443 by default

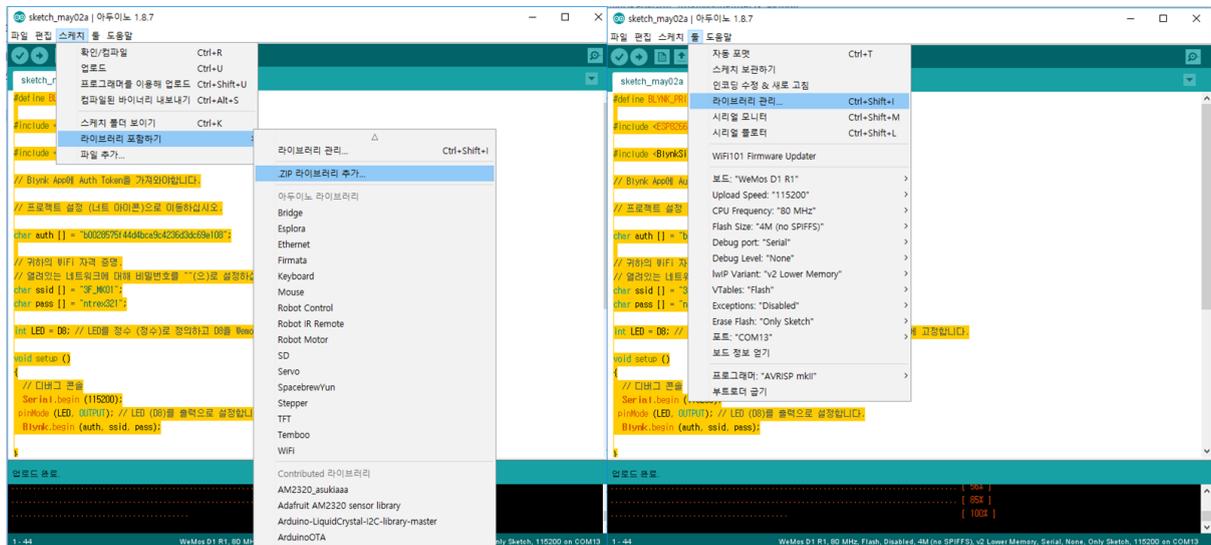
3 months ago

이후 해당 라이브러리를 추가하기 위해 스케치 -> 라이브러리 포함하기 -> .zip 라이브러리 추가를 눌러 이전에 다운 받은 파일을 선택해줍니다. 추가 후에 툴 -> 라이브러리 관리를 눌러주면 자동으로 적용됩니다. 라이브러리를 추가 하는김에 사용할 모든 라이브러리를 추가합니다. 첨부 파일로 올려 둔

