

코딩거점학교 심화교육과정

2023.08.13.(일)

코딩거점학교 심화교육과정(1일차)

일시		차시	주제	내용
8.12. (토)	09:00-10:00	1	ESP32 소개	- ESP32를 활용
	10:00-11:00	2	ESP32 입력	- ESP32를 활용한 센서 설명
	11:00-12:00	3		- ESP32를 활용한 센서 실습
	13:00-14:00	4	ESP32 출력	- ESP32를 활용한 액추에이터 설명
	14:00-15:00	5		- ESP32를 활용한 액추에이터 실습
	15:00-16:00	6	ESP32 응용	- ESP32를 활용한 작품 실습

코딩거점학교 심화교육과정(2일차)

일시		차시	주제	내용
8.13. (일)	09:00-10:00	7	프로젝트	- 프로젝트 안내 및 사례 소개
	10:00-11:00	8		- 프로젝트 진행(주제 선정 및 역할 분배)
	11:00-12:00	9		- 프로젝트 진행(제작)
	13:00-14:00	10		- 프로젝트 진행(제작)
	14:00-15:00	11		- 프로젝트 진행(제작)
	15:00-16:00	12	발표	- 프로젝트 발표

1

ESP32 프로젝트 예제

2

프로젝트 안내 및 진행

3

프로젝트 결과 발표

ESP32 프로젝트 예제

ESP32 프로젝트 예제

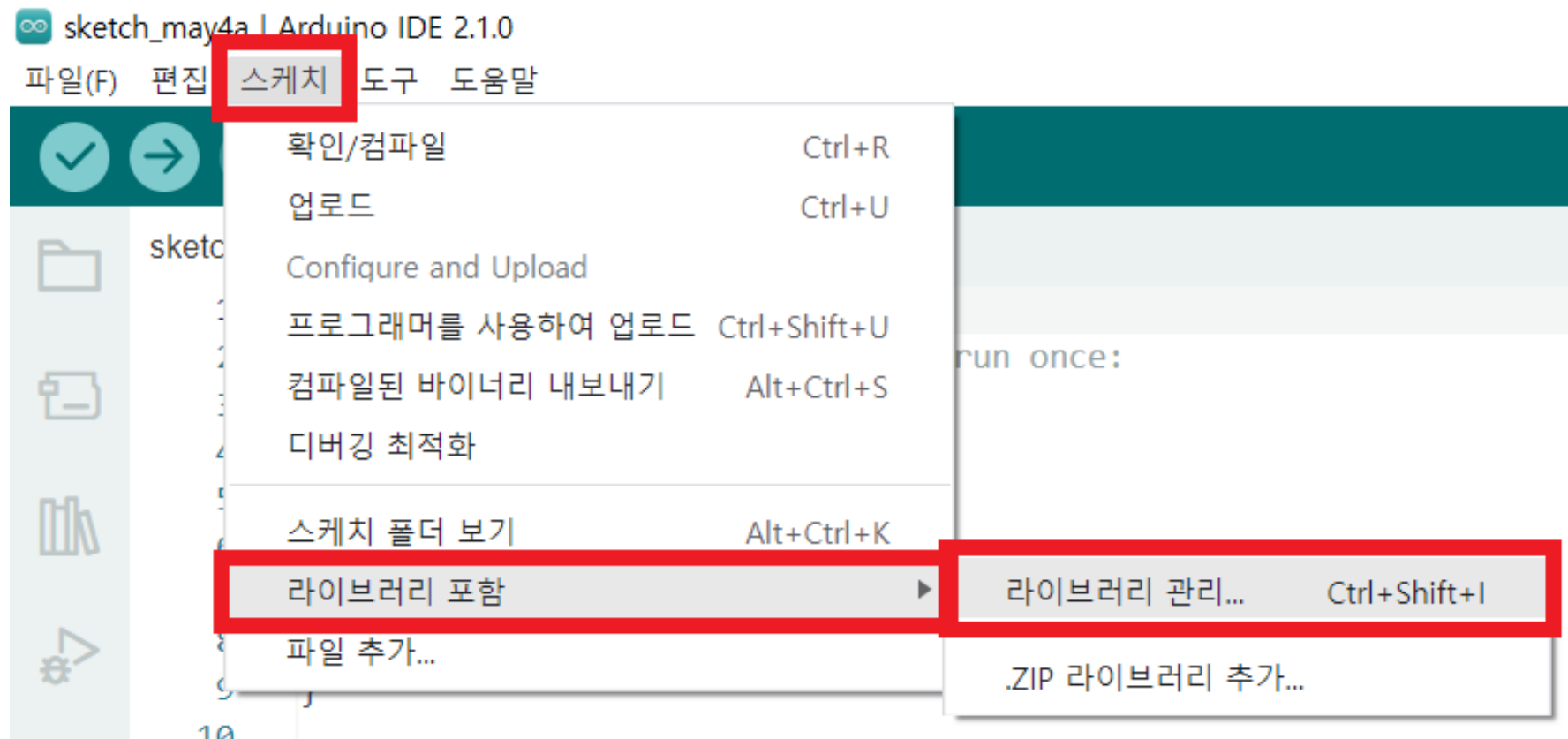
- LCD I2C 7
- 적외선(IR) 수신 센서 19
- 블루투스(Bluetooth) 30
- 와이파이(Wi-Fi) 41
- ThingSpeak for IoT Projects 51

LCD I2C



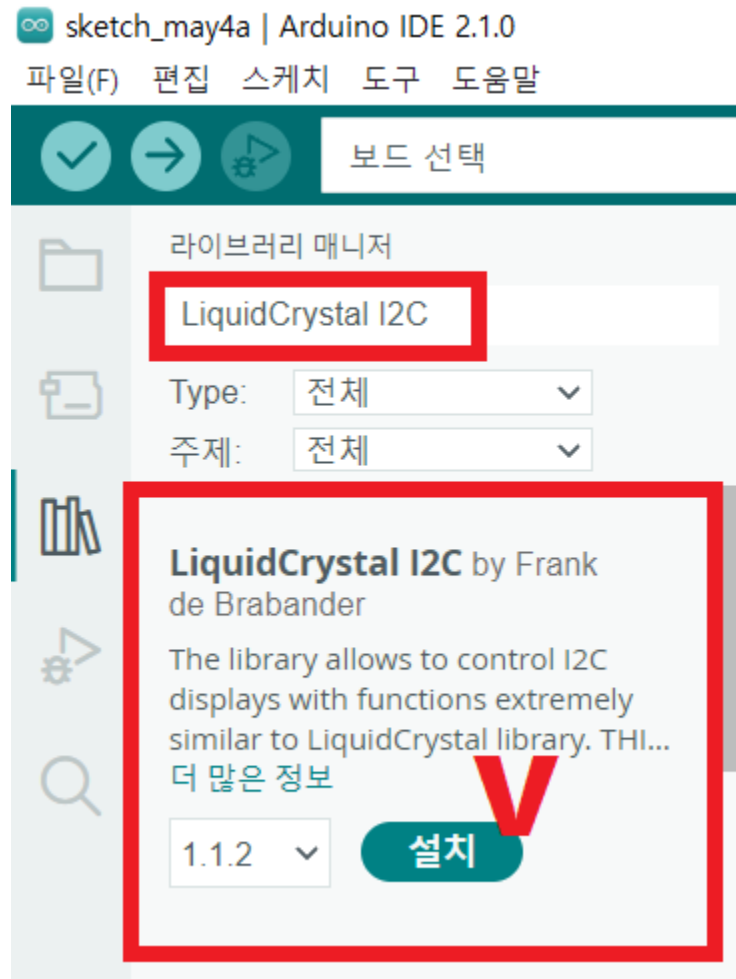
‘LiquidCrystal I2C’ 라이브러리 설치하기

Arduino IDE > 스케치 > 라이브러리 포함 > 라이브러리 관리



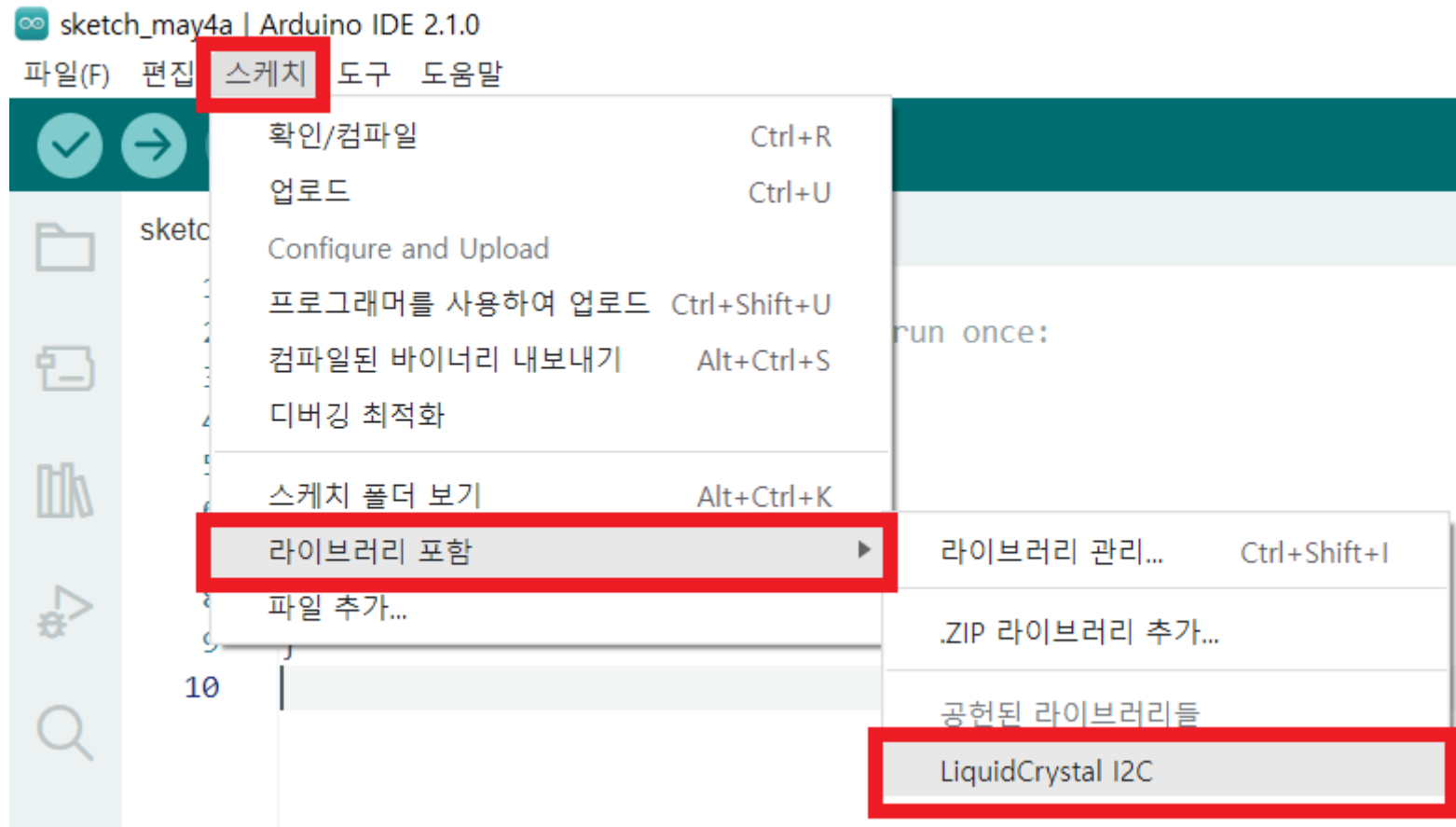
‘LiquidCrystal I2C’ 라이브러리 설치하기

라이브러리 매니저에서 ‘LiquidCrystal I2C’ 검색 후 설치



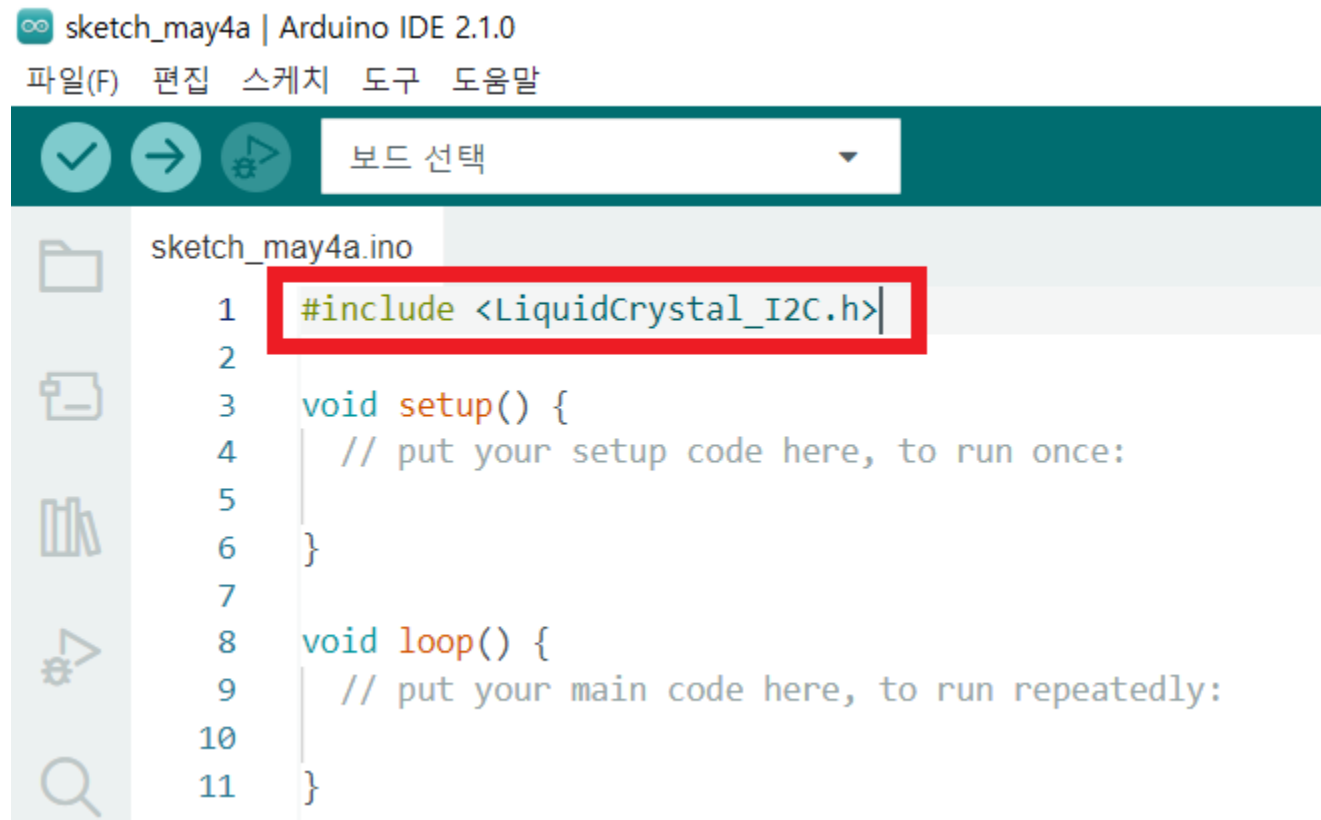
‘LiquidCrystal I2C’ 라이브러리 설치하기

스케치 > 라이브러리 포함 > 공헌된 라이브러리들에서 ‘LiquidCrystal I2C’ 선택



‘LiquidCrystal I2C’ 라이브러리 설치하기

라이브러리가 정상적으로 설치되어 코드에 반영되었는지 확인



```
sketch_may4a | Arduino IDE 2.1.0
파일(F) 편집 스케치 도구 도움말

✓ → ⚙ 보드 선택

sketch_may4a.ino
1  #include <LiquidCrystal_I2C.h>
2
3  void setup() {
4      // put your setup code here, to run once:
5
6  }
7
8  void loop() {
9      // put your main code here, to run repeatedly:
10
11 }
```

LCD 예제

```
#include <LiquidCrystal_I2C.h>

int lcdColumns = 16;           // lcd 열의 개수 설정
int lcdRows = 2;              // lcd 행의 개수 설정

LiquidCrystal_I2C lcd(0x27, 16, 2); // lcd 표시 주소, 열, 행

void setup(){
    lcd.init();                // lcd 초기화
    lcd.backlight();           // lcd 백라이트 켜기
}

...
```

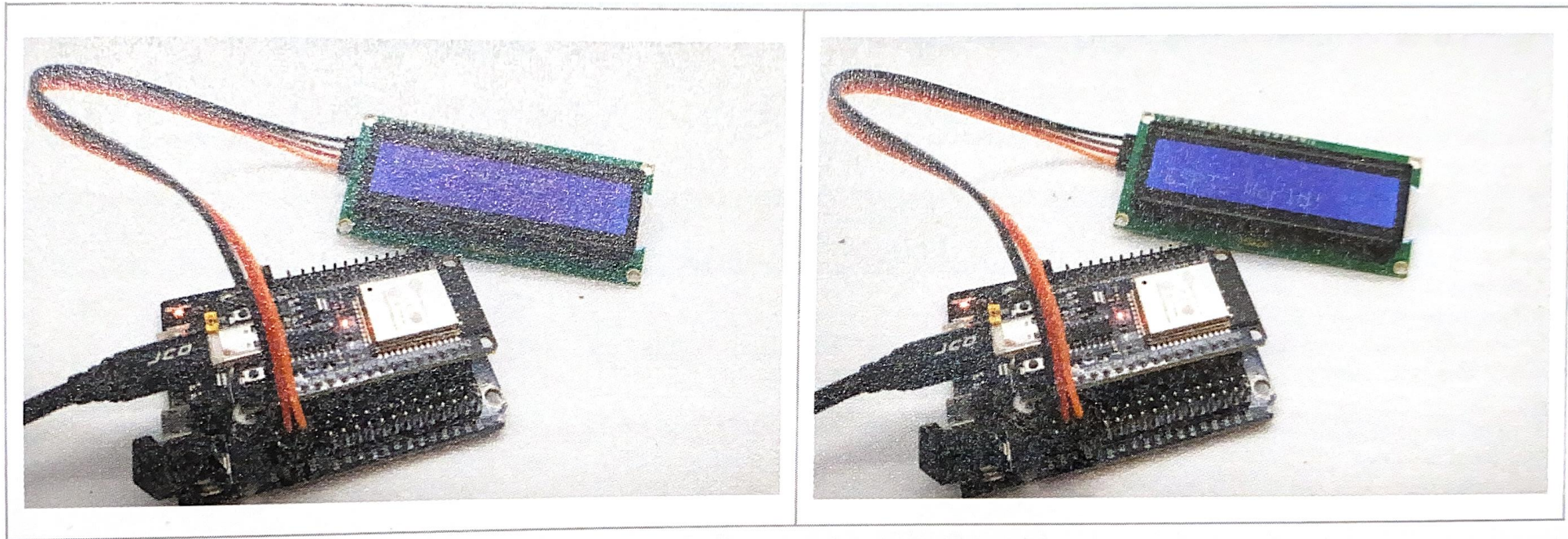
LCD 예제

...

```
void loop(){  
    lcd.setCursor(0, 0);           // 메시지를 쓸 위치(행, 열)  
    lcd.print("ESP32 World!");     // 메시지 출력  
    delay(1000);                  // 1초 기다리기  
    lcd.clear();                  // 메시지 지우기  
    lcd.setCursor(0, 1);  
    lcd.print("ESP32 World!");  
    delay(1000);  
    lcd.clear();  
}
```