### Arduino-IRremote-master.zip

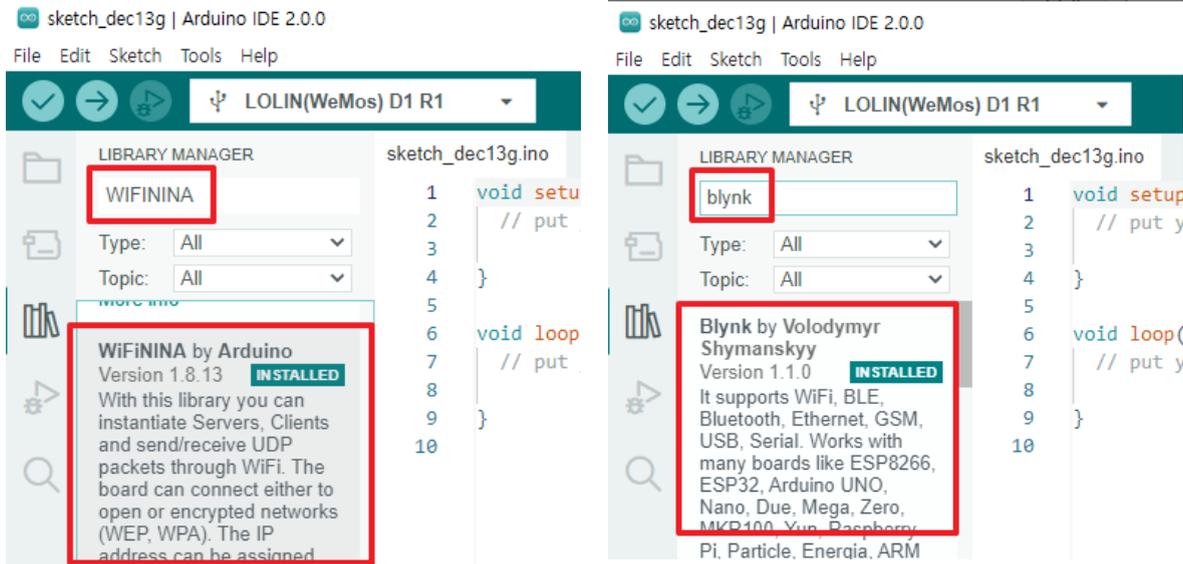
### IRremoteESP8266-master.zip

를 같은 방법으로 라이브러리에 추가해 줍니다.



Blynk가 2.0버전으로 업데이트 됨에 따라 추가적인 라이브러리 설치가 필요합니다.  
 Sketch > Include Library > Manage Libraries.. 를 클릭해 다음과 같이 검색 후 설치합니다.

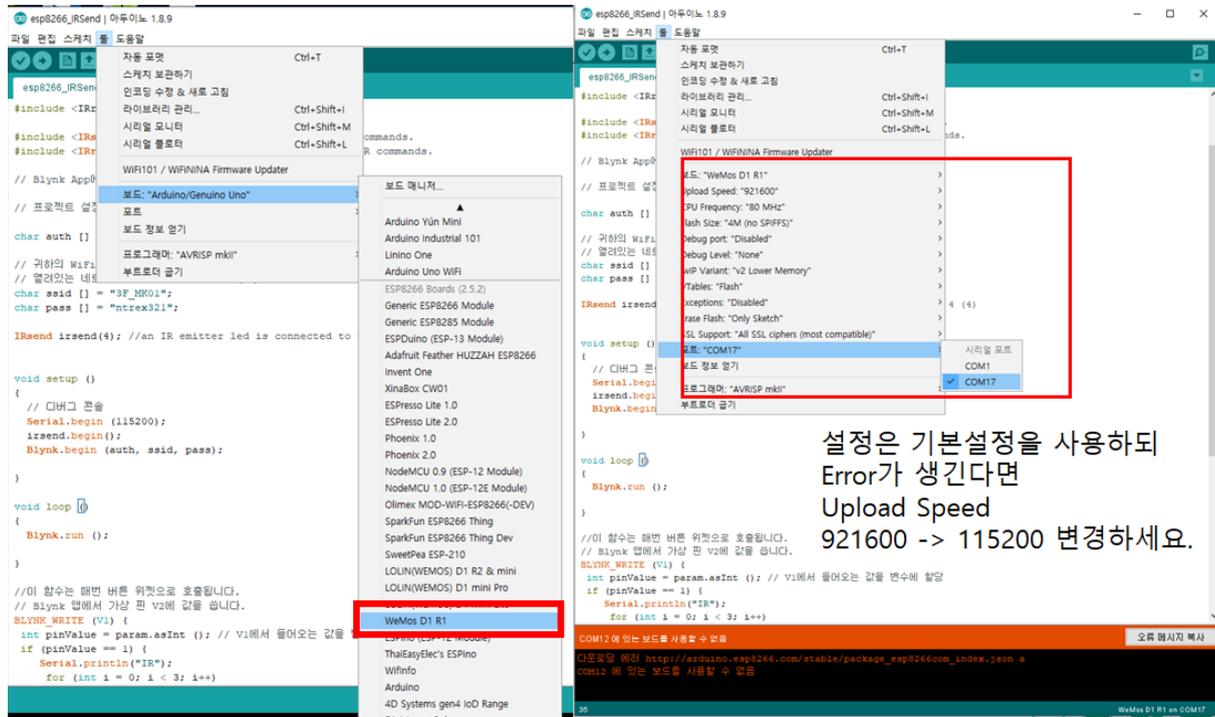
## WIFININA / blynk



### IRrecv.ino를 실행 합니다.(리모컨 해킹 코드)

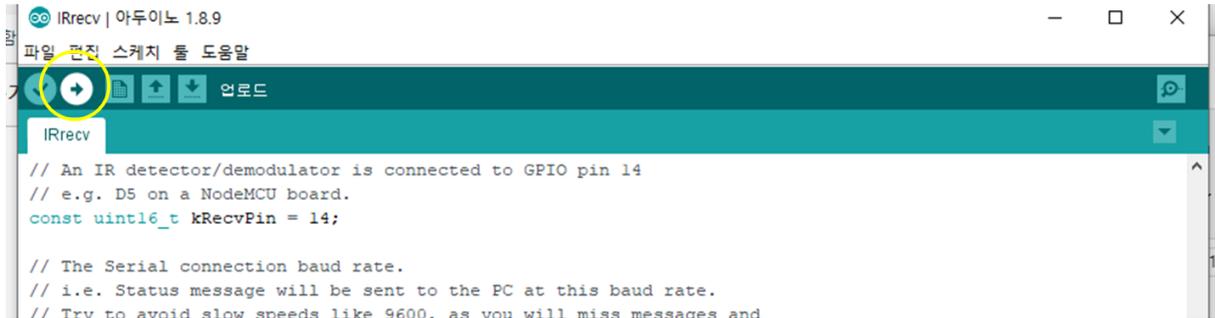
이후 툴->보드 -> WeMos D1 R1을 선택합니다.

이후 설정을 확인 한 후에 연결된 아두이노의 포트를 확인합니다.

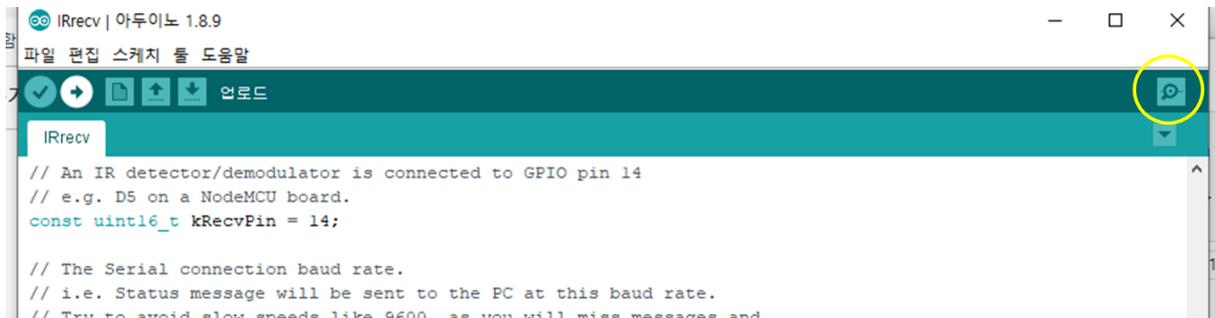


설정은 기본설정을 사용하되  
 Error가 생긴다면  
 Upload Speed  
 921600 -> 115200 변경하세요.