LCD 예제 실행 결과

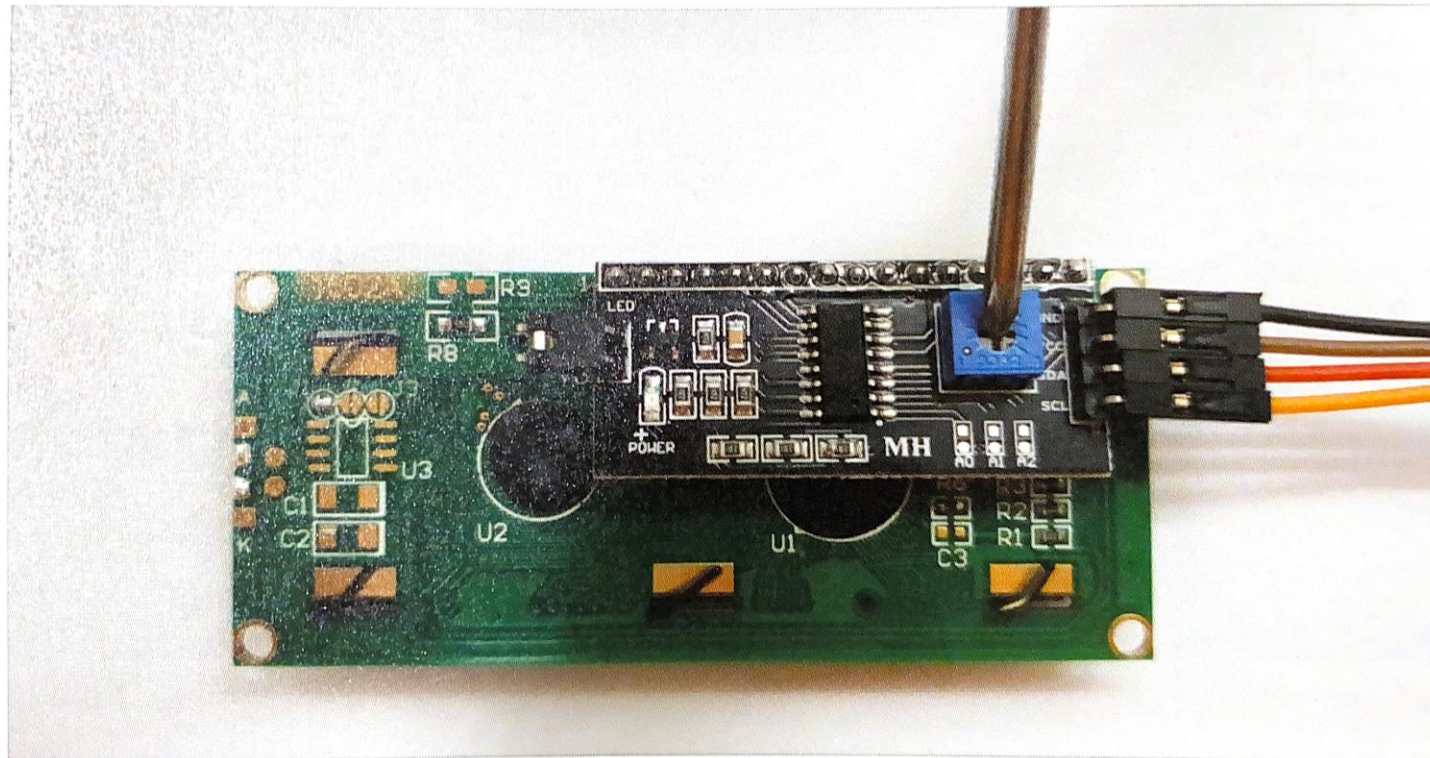
‘ESP32 World!’ 메시지가 첫 번째 열과 두 번째 열에 1초마다 번갈아가며 표시



참고 사항

LCD 화면에 빛은 들어오는데 글자가 보이지 않는 경우

☞ LCD에 설정된 저항값을 조절한다.



참고 사항

저항값 조절 방법

 드라이버로 LCD 뒷면의 저항값을 조절한다.

저항을 시계방향으로 돌리면 저항값이 낮아져 밝아지고,
반시계방향으로 돌리면 저항값이 높아져 어두워진다.

참고 사항

드라이버를 돌려가며 LCD에 네모(■)가 출력되는 것을 확인했다면
보드에 연결된 전원케이블을 뺐다가 다시 꽂아준다.

전원을 뺐다 꽂았을 때 네모만 출력된다면
저항값이 너무 낮아 과하게 밝은 상태이므로
전원을 연결한 채 드라이버를 돌려 글자가 잘 보이도록 조정한다.

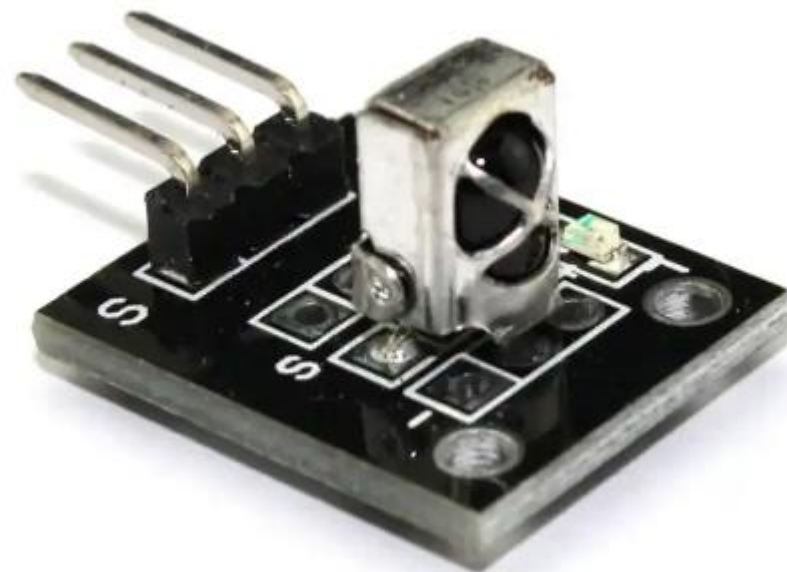
참고 사항

그래도 글자가 출력되지 않고 네모만 보인다면 'LCD모듈의 주솟값'이 잘못 설정되어 있을 수 있다.

```
LiquidCrystal_I2C lcd(주솟값, 16, 2);
```

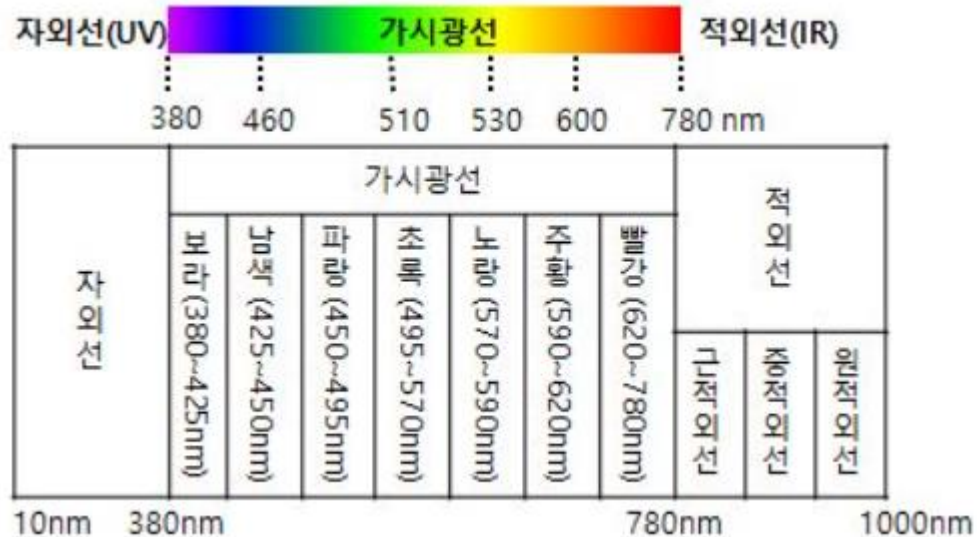
위 코드에서 주솟값이 올바르게 기재되어 있는지 확인한다.

적외선(IR) 수신 센서



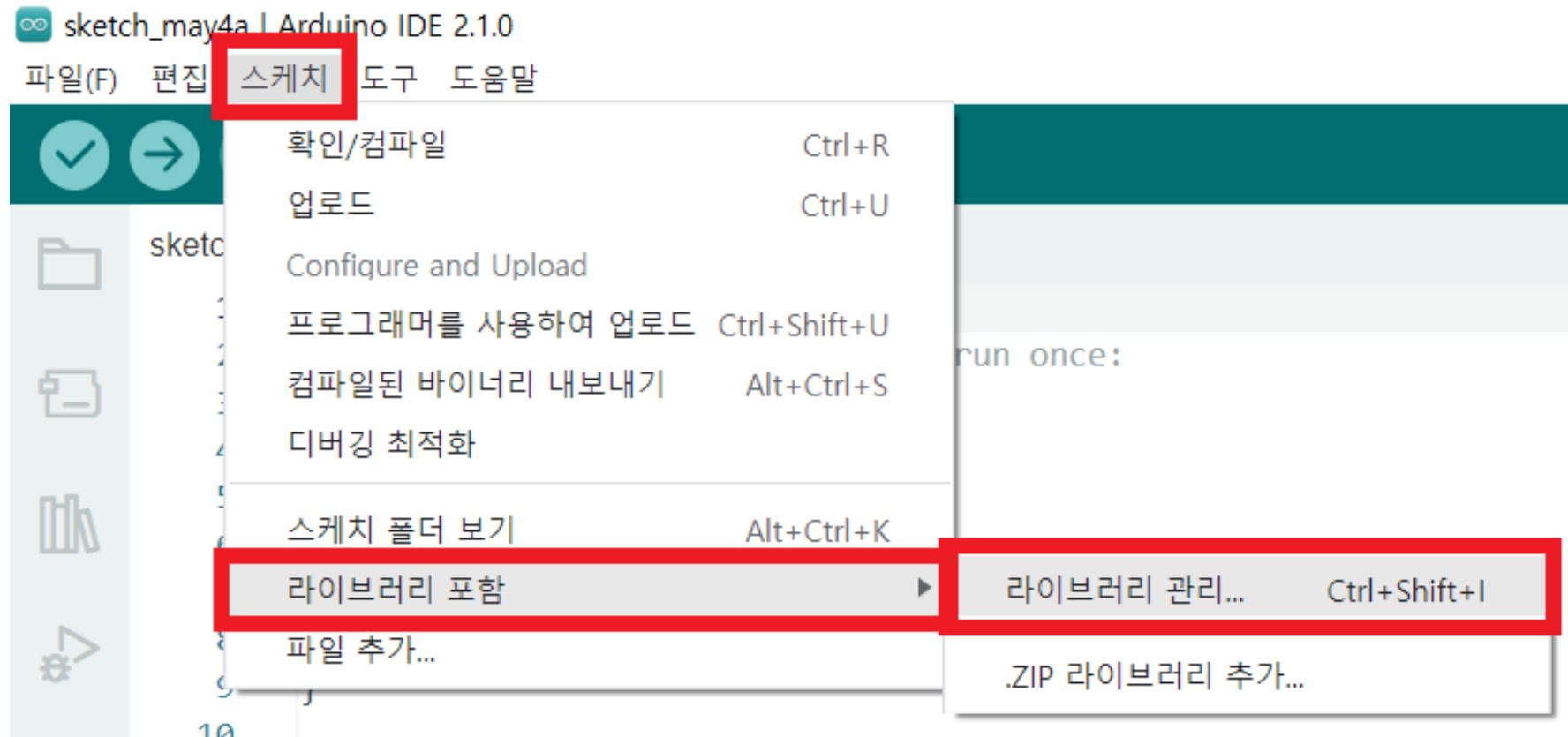
적외선(IR)

우리가 흔히 사용하는 TV 리모컨 신호는 적외선(Infrared Radiation)을 이용한다. 적외선을 방사하는 전력에 따라 적외선 신호의 세기도 달라지므로 이에 맞춰 수신 거리가 정해지지만 통상 10m 내외이며, 빛을 이용하므로 수신 센서를 향해 방향을 유지해야 한다.



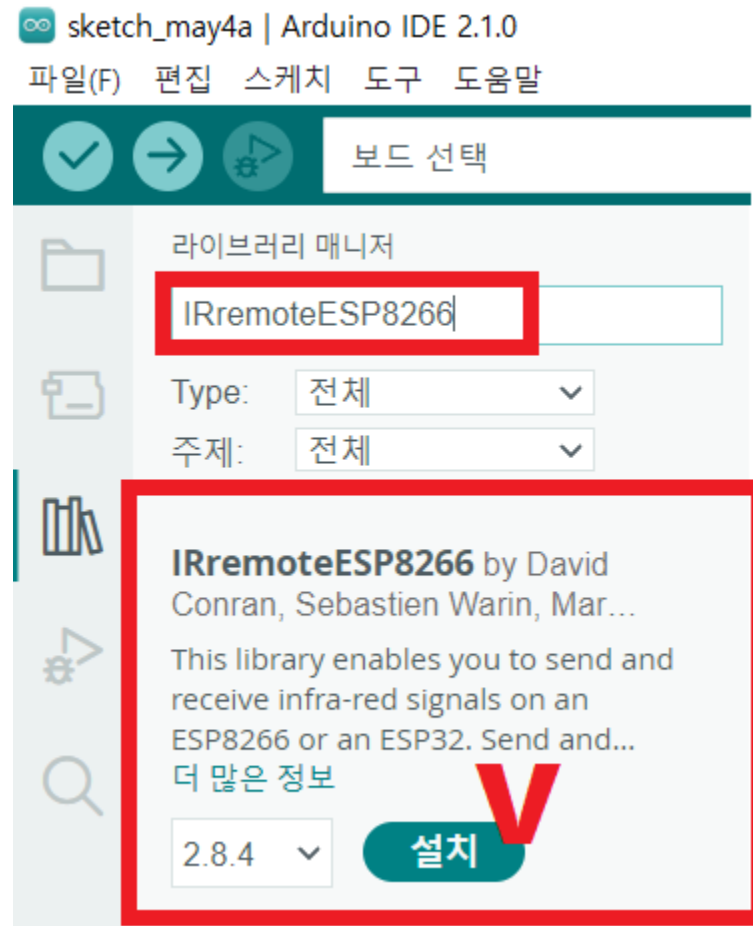
'IRremoteESP8266' 라이브러리 설치하기

Arduino IDE > 스케치 > 라이브러리 포함 > 라이브러리 관리



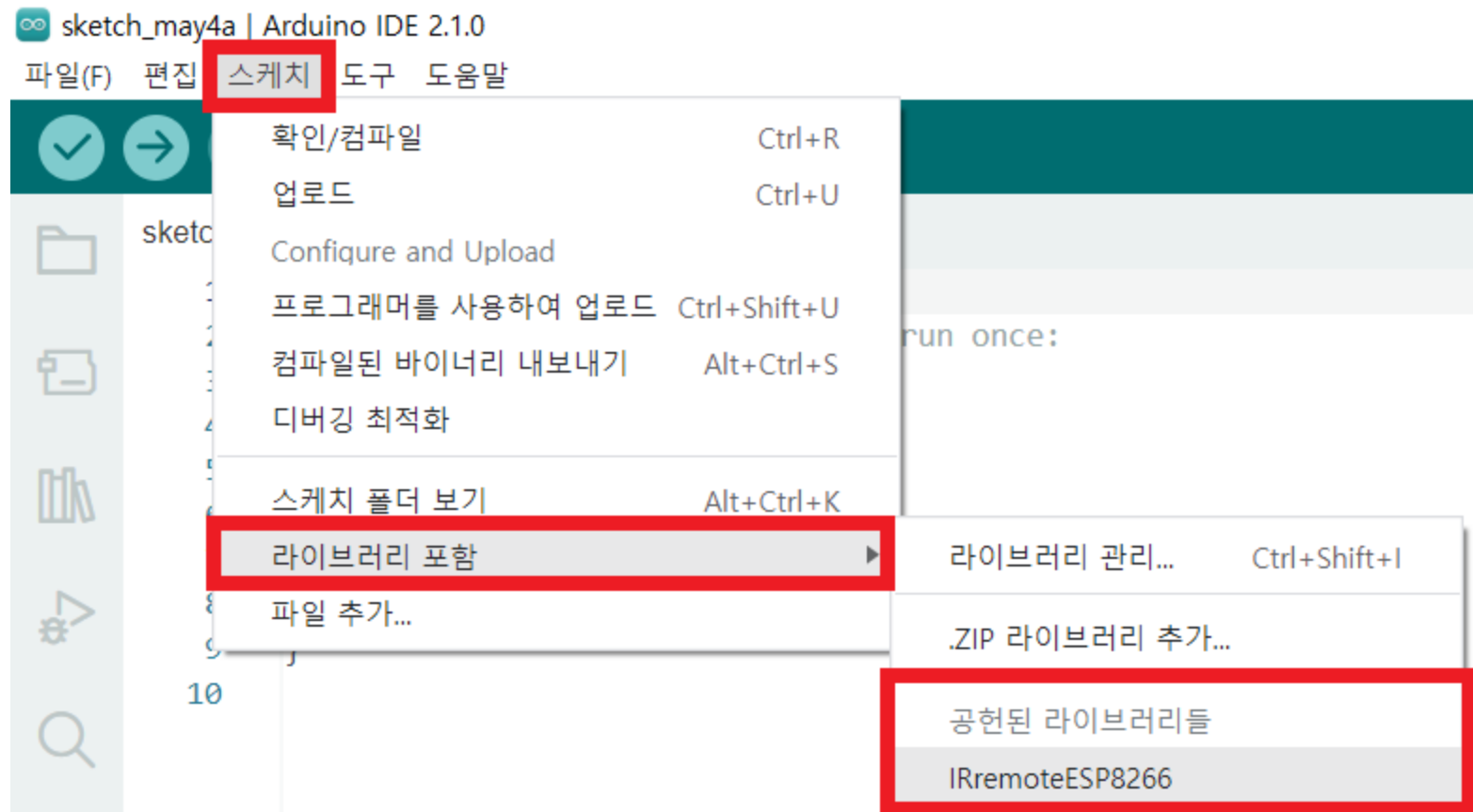
'IRremoteESP8266' 라이브러리 설치하기

라이브러리 매니저에서 'IRremoteESP8266' 검색 후 설치



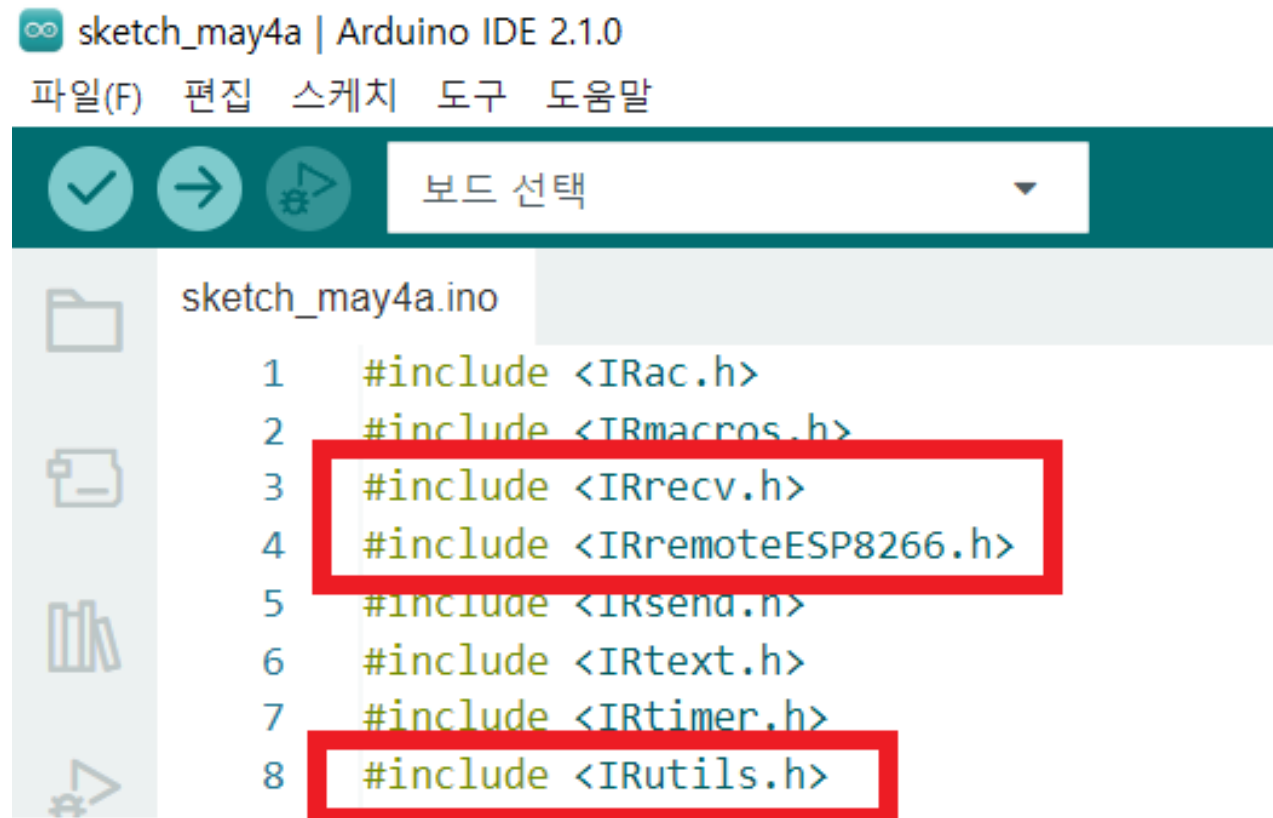
'IRremoteESP8266' 라이브러리 설치하기

스케치 > 라이브러리 포함 > 공헌된 라이브러리들에서 'IRremoteESP8266' 선택



'IRremoteESP8266' 라이브러리 설치하기

라이브러리가 정상적으로 설치되어 코드에 반영되었는지 확인



```
sketch_may4a | Arduino IDE 2.1.0
파일(F) 편집 스케치 도구 도움말

[확인] [다음] [업로드] 보드 선택

sketch_may4a.ino
1  #include <IRac.h>
2  #include <IRmacros.h>
3  #include <IRrecv.h>
4  #include <IRremoteESP8266.h>
5  #include <IRsend.h>
6  #include <IRtext.h>
7  #include <IRtimer.h>
8  #include <IRutils.h>
```


적외선 수신 센서 예제

```
#include <Arduino.h>
#include <IRremoteEPS8266.h>
#include <IRrecv.h>
#include <IRutils.h>

IRrecv irrecv(14);           // 14번 핀에 적외선 수신 센서 연결
decode_results results;

...
```

적외선 수신 센서 예제

...

```
void setup(){  
    Serial.begin(115200);  
    irrecv.enableIRIn();  
    while(!Serial){  
        delay(50);  
    }  
    Serial.println();  
}
```

...

적외선 수신 센서 예제

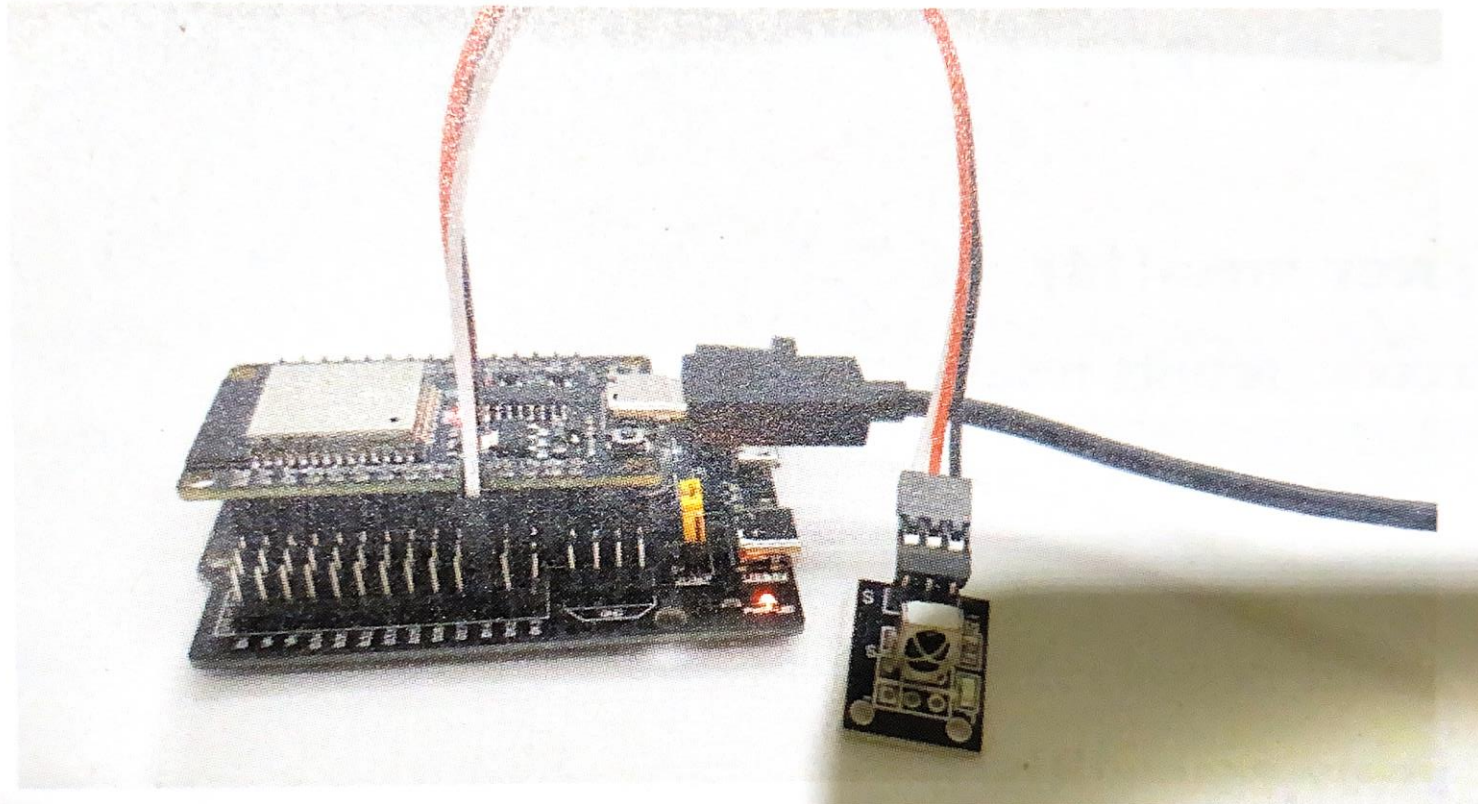
...

```
void loop(){  
    if(irrecv.decode(&results)){  
        serialPrintUnit64(results.value, DEC);  
        Serial.println("");  
        irrecv.resume();  
    }  
    delay(100);  
}
```

10진수 표기: DEC
16진수 표기: HEX
2진수 표기: BIN

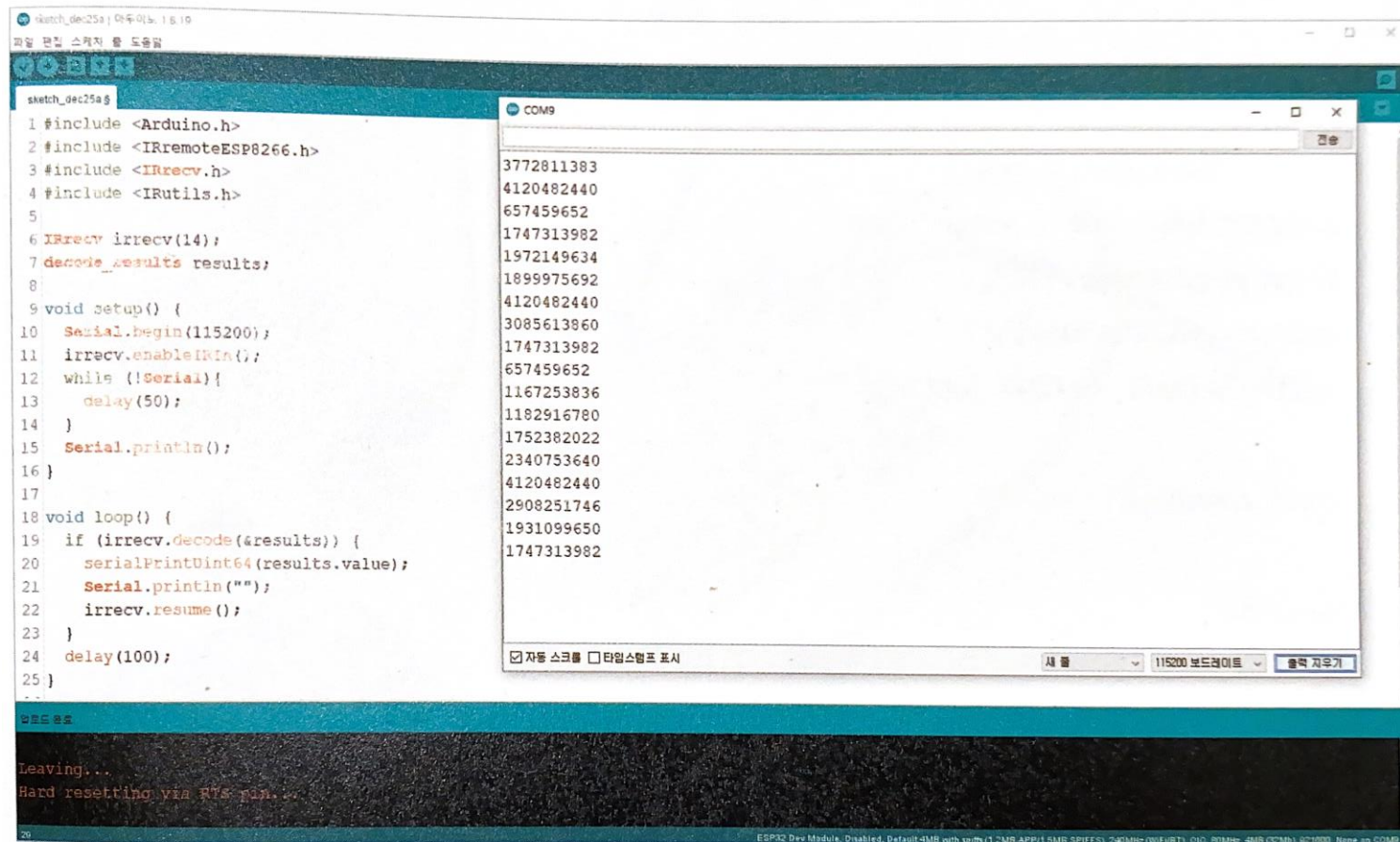
회로 구성

14번 핀에 적외선 수신 센서를 연결



적외선 수신 센서 예제 실행 결과(예)

시리얼 모니터를 열고 리모컨을 누르면 시리얼 모니터에 수신된 적외선 신호가 표시



The screenshot shows the Arduino IDE interface. The sketch editor on the left contains the following code:

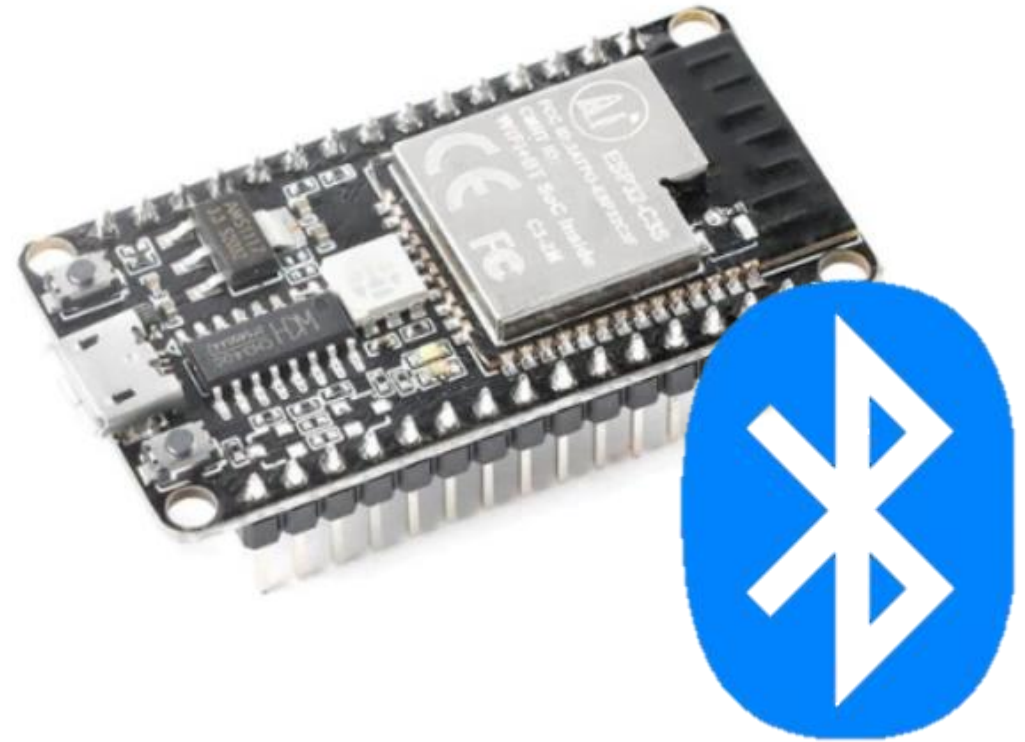
```
1 #include <Arduino.h>
2 #include <IRremoteESP8266.h>
3 #include <IRrecv.h>
4 #include <IRutils.h>
5
6 IRrecv irrecv(14);
7 decode_results results;
8
9 void setup() {
10   Serial.begin(115200);
11   irrecv.enableIRIn();
12   while (!Serial){
13     delay(50);
14   }
15   Serial.println();
16 }
17
18 void loop() {
19   if (irrecv.decode(&results)) {
20     SerialPrintUInt64(results.value);
21     Serial.println("");
22     irrecv.resume();
23   }
24   delay(100);
25 }
```

The serial monitor on the right, titled 'COM9', displays a list of hexadecimal values received from the IR receiver:

```
3772811383
4120482440
657459652
1747313982
1972149634
1899975692
4120482440
3085613860
1747313982
657459652
1167253836
1182916780
1752382022
2340753640
4120482440
2908251746
1931099650
1747313982
```

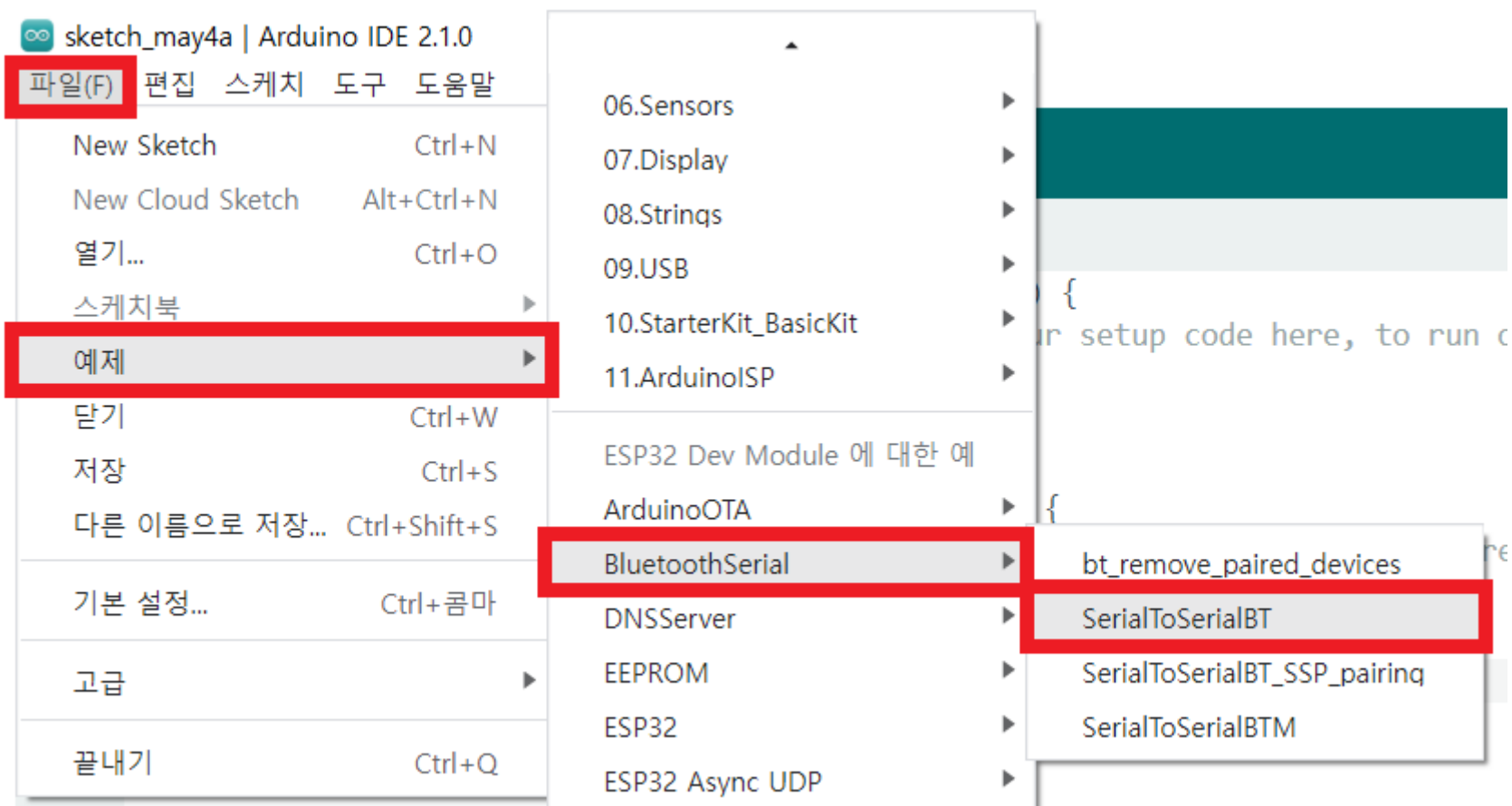
At the bottom of the IDE, the status bar indicates 'ESP32 Dev Module, Disabled, Default-4MB with 1.3MB APP+1.5MB SPIFFS, 240MHz (WiFi/BT), QIO, 80MHz, 4MB (32Mb) 521000, None on COM9'.