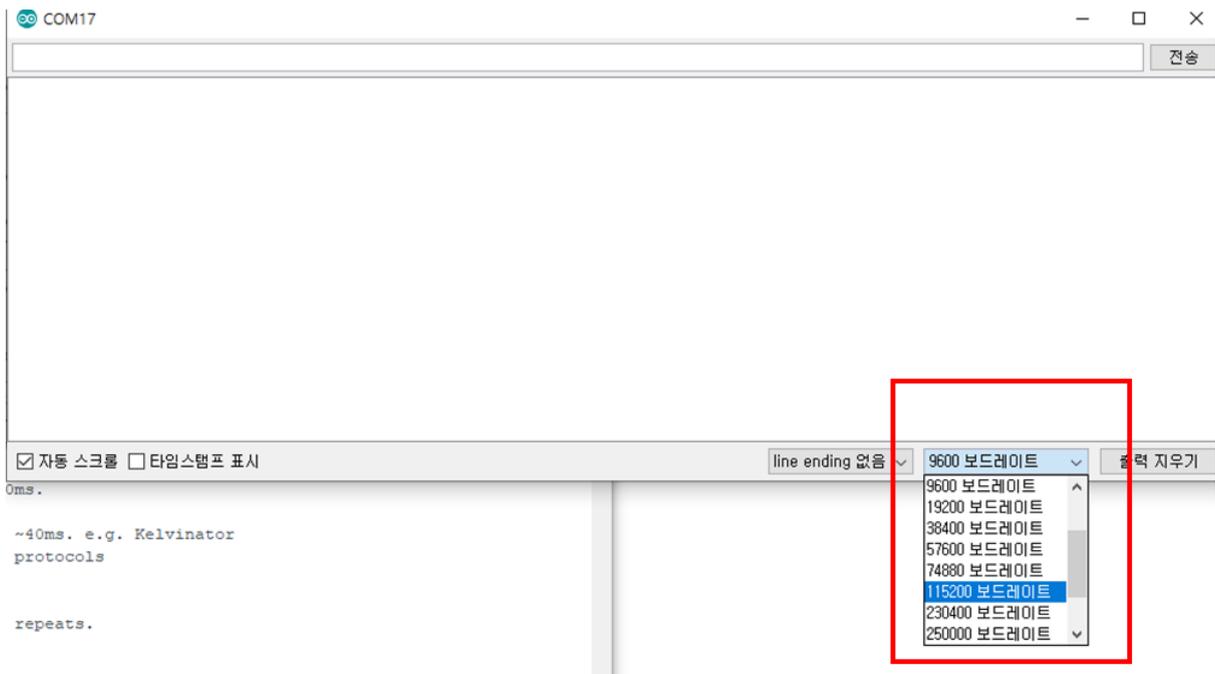
이후 상단에 업로드 버튼을 눌러 업로드 해줍니다.



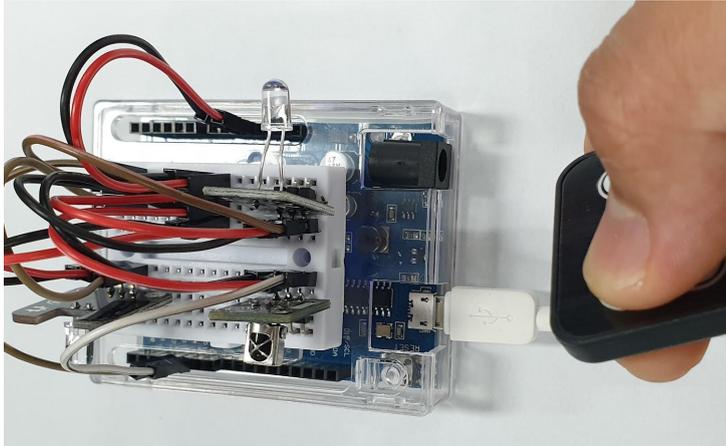
업로드가 끝나면 오른쪽 상단에 시리얼모니터를 실행해 주세요.



시리얼 모니터 하단의 보드레이트를 9600 -> 115200으로 변경해 주세요.



리모컨을 수신부 쪽으로 향한 후 눌러주면 아래와 같이 리모컨 정보가 해킹 됩니다.  
여러번 눌러서 일정한 값으로 확인되는 정보를 사용하시길 바랍니다.



```

COM17
Timestamp : 000177.151
Library   : v2.6.4

Encoding  : NEC
Code      : FFA25D (32 bits)
Raw Timing[71]:
+ 9096, - 4428, + 638, - 490, + 638, - 492, + 638, - 492,
+ 638, - 492, + 638, - 492, + 638, - 492, + 638, - 492,
+ 640, - 490, + 638, - 1594, + 638, - 1594, + 638, - 1594,
+ 638, - 1594, + 638, - 1596, + 638, - 1594, + 638, - 1592,
+ 638, - 1594, + 636, - 1596, + 638, - 492, + 636, - 1594,
+ 638, - 492, + 638, - 492, + 638, - 492, + 636, - 1594,
+ 638, - 492, + 638, - 492, + 636, - 1594, + 638, - 492,
+ 636, - 1594, + 638, - 1596, + 618, - 1614, + 612, - 516,
+ 636, - 1596, + 612, - 39640, + 9122, - 2166, + 640

uint16_t rawData[71] = {9096, 4428, 638, 490, 638, 492, 638, 492, 638, 492, 638, 492, 638, 492, 638, 492, 640, 49
uint32_t address = 0x0;
uint32_t command = 0x15;
uint64_t data = 0xFFA25D;
    
```