블루투스 (Bluetooth)



ESP32 Dev Module의 블루투스 예제

파일 > 예제 > BluetoothSerial > SerialToSerialBT



SerialToSerialBT | Arduino IDE 2.1.0

파일(F) 편집 스케치 도구 도움말

ESP32 Dev Module

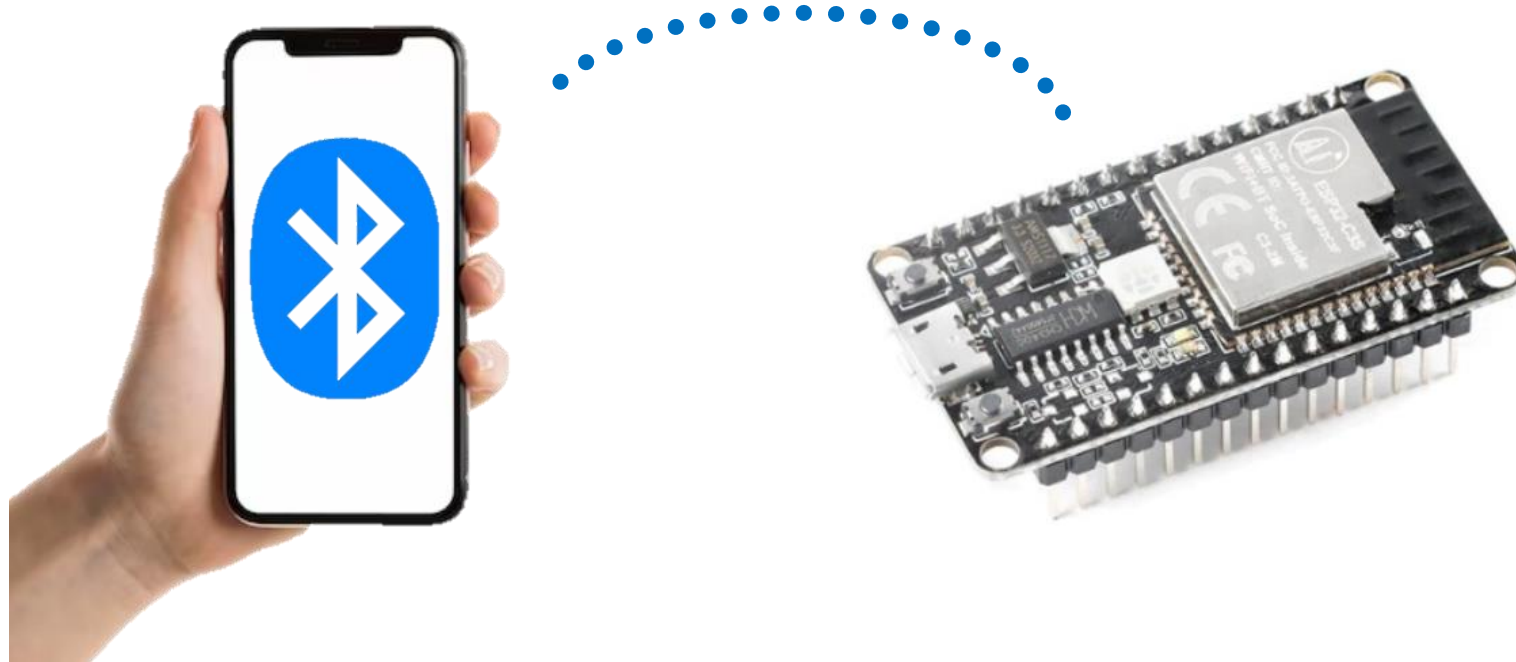
SerialToSerialBT.ino

```
1 //This example code is in the Public Domain (or CC0 licensed, at your option.)
2 //By Evandro Copercini - 2018
3 //
4 //This example creates a bridge between Serial and Classical Bluetooth (SPP)
5 //and also demonstrate that SerialBT have the same functionalities of a normal Serial
6
7 #include "BluetoothSerial.h"
8
9 #if !defined(CONFIG_BT_ENABLED) || !defined(CONFIG_BLUEDROID_ENABLED)
10 #error Bluetooth is not enabled! Please run `make menuconfig` to and enable it
11 #endif
12
13 BluetoothSerial SerialBT;
14
15 void setup() {
16   Serial.begin(115200);
17   SerialBT.begin("ESP32test"); //Bluetooth device name
18   Serial.println("The device started, now you can pair it with bluetooth!");
19 }
20
21 void loop() {
22   if (Serial.available()) {
23     SerialBT.write(Serial.read());
24   }
25   if (SerialBT.available()) {
26     Serial.write(SerialBT.read());
27   }
28   delay(20);
29 }
```

줄 1, 열 79 ESP32 Dev Module [연결되지 않음]

스마트폰과 블루투스 통신하기

Arduino, ESP32와 스마트폰 간의 블루투스 통신 방법은 크게 두 가지로 구분할 수 있다. SPP(Serial Port Profile) 방식과 BLE(Bluetooth Low Energy) 방식으로, 일반적으로는 SPP 방식을 사용한다. 이는 무선으로 상호간에 시리얼 통신을 하는 방식이다.



블루투스 시리얼 통신 예제

```
#include "BluetoothSerial.h"  
BluetoothSerial BT;
```

```
void setup(){  
    Serial.begin(115200);  
    BT.begin("ESP32_jw");  
}
```

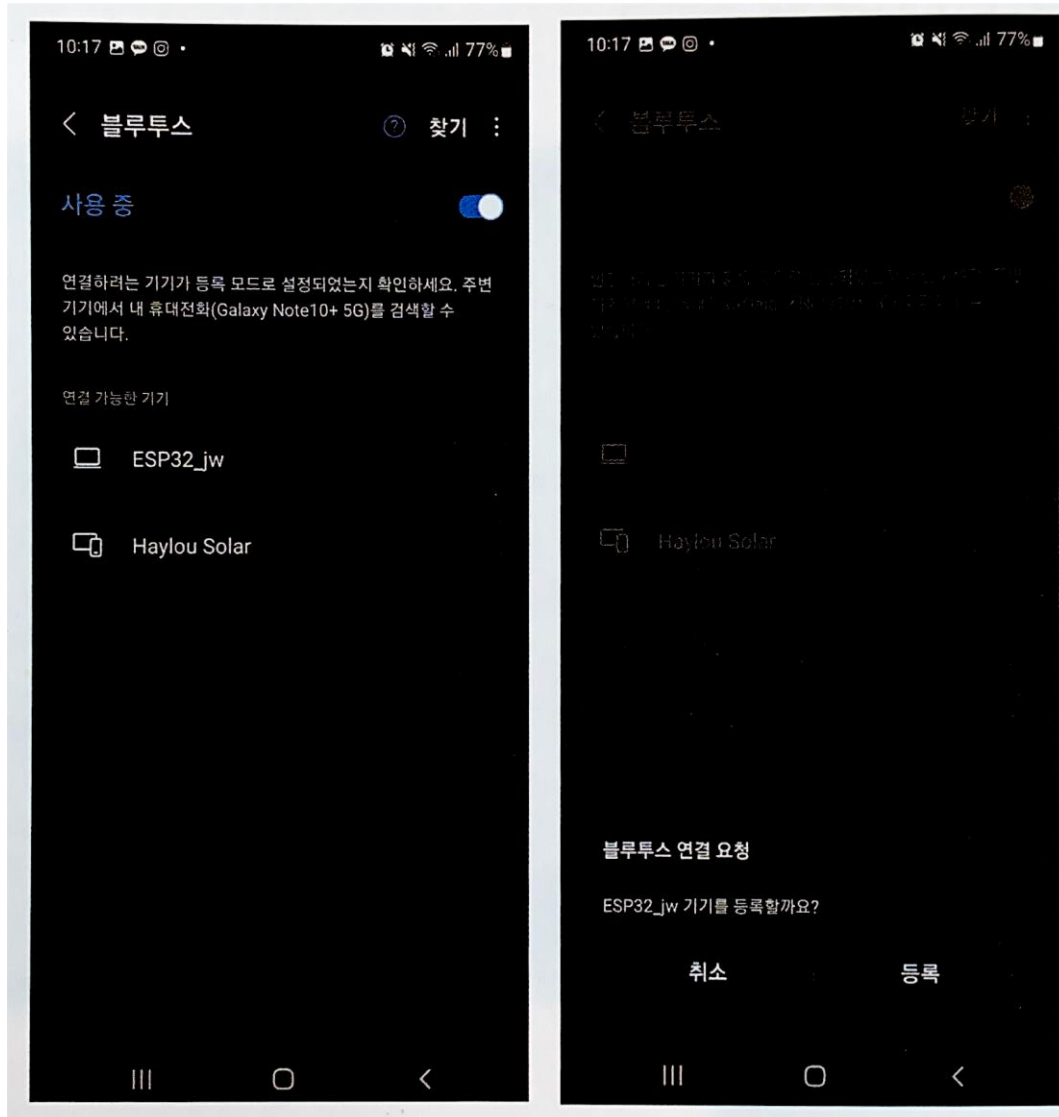
```
...
```

// 블루투스의 이름 설정

블루투스 시리얼 통신 예제

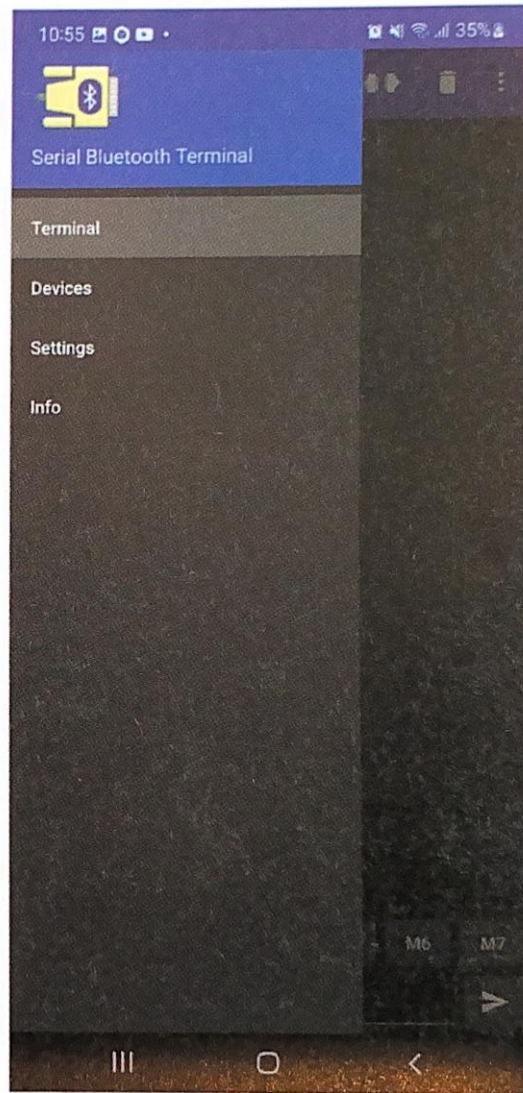
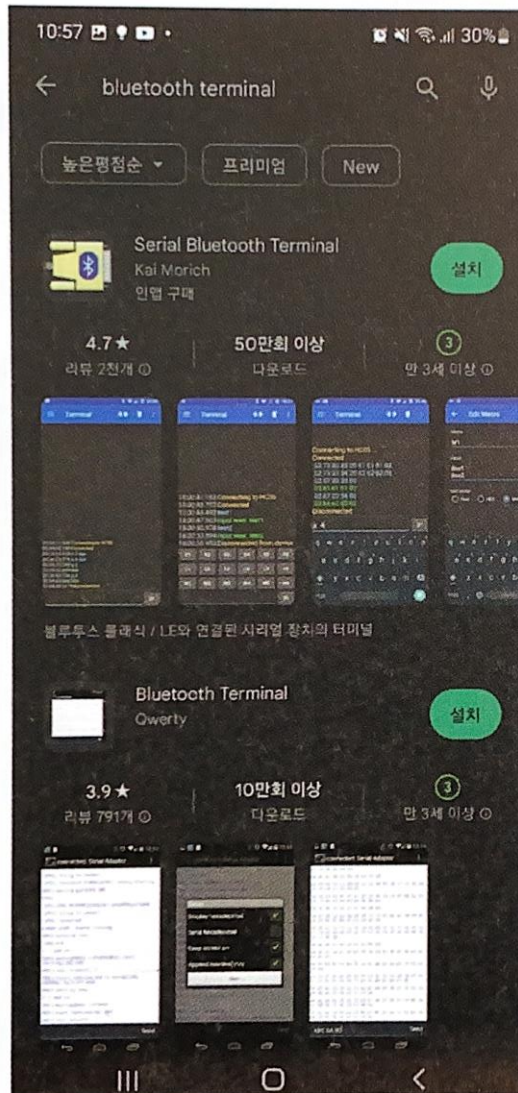
```
...  
void loop(){  
    if(Serial.available()){  
        BT.write(Serial.read());  
    }  
    if(BT.available()){  
        Serial.write(BT.read());  
    }  
    delay(20);  
}
```

블루투스 시리얼 통신 예제



앞의 코드를 작성하여 업로드한 후
스마트폰의 블루투스 기능을 켜고
[연결 가능한 기기] 중 'ESP32_jw'를
선택하여 페어링한다.

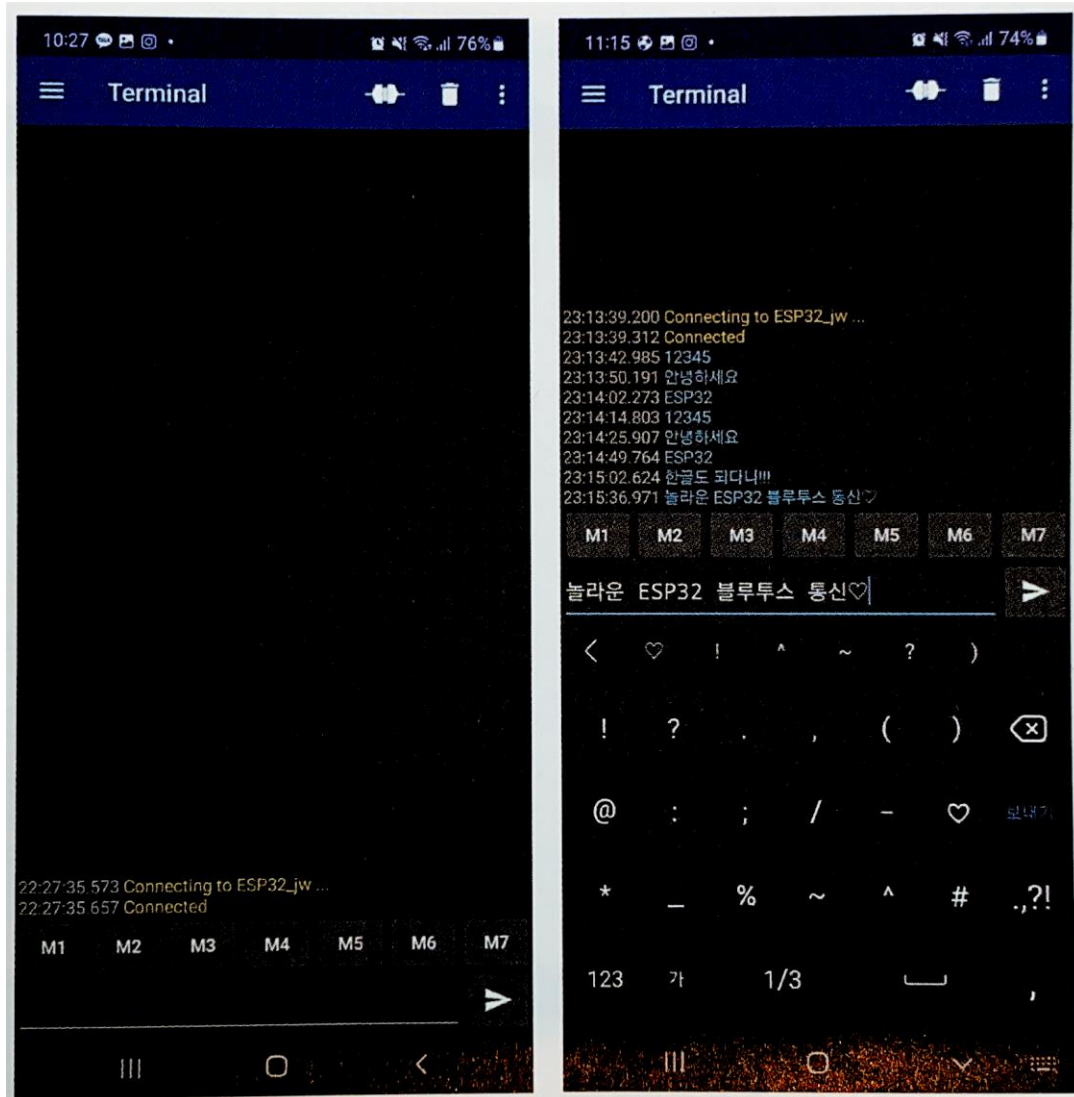
블루투스 시리얼 통신 예제



ESP32와 스마트폰 간 통신을 위한
애플리케이션을 설치한다.

(예) Serial Bluetooth Terminal

블루투스 시리얼 통신 예제



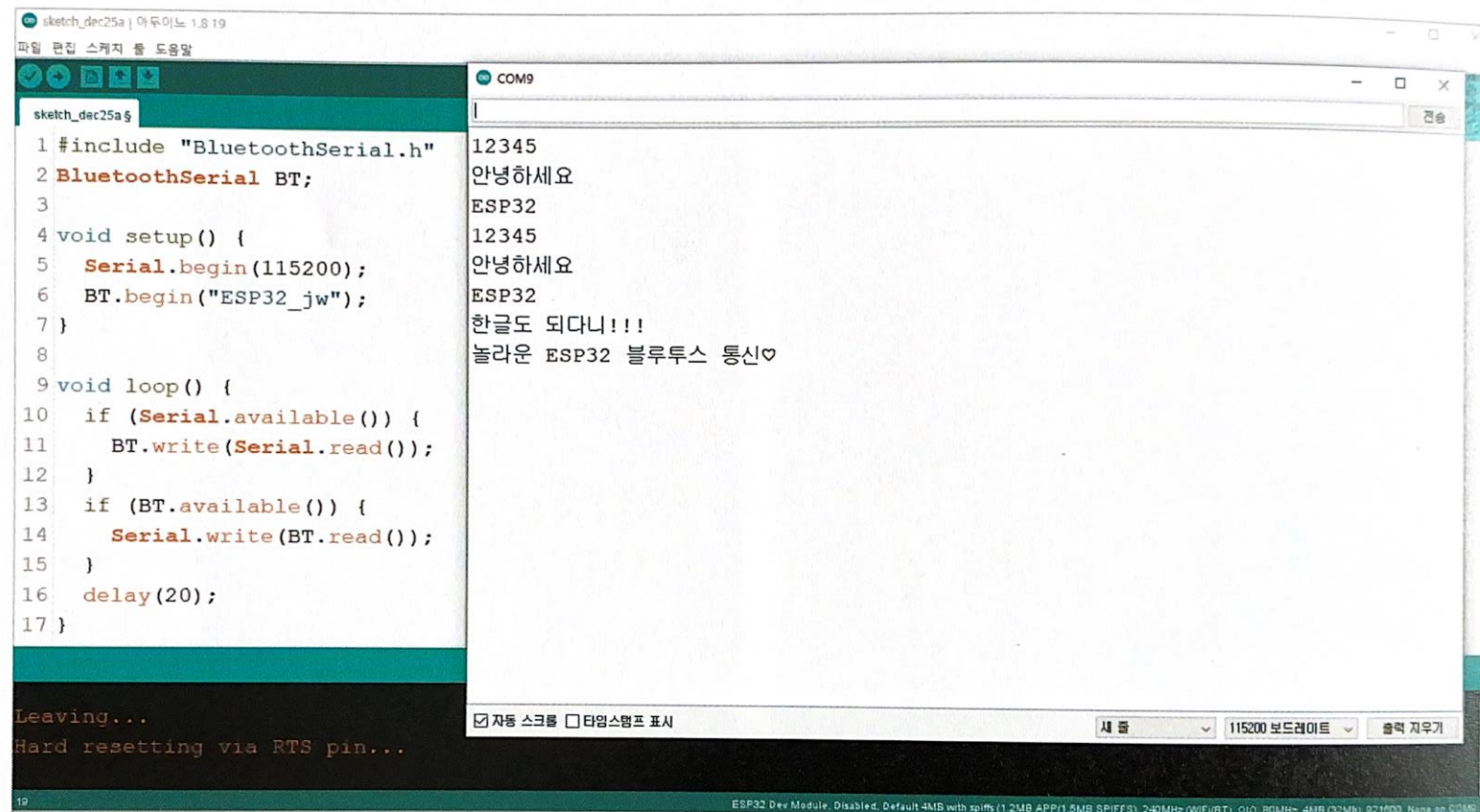
좌측 메뉴(三) > Device 선택 후
페어링된 'ESP32_jw' 선택하여 연결

연결이 완료되면 화면에 'Connected'
라는 메시지가 표시된다.

블루투스 시리얼 통신 예제 실행 결과

스마트폰에서 문자를 전송하면 시리얼 모니터에 해당 메시지가 표시된다.

(※ 터미널 프로그램(앱)에 따라 한글 표현이 불가능할 수 있음)



The screenshot displays the Arduino IDE interface. The left pane shows a sketch named 'sketch_dec25a' with the following code:

```
1 #include "BluetoothSerial.h"
2 BluetoothSerial BT;
3
4 void setup() {
5   Serial.begin(115200);
6   BT.begin("ESP32_jw");
7 }
8
9 void loop() {
10  if (Serial.available()) {
11    BT.write(Serial.read());
12  }
13  if (BT.available()) {
14    Serial.write(BT.read());
15  }
16  delay(20);
17 }
```

The right pane shows the serial monitor for COM9, displaying the received data:

```
12345
안녕하세요
ESP32
12345
안녕하세요
ESP32
한글도 되다니!!!
놀라운 ESP32 블루투스 통신♡
```

At the bottom of the IDE, status messages indicate 'Leaving...' and 'Hard resetting via RTS pin...'. The bottom status bar shows 'ESP32 Dev Module, Disabled, Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS), 240MHz (WiFi/BT), QIO, 80MHz, 4MB (32Mb), 921600, Nana on COM9'.

참고 사항

ESP32에는 블루투스 기능이 내장되어 있어 'BluetoothSerial.h' 라이브러리도 기본으로 설치되어 있다.

또한 블루투스를 특정 핀에 연결하지 않고 사용할 수 있는 핀이 지정되어 있어 rx, tx 핀을 지정하는 코드가 없으며, 통신속도(Baud)도 정해져 있어 별도의 지정은 하지 않는다.

와이파이
(Wi-Fi)



ESP32와 Wi-Fi

ESP32에는 Wi-Fi를 이용할 수 있는 모듈이 탑재되어 있다. 해당 모듈을 활용하여 무선 인터넷을 사용한 데이터 송수신 기능을 구현해보자.

ESP의 Wi-Fi 모드는 `WiFi.mode()`를 통해 설정하며, `WIFI_STA`와 `WIFI_AP`의 두 가지 모드가 존재한다. 보통 `WIFI_STA`(station mode, stand-alone mode)가 사용되며 `WIFI_AP`는 ESP32 옵션을 변경하는 데 사용된다.

Wi-Fi 통신 예제(1) – 무선인터넷 찾기

```
#include "WiFi.h"

void setup(){
    Serial.begin(115200);
    WiFi.mode(WIFI_STA);
    WiFi.disconnect();           // 사전에 연결된 WiFi 접속을 해제
    delay(100);
}

...
```

Wi-Fi 통신 예제(1) – 무선인터넷 찾기

```
...

void loop(){
    int n = WiFi.scanNetworks();  // 주변의 AP 수를 알려주는 함수
    if(n==0){
        Serial.println("WiFi AP가 없음");
    }
    else{
        Serial.print(n);
        Serial.println("개의 WiFi AP를 찾음");
        ...
    }
}
```

Wi-Fi 통신 예제(1) – 무선인터넷 찾기

```
...
for(int i=0 ; i<n ; ++i){
    Serial.print(i+1);
    Serial.print(": ");
    Serial.println(WiFi.SSID(i));    // AP의 이름을 출력
    delay(10);
}
}
Serial.println("");
delay(5000);                        // 5초마다 주변을 스캔하여 출력
}
```

실행 결과

시리얼 모니터를 통해 주변의 WiFi AP를 찾아 AP 이름을 출력

```
1 #include "WiFi.h"
2
3 void setup() {
4   Serial.begin(115200);
5   WiFi.mode(WIFI_STA);
6   WiFi.disconnect();
7   delay(100);
8 }
9
10 void loop() {
11   int n = WiFi.scanNetworks();
12   if (n == 0) {
13     Serial.println("WiFi AP가 없음");
14   } else {
15     Serial.print(n);
16     Serial.println("개의 WiFi AP를 찾음");
17     for (int i = 0; i < n; ++i) {
18       Serial.print(i + 1);
19       Serial.print(": ");
20       Serial.println(WiFi.SSID(i));
21       delay(10);
22     }
23   }
24   Serial.println("");
25   delay(5000);
26 }
```

16개의 WiFi AP를 찾음
1: kkh8959
2: DIRECT-19 C51x Series
3: U+NetC050
4: SK VIEW 506 1704
5: U+Net98D9
6: KT_GiGA_2G_D5F4
7: SK_WiFiGIGA174F_2.4G
8: KT_GiGA_0588
9: olleh_WiFi_B63A
10: 5061504
11: SK_WiFiGIGA89DC_2.4G
12: dodo2.4
13: DIV01-07763C
14: KT_GiGA_Mesh_0588
15: KT_GiGA_324F
16: U+Net283C

Wi-Fi 통신 예제(2) – 무선인터넷 연결하기

```
#include <WiFi.h>
#include <WiFiClient.h>
#include <WebServer.h>
#include <ESPmDNS.h>
const char* ssid = "iptime";           // AP 이름: iptime
const char* password = "qwerty";      // AP 비밀번호: qwerty
WebServer server(80);

void conn(){
    server.send(200, "text/plain", Welcome ESP32 World!");
}
...
```

Wi-Fi 통신 예제(2) – 무선인터넷 연결하기

```
...  
void setup(void){  
    Serial.begin(115200);  
    WiFi.mode(ssid, password);  
    while(WiFi.status() != WL_CONNECTED){  
        delay(500);  
        Serial.print(".");  
    }  
    Serial.print("IP address: ");  
    Serial.println(WiFi.localIP());    // ESP32에 부여된 IP 출력  
    ...
```

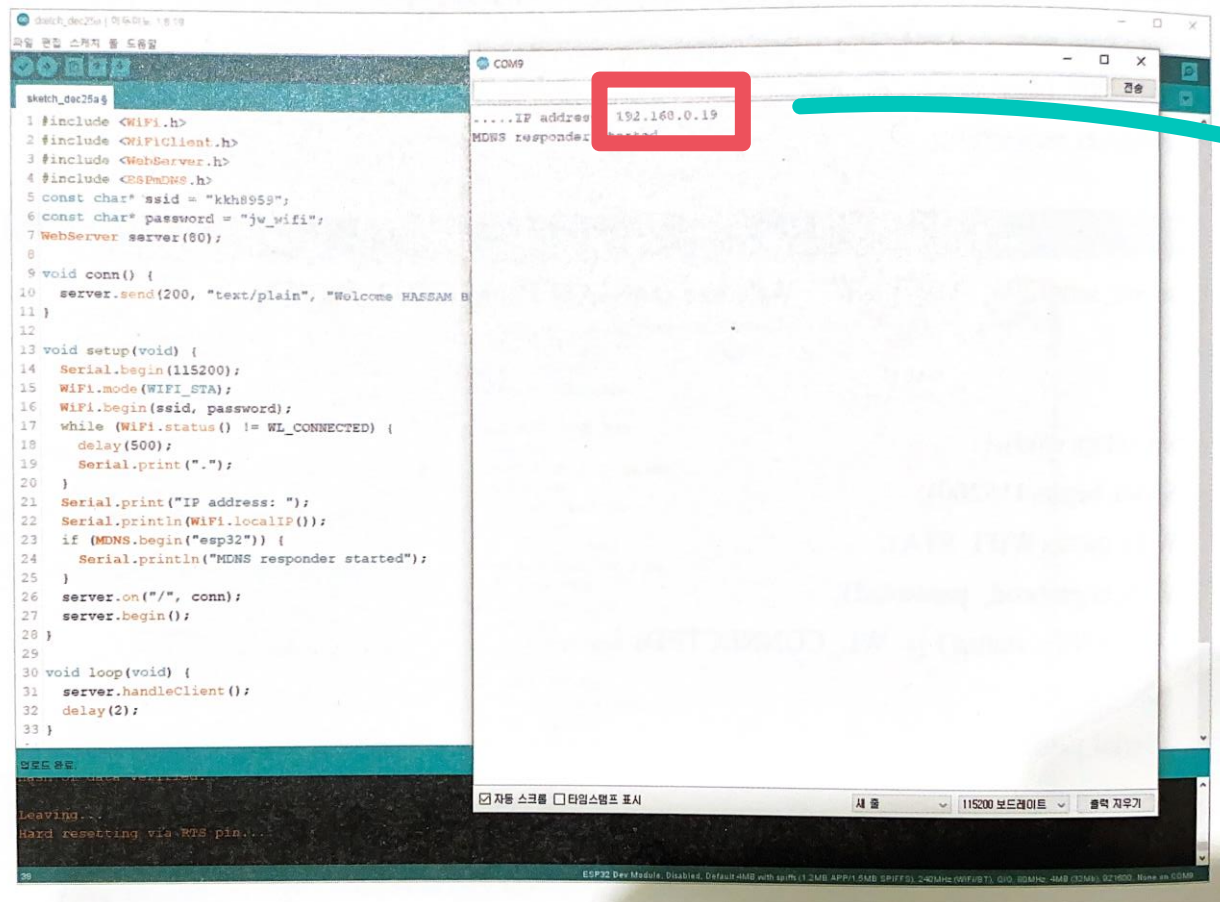

Wi-Fi 통신 예제(2) – 무선인터넷 연결하기

```
...
if(MDNS.begin("esp32")){
    Serial.println("MDNS responder started");
}
server.on("/", conn);
server.begin();
}

void loop(void){
    server.handleClient();
    delay(2);
}
```

실행 결과

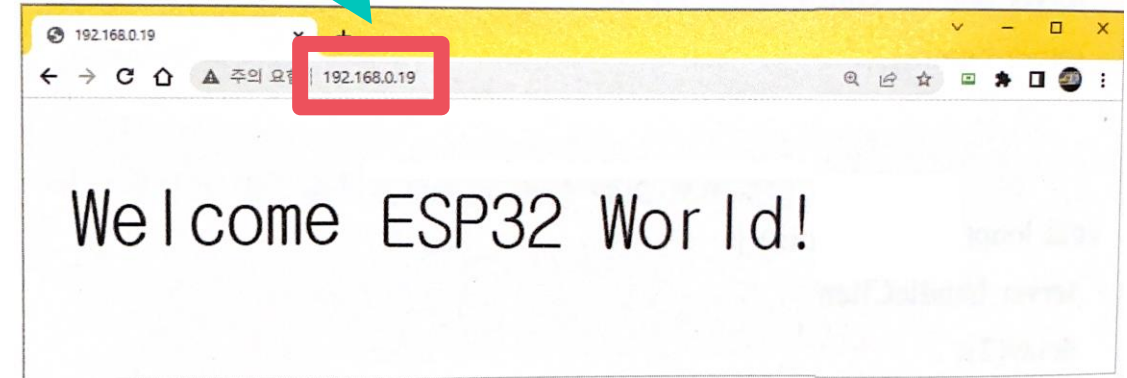
시리얼 모니터에 출력된 IP 주소를 웹 주소창에 입력하면 메시지가 출력



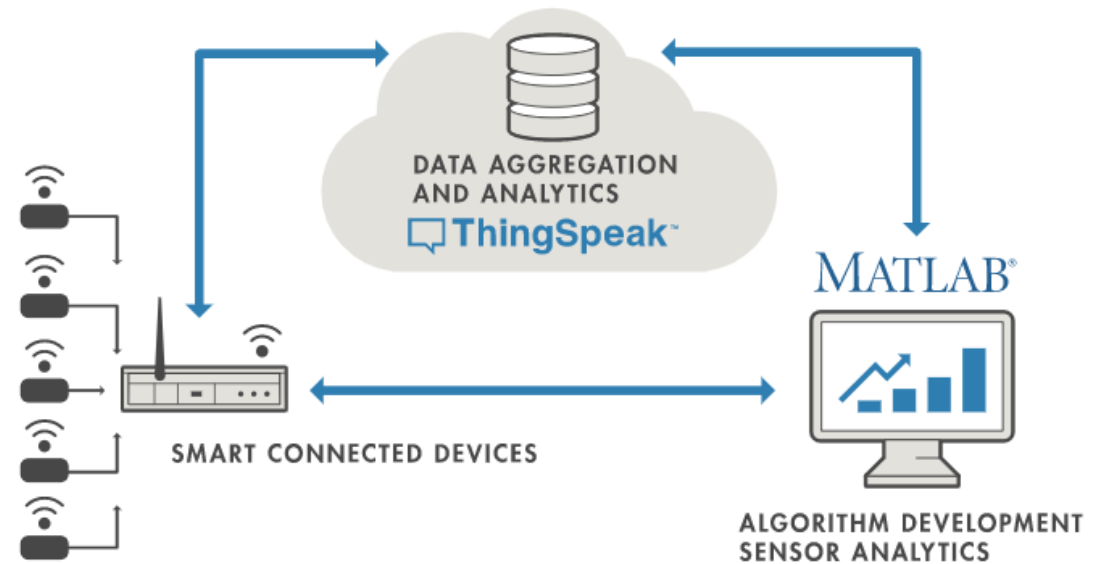
```
sketch_dec25a.g
1 #include <WiFi.h>
2 #include <WiFiClient.h>
3 #include <WebServer.h>
4 #include <ESPmDNS.h>
5 const char* ssid = "kkh8959";
6 const char* password = "jw_wifi";
7 WebServer server(80);
8
9 void conn() {
10   server.send(200, "text/plain", "Welcome HASSAM B");
11 }
12
13 void setup(void) {
14   Serial.begin(115200);
15   WiFi.mode(WIFI_STA);
16   WiFi.begin(ssid, password);
17   while (WiFi.status() != WL_CONNECTED) {
18     delay(500);
19     Serial.print(".");
20   }
21   Serial.print("IP address: ");
22   Serial.println(WiFi.localIP());
23   if (MDNS.begin("esp32")) {
24     Serial.println("MDNS responder started");
25   }
26   server.on("/", conn);
27   server.begin();
28 }
29
30 void loop(void) {
31   server.handleClient();
32   delay(2);
33 }
```

Serial Monitor Output:


```
.....IP address: 192.168.0.19
MDNS responder started
```



ThingSpeak for IoT Projects



ThingSpeak란?



The banner features a background image of a hand holding a tablet displaying various weather data visualizations. The text 'ThingSpeak for IoT Projects' is prominently displayed in white. Below it, a subtitle reads 'Data collection in the cloud with advanced data analysis using MATLAB'. Two buttons, 'Get Started For Free' and 'Learn More', are positioned at the bottom left.