### esp8266\_IRSend.ino를 실행 합니다.

blynk 웹에서 템플릿 정보를 복사해서 아래 이미지와 같이 붙여 넣어줍니다.  
이후 wifi 정보를 사용하시는 정보로 작성해주세요. (대소문자 구별)

Dashboard   Timeline   **Device Info**   Metadata   Actions Log

---

<b>STATUS</b> Offline	<b>LAST UPDATED</b> Not updated yet	<b>FIRMWARE CONFIGURATION</b> <pre> #define BLYNK_TEMPLATE_ID "TMPLtey3SP9q" #define BLYNK_DEVICE_NAME "RemoteControl" #define BLYNK_AUTH_TOKEN "3JmyTDd4BG9jCpYzeJVGyuf_soGGg3su"           </pre>
<b>DEVICE ACTIVATED</b> 10:38 AM Today by chixogus33@gmail.com	<b>ORGANIZATION</b> My organization - 1332YW	Template ID, Device Name, and AuthToken should be declared at the very top of the firmware code.

```

1 #define BLYNK_PRINT Serial
2
3 /* Fill-in your Template ID (only if using Blynk.Cloud) */
4 // #define BLYNK_TEMPLATE_ID "YourTemplateID"
5 #define BLYNK_TEMPLATE_ID "TMPLtey3SP9q"
6 #define BLYNK_DEVICE_NAME "RemoteControl"
7 #define BLYNK_AUTH_TOKEN "3JmyTdd4B69jCpYzeJVGyuf_soGGg3su"
8
9 #include <ESP8266WiFi.h>
10 #include <BlynkSimpleEsp8266.h>
11 #include <IRremoteESP8266.h>
12 #include <IRsend.h> // Needed if you want to send IR commands.
13 #include <IRrecv.h> // Needed if you want to receive IR commands.
14
15 IRsend irsend(4); //an IR emitter led is connected to GPIO pin 4 (4)
16
17 char auth[] = BLYNK_AUTH_TOKEN;
18
19 // 귀하의 WiFi 자격 증명.
20 // 열려있는 네트워크에 대해 비밀번호를 ""(으)로 설정하십시오.
21 char ssid[] = "*****"; // WIFI ID
22 char pass[] = "*****"; // WIFI PASSWORD
23

```

웹에서 가져온 템플릿 정보를 복사해서 붙여 넣어주세요

자신 사용하는 wifi 정보를 입력해주세요 (대소문자 구별)

업로드 코드 수정을 해줍니다.

Blynk 의 Virtual 변수를 넣어주는 겁니다. V3를 사용하기로 했으니 V3로 바꿔주세요

```

BLYNK_WRITE (V1) {
  int pinValue = param.asInt (); // v1에서 들어오는 값을 변수에 할당
  if (pinValue == 1) {
    Serial.println("IR");
    for (int i = 0; i < 3; i++)
    {
      irsend.sendNEC (0xFFA25D, 32); //ON
      Serial.print ("NFC");
      delay (40);
    }
  }
}

```

이전에 해킹한 리모컨 정보를 넣어주시면 됩니다.

여긴 비트 수 입니다. 해당 글자는 16진수기 때문에 2진수로 표현 되는 자리수는 32비트입니다.  
 0xFFA25D -> 'FFA25D' 6자리로 2의 6승 = 32를 뜻합니다.  
 0xFFA25 -> 'FFA25' 5자리로 2의 5승 = 16 을 넣어주면 됩니다.

시리얼 모니터에 표시할 내용입니다. 버튼이 눌러 해당 정보가 잘 송출되었는지 확인하는 것입니다.