ThingSpeak for IoT Projects

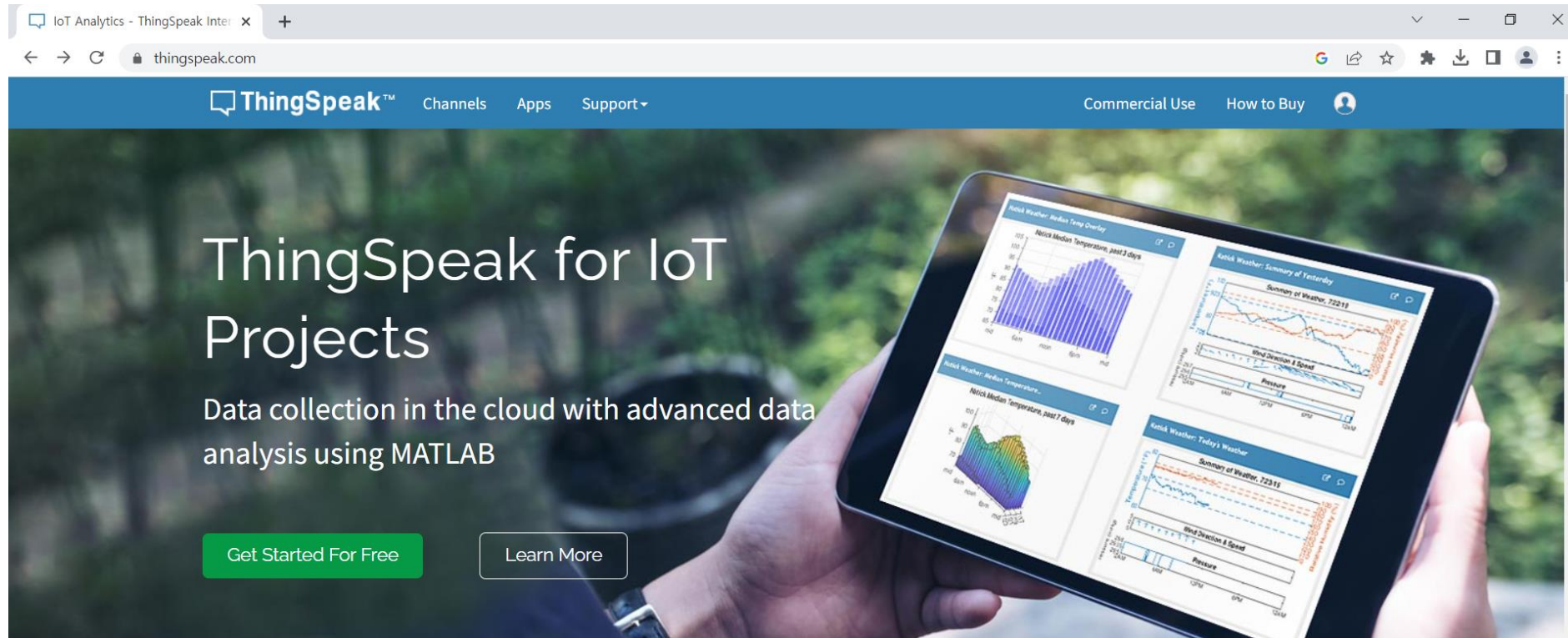
Data collection in the cloud with advanced data analysis using MATLAB

[Get Started For Free](#) [Learn More](#)

☞ 사용자가 전송하는 데이터를 기록하고 분석하여 원하는 형태의 시각화를 가능하게 하는 IoT 분석 플랫폼 서비스

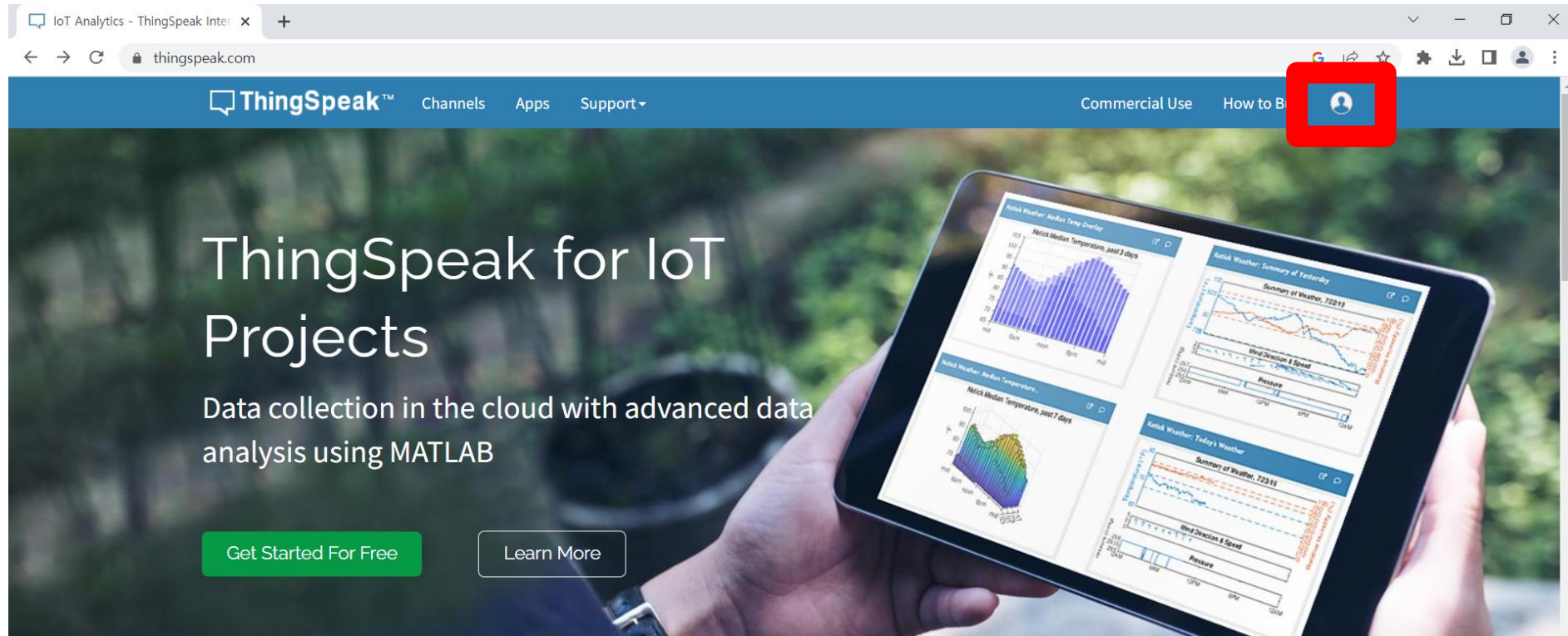
ThingSpeak 이용하기

ThingSpeak 홈페이지(<https://thingspeak.com>) 접속



ThingSpeak 사용하기

사이트 우측 상단의  아이콘을 클릭



ThingSpeak 사용하기

가입된 계정이 없는 경우 'Create one!'을 클릭하여 회원가입

The screenshot shows the ThingSpeak login page in a web browser. The browser's address bar displays the URL `thingspeak.com/login?skipSSOCheck=true`. The page header includes the ThingSpeak logo and navigation links for Channels, Apps, Support, Commercial Use, and How to Buy. The main content area contains instructions for signing in with a MathWorks account or creating a new one. Below the text, there is a MathWorks logo, an email input field, and a red box highlighting the "Create one!" link. A "Next" button is located at the bottom right of the login form.

To use ThingSpeak, you must sign in with your existing MathWorks account or create a new one.

Non-commercial users may use ThingSpeak for free. Free accounts offer limits on certain functionality. Commercial users are eligible for a time-limited free evaluation. To get full access to the MATLAB analysis features on ThingSpeak, log in to ThingSpeak using the email address associated with your university or organization.

To send data faster to ThingSpeak or to send more data from more devices, consider the [paid license options](#) for commercial, academic, home and student usage.

MathWorks®

Email

No account? **Create one!**

By signing up, you agree to our [privacy policy](#).

Next

The diagram illustrates the ThingSpeak architecture. It shows a cloud labeled "DATA AGGREGATION AND ANALYTICS" with the ThingSpeak logo. Arrows indicate data flow from "SMART CONNECTED DEVICES" (represented by a group of wireless sensors) to the cloud, and from the cloud to a "MATLAB" computer icon. Below the MATLAB icon, the text "ALGORITHM DEVELOPMENT SENSOR ANALYTICS" is displayed.

ThingSpeak 이용하기

정보 기입 후 'Continue' 클릭하여 회원가입 완료하기

Sign In - ThingSpeak IoT

thingspeak.com/login?skipSSOCheck=true

ThingSpeak™ Channels Apps Support- Commercial Use How to Buy

To use ThingSpeak, you must sign in with your existing MathWorks account or create a new one.

Non-commercial users may use ThingSpeak for free. Free accounts offer limits on certain functionality. Commercial users are eligible for a time-limited free evaluation. To get full access to the MATLAB analysis features on ThingSpeak, log in to ThingSpeak using the email address associated with your university or organization.

To send data faster to ThingSpeak or to send more data from more devices, consider the [paid license options](#) for commercial, academic, home and student usage.

Create MathWorks Account

Email Address

honggildong@gmail.com

Location

Korea (Republic Of)

Last Name

홍

First Name

길동

Last Name (English)

Hong

First Name (English)

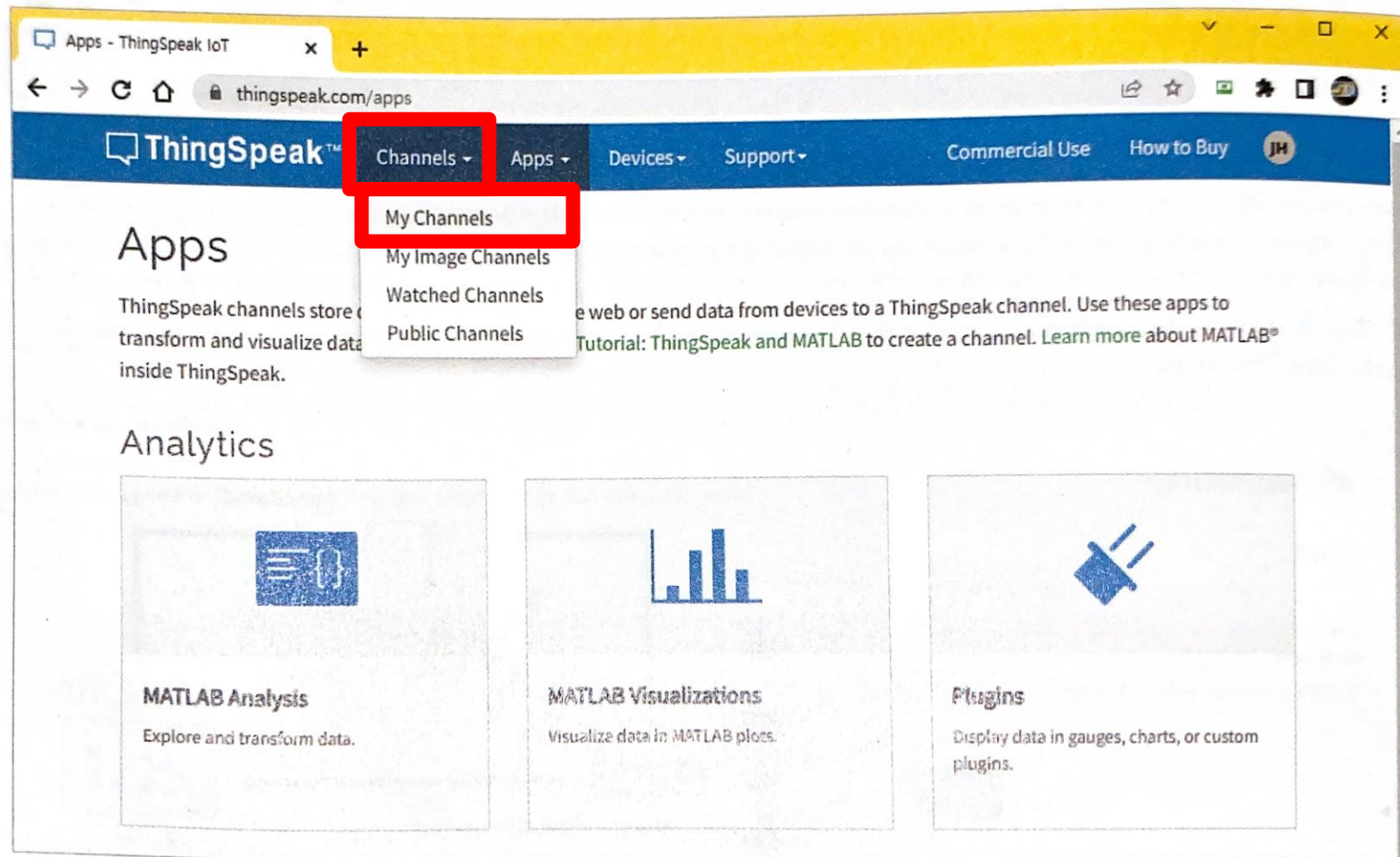
Gildong

Continue

Cancel

ThingSpeak 사용하기

로그인 후 'Channels' > 'My Channels' 클릭



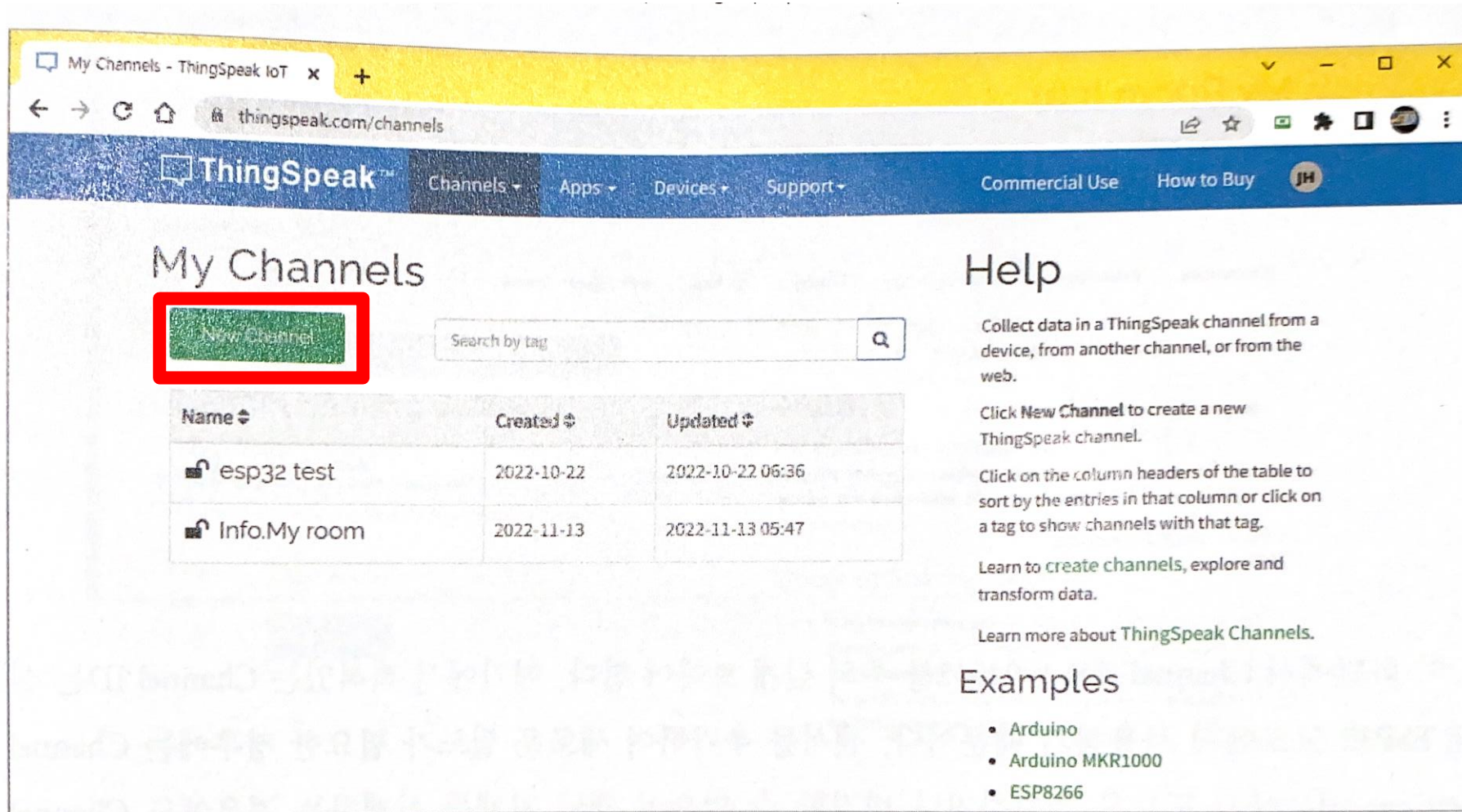
ThingSpeak 이용하기

※ 로그인을 했는데도 'Channels' 메뉴가 나타나지 않는 경우

회원 가입 후 하루 정도 기다리면 해당 메뉴가 정상적으로 나타납니다.

ThingSpeak 이용하기

‘New Channel’을 클릭하여 새로운 채널 만들기 (*채널: 데이터 기록 공간)



The screenshot shows the 'My Channels' page on the ThingSpeak website. The browser address bar shows 'thingspeak.com/channels'. The page has a blue header with the ThingSpeak logo and navigation links: Channels, Apps, Devices, Support, Commercial Use, and How to Buy. The main content area is titled 'My Channels' and features a green 'New Channel' button, which is highlighted with a red rectangle. To the right of the button is a search bar labeled 'Search by tag'. Below the button is a table with two columns: 'Name' and 'Created'. The table lists two channels: 'esp32 test' and 'Info.My room'. To the right of the table is a 'Help' section with instructions on how to create a new channel and how to sort the table. Below the 'Help' section is an 'Examples' section with a list of links to Arduino, Arduino MKR1000, and ESP8266.

Name	Created	Updated
esp32 test	2022-10-22	2022-10-22 06:36
Info.My room	2022-11-13	2022-11-13 05:47

Help

Collect data in a ThingSpeak channel from a device, from another channel, or from the web.

Click **New Channel** to create a new ThingSpeak channel.

Click on the column headers of the table to sort by the entries in that column or click on a tag to show channels with that tag.

Learn to **create channels**, explore and transform data.

Learn more about **ThingSpeak Channels**.

Examples

- Arduino
- Arduino MKR1000
- ESP8266

ThingSpeak 사용하기

Name을 입력하고 실제 데이터를 기록하기 위한 Field를 생성한다.

The screenshot shows the 'New Channel' form on the ThingSpeak website. The form includes fields for Name, Description, and a list of Fields. A red rectangle highlights the 'Field 1' and 'Field 2' sections. Two teal speech bubbles provide additional context: the first points to 'Field 1' (Light) and the second points to 'Field 2' (Ultrasonic).

Field 1: Light ☒

Field 2: Ultrasonic ☒

Field 3: ☐

Field 4: ☐

Field 5: ☐

Field 6: ☐

Annotations:

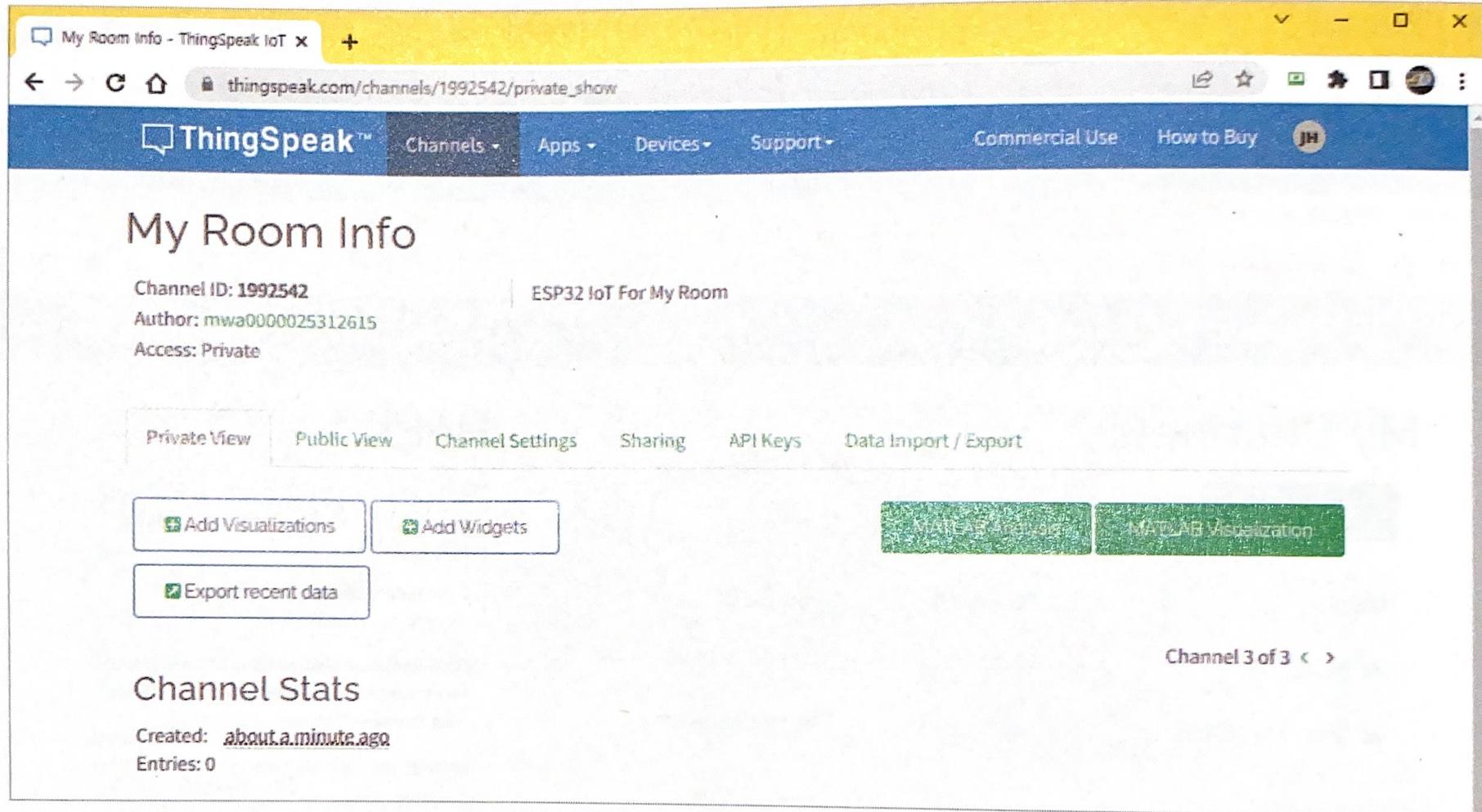
- 조도 센서를 통해 값을 입력 받는 Field 1 (Light)
- 초음파 센서를 통해 값을 입력 받는 Field 2 (Ultrasonic)

Additional Form Fields:

- Percentage complete:** Calculated based on data entered into the various fields of a channel. Enter the name, description, location,
- Tags:** Enter keywords that identify the channel. Separate tags with commas.