상단에 업로드를 눌러 업로드 해줍니다.

```

1 #define BLYNK_PRINT Serial
2
3 /* Fill-in your Template ID (only if using Blynk.Cloud) */
4 // #define BLYNK_TEMPLATE_ID "YourTemplateID"
5 #define BLYNK_TEMPLATE_ID "TMPLtey3SP9q"
6 #define BLYNK_DEVICE_NAME "RemoteControl"
7 #define BLYNK_AUTH_TOKEN "3JmyTdd4B69jCpYzeJVGyuf_soGGg3su"
8

```

```
Output Serial Monitor x
Message (Ctrl + Enter to send message to 'LOLIN(WeMos) D1 R1' on 'COM24')
[6294] Connected to WiFi
[6294] IP: 192.168.1.45
[6294]
  _ _ _ _ _
 / _ ) / / _ _ _ _ _ / / _
 / _ / / / / / _ # / ' _ /
 / _ _ _ / / # _ , / _ / / / / # _ #
      / _ _ / v1.1.0 on ESP8266

#StandWithUkraine   https://bit.ly/swua

[6304] Connecting to blynk.cloud:80
[6584] Ready (ping: 90ms).
```

업로드가 완료되면 시리얼 모니터를 열고 Blynk에 잘 접속이 되는지 확인합니다.  
보드레이트를 115200으로 변경 후 Blynk app 을 실행 합니다.

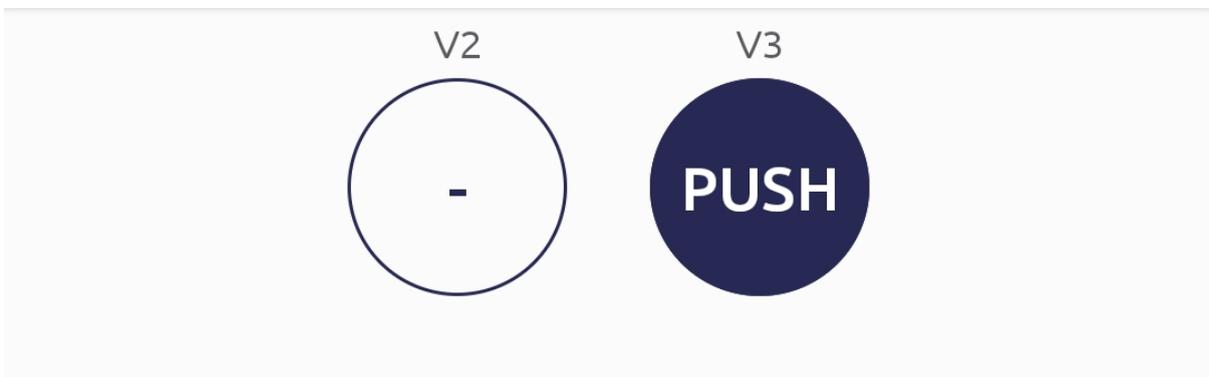
이전에 만들어 놓은 프로젝트를 들어갑니다.

미리 만들어둔 템플릿을 클릭해 접속합니다.  
각 버튼을 눌러 시리얼 모니터를 확인하세요.

U+ 10:50 [notification icons]

[notification icons] 56% [battery icon]

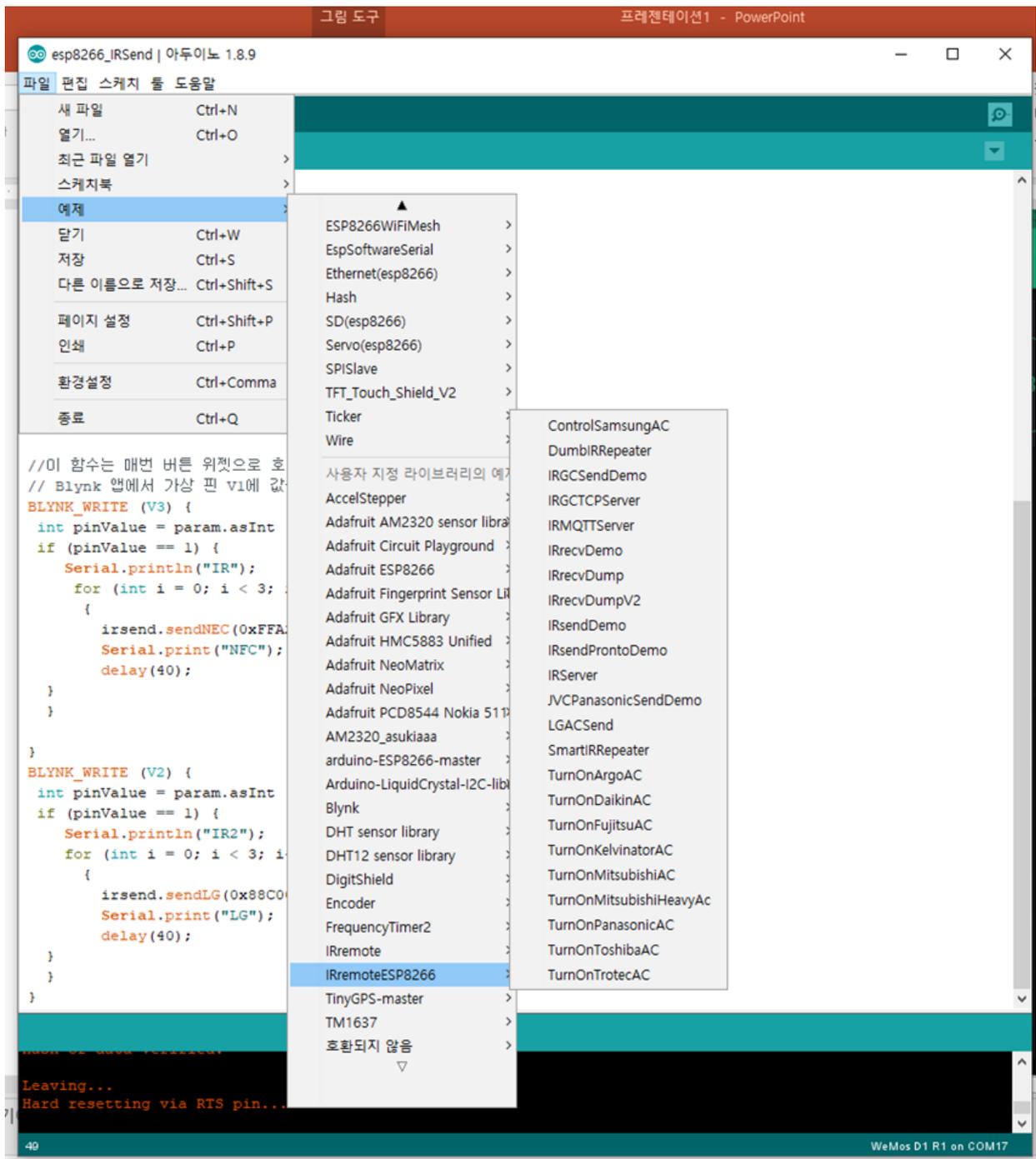
# RemoteControl



이 상태에서 버튼을 누르게 되면 시리얼모니터 상에 설정해둔 버튼이름이 ((ex)V3) 라고 출력합니다.

```
[6304] Connecting to blynk.cloud:80
[6584] Ready (ping: 90ms).
V3
0x88C0051
0x88C0051
0x88C0051
V3
0x88C0051
0x88C0051
0x88C0051
V3
0x88C0051
0x88C0051
0x88C0051
V2
0xFFA25D
0xFFA25D
0xFFA25D
```

추가로 삼성 에어컨이나 LG 에어컨은 파일 -> 예제 -> **IRremoteESP8266** 에 있습니다.  
생각보다 최신 에어컨도 동작을 하니 참고하시길 바랍니다.



이 예제와 제공해드린 예제들을 잘 조합하면 거실에 있는 **TV**, 에어컨, 선풍기, 공기청정기등 다양한 제품을 집밖에서 컨트롤 할 수 있는 **IOT** 리모컨을 만드실 수 있습니다.

화이팅...★