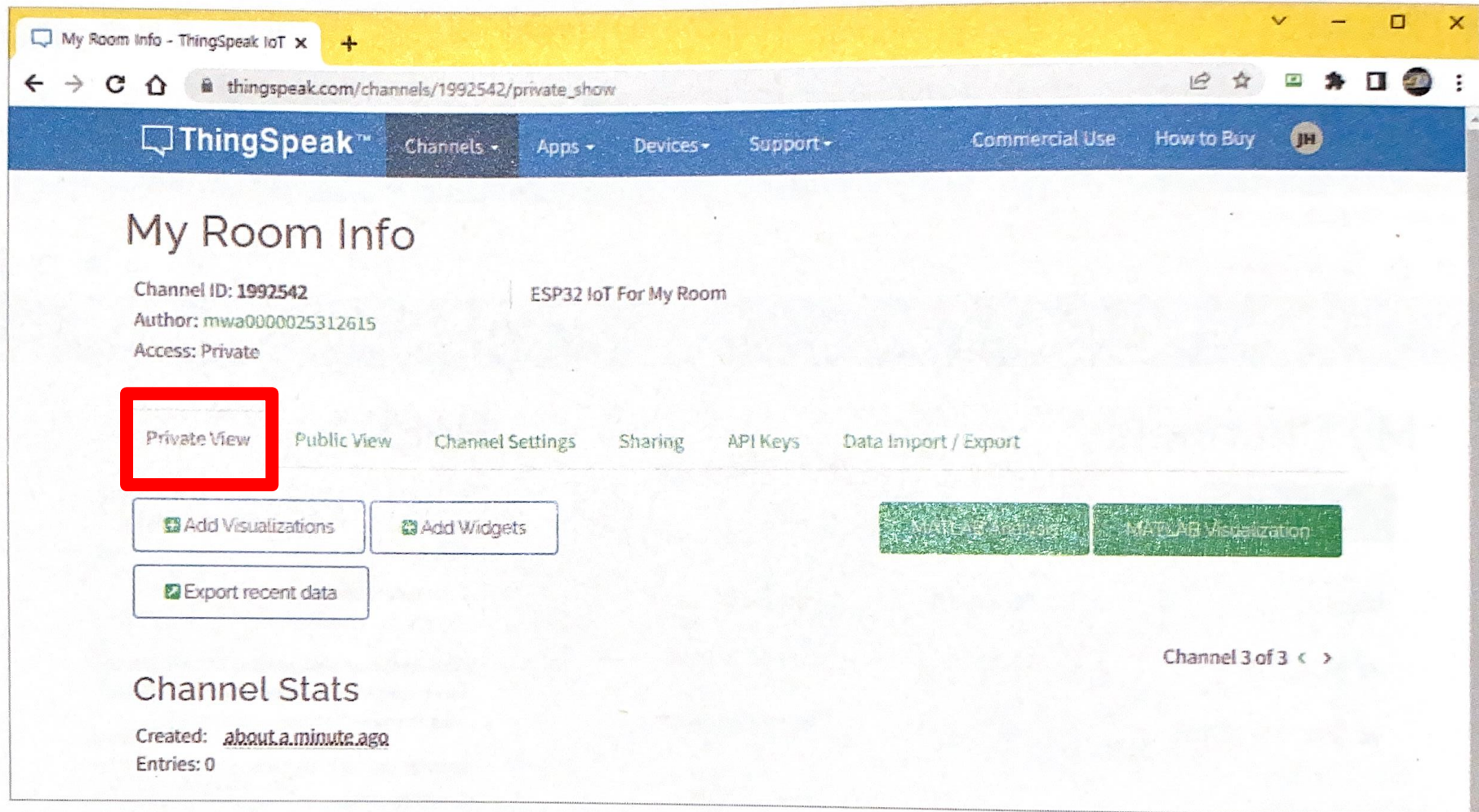
ThingSpeak 이용하기

Name과 Field 정보 입력 후 'Save Channel'을 클릭하여 채널 만들기



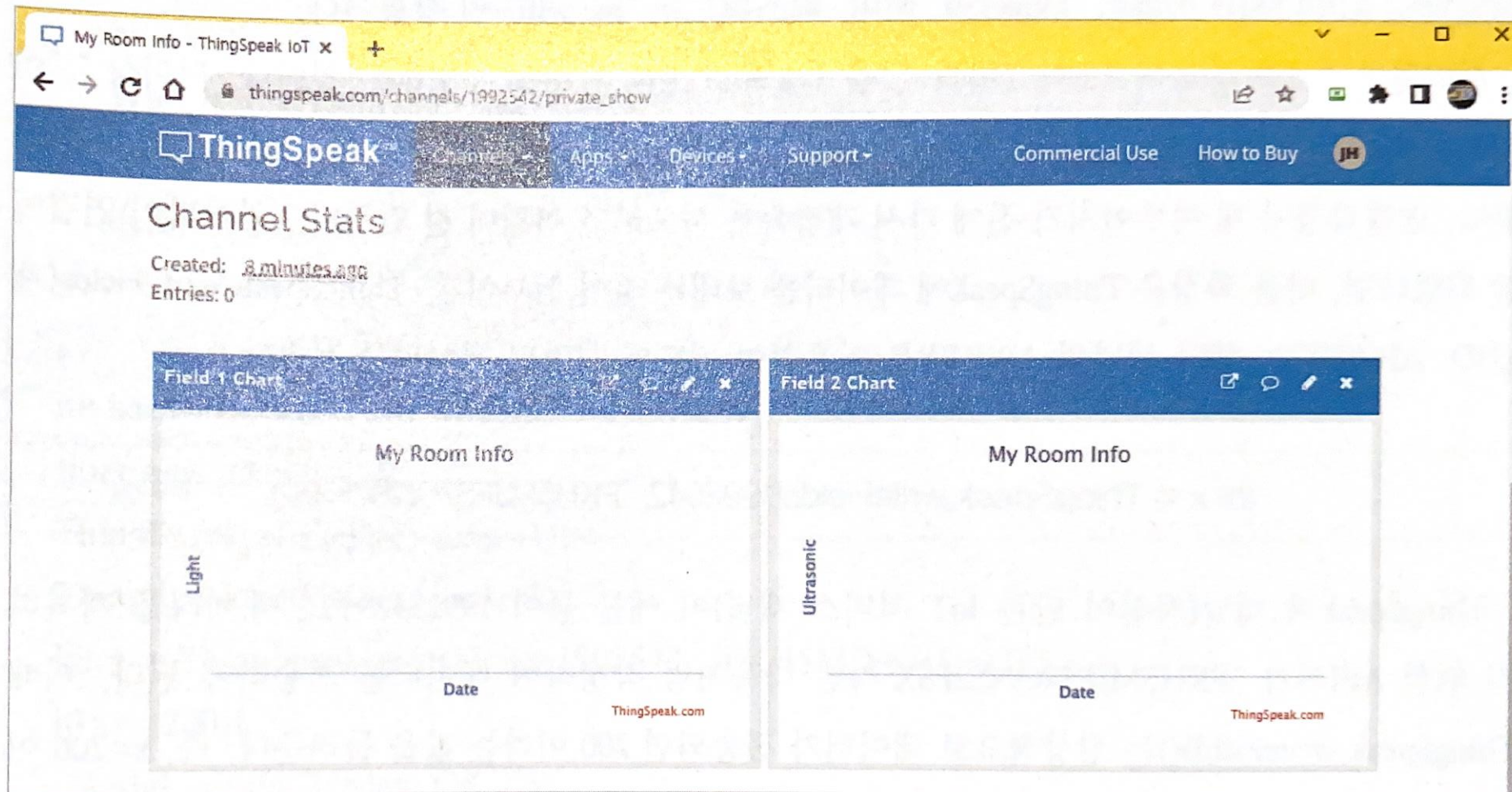
ThingSpeak 사용하기

‘Private View’ 메뉴 클릭




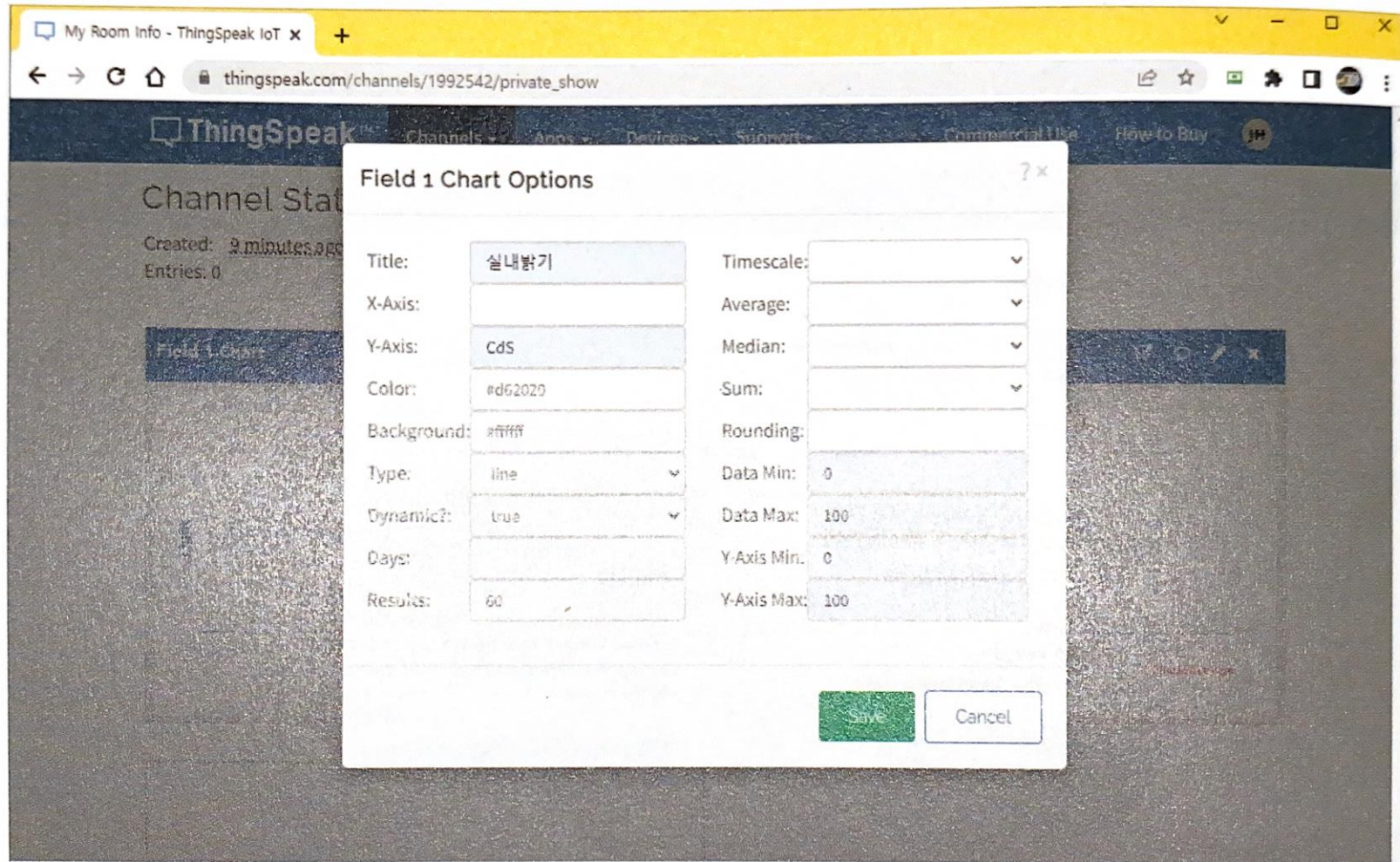
ThingSpeak 사용하기

입력한 Field별로 Chart가 구성되어 있음을 확인



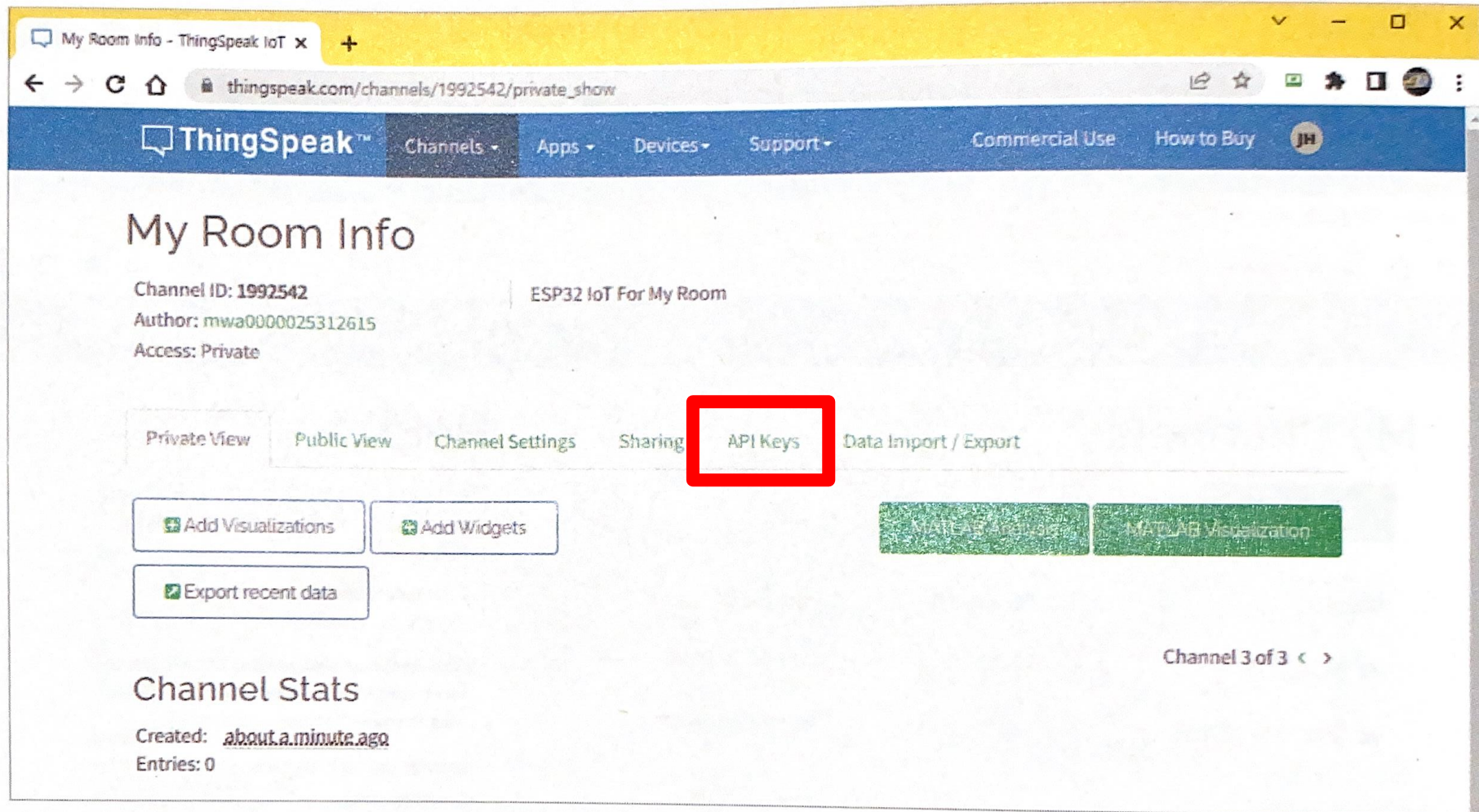
ThingSpeak 사용하기

연필 아이콘  을 클릭하여 Field의 정보를 입력



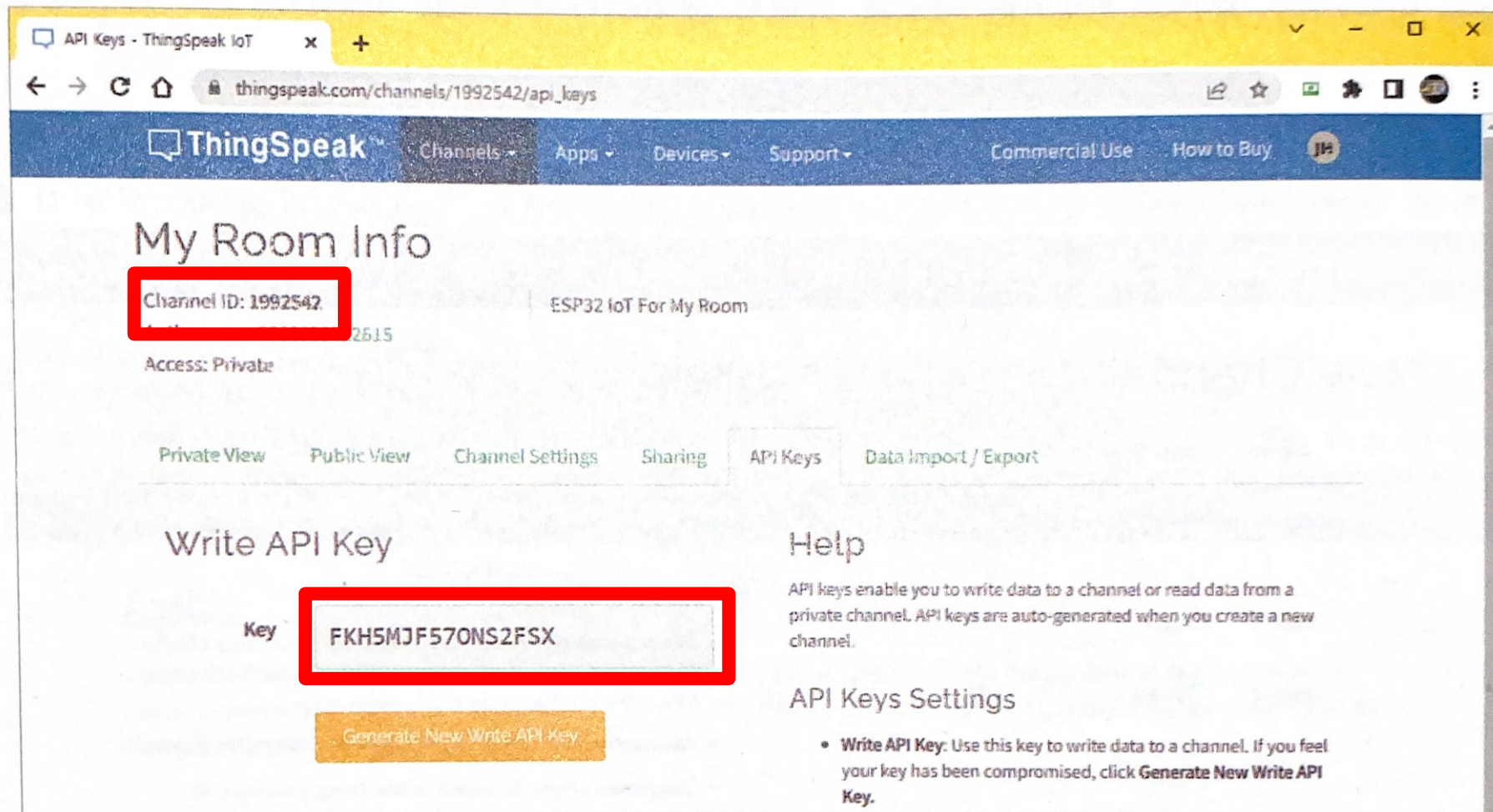
ThingSpeak 사용하기

‘API Keys’ 메뉴 클릭




ThingSpeak 이용하기

‘Channel ID’와 ‘API Key’값을 확인하고 코드 작성 시 사용 ★ ★ ★



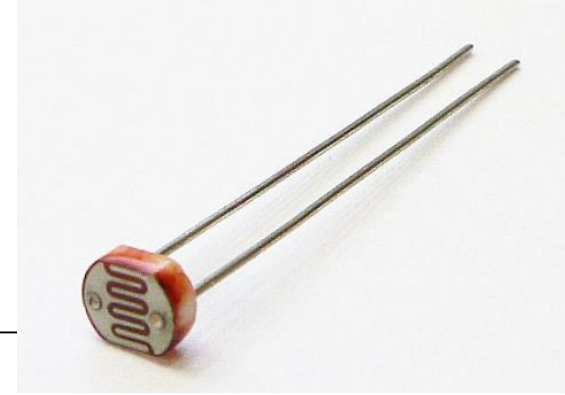
A large teal-colored circle with a thin black outline, containing white Korean text.

실내 밝기
측정하기
(조도 센서)

A large light gray-colored circle with a thin black outline, containing white Korean text.

컴퓨터
사용 정보
측정하기
(초음파 센서)

실내 밝기 측정하기 예제



```
int cds = 34;                                     // 34핀에 CdS센서(조도센서) 연결  
                                                    카드뮴 황  
void setup(){  
    Serial.begin(115200);  
    pinMode(cds, INPUT_PULLUP); // ESP32의 내부 풀업저항 활성화  
}  
...
```

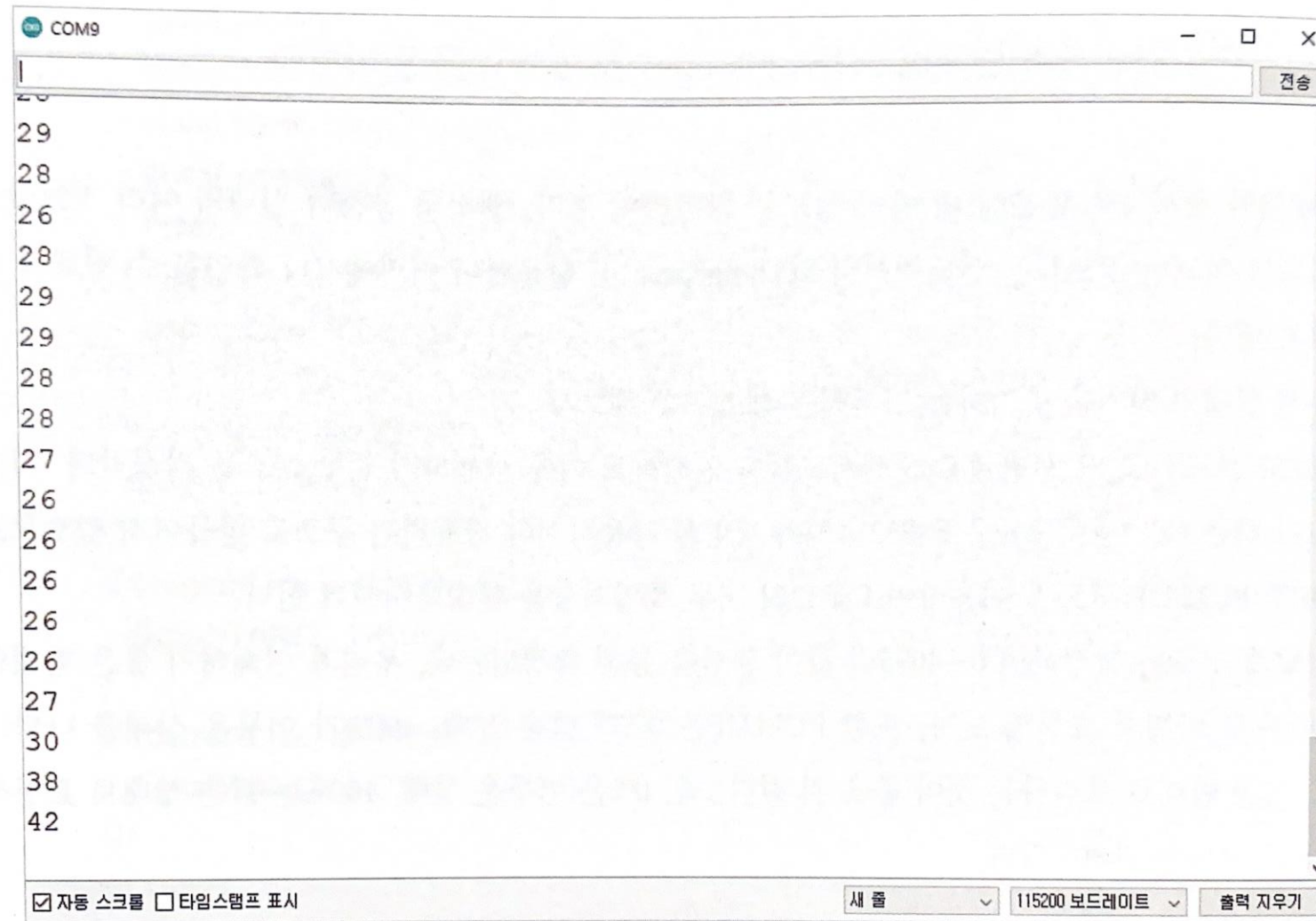
실내 밝기 측정하기 예제

```
...  
void loop(){  
    int sensor1 = light();  
    Serial.println(sensor1);  
    delay(100);  
}  
  
void light(){  
    int cds_val = analogRead(cds);  
    int value = map(cds_val, 0, 4095, 100, 0);  
    return value;  
}
```

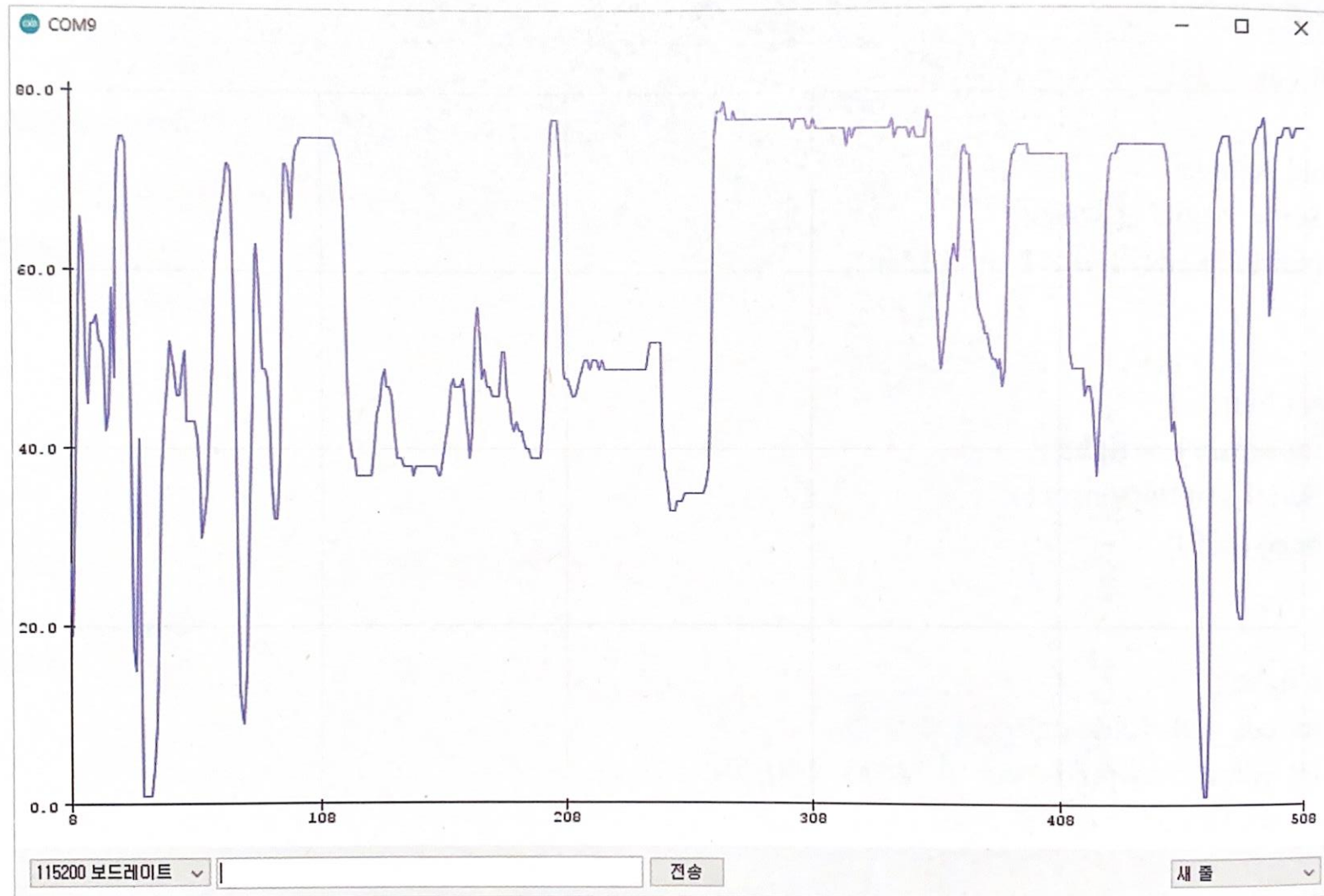
PULLUP은 0%가 밝은 상태,
100%가 어두운 상태를
나타내므로
이를 반대로 변경


// 0~4095 범위값
// 100~0로 변경

실내 밝기 측정하기 실행 결과(예)




실내 밝기 측정하기 실행 결과(예)





실내 밝기
측정하기
(조도 센서)



컴퓨터
사용 정보
측정하기
(초음파 센서)

컴퓨터 사용 정보 측정하기 예제

```
int trig = 13;           // 13핀에 trig핀 연결
int echo = 14;           // 14핀에 echo핀 연결

void setup(){
    Serial.begin(115200);
    pinMode(trig, OUTPUT);
    pinMode(echo, INPUT);
}

...
```

컴퓨터 사용 정보 측정하기 예제

...

```
void loop(){  
    float sensor2 = ultrasonic();  
    Serial.println(sensor2);  
    delay(100);  
}
```

...

컴퓨터 사용 정보 측정하기 예제

...

```
float ultrasonic(){  
    digitalWrite(trig, HIGH);  
    delayMicroseconds(5);  
    digitalWrite(trig, LOW);  
    float cm = pulseIn(echo, HIGH)/58.0;      // pulseIn 함수  
    float value = constrain(cm, 3.0, 100.0);  // constrain 함수  
    return value;  
}
```

컴퓨터 사용 정보 측정하기 예제

- pulseIn() 함수
: 펄스신호가 돌아오기까지의 시간을 계산하는 함수로 초음파 센서의 신호값이 돌아오는 것을 기다리는 것을 계산하고 HIGH 또는 LOW 신호가 몇 마이크로초 동안 들어왔는지를 계산

$$1\text{초}(s) = 10^6\text{마이크로초}(\mu s)$$

컴퓨터 사용 정보 측정하기 예제

```
float cm = pulseIn(echo, HIGH)/58.0;
```

- 초음파 속도: 340m/s
- 거리(cm) = 시간 × 속도

= pulseIn() (μs) × 170(m/s) // 편도로 계산

= pulseIn() (μs) × 170 × 10²(cm) / 10⁶(μs)

= pulseIn() × 17 × 10⁻³

≈ pulseIn() / 58.0

컴퓨터 사용 정보 측정하기 예제

- constrain() 함수

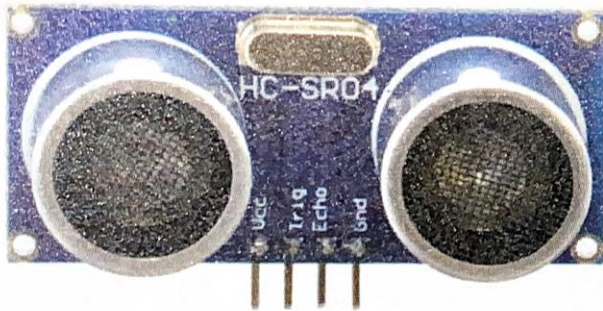
: 최솟값과 최댓값을 지정하여, 최솟값보다 작은 값과 최댓값보다 큰 값을 버림
(신뢰하기 어려운 데이터나 불필요한 데이터를 버리기 위해 주로 사용)

```
float value = constrain(cm, 3.0, 100.0);
```

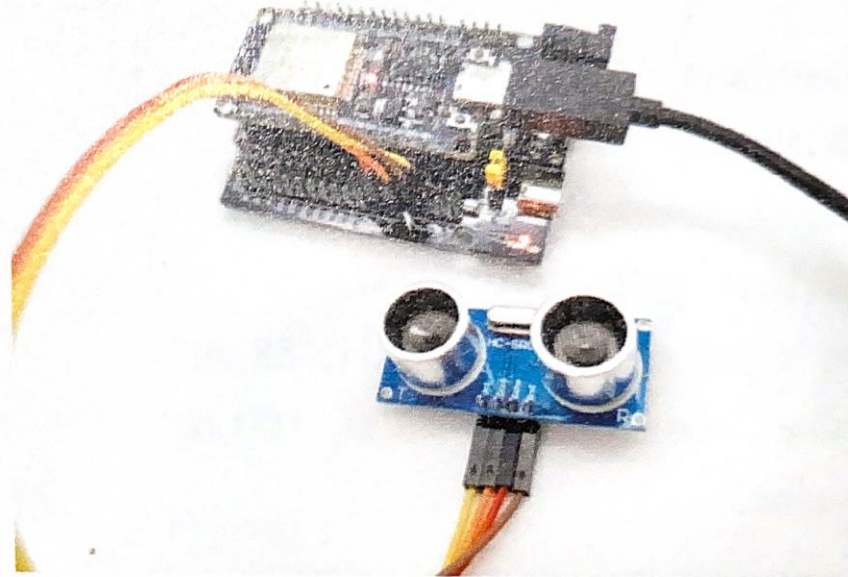
- ☞ 3cm ~ 100cm 범위에 장애물이 인식되는지 판단하기 위한 코드
(센서 자체의 성능상 3cm 미만의 값은 신뢰하기 어렵고, 컴퓨터 앞에 사람이 있는지 여부를 판단하기에 100cm 이상의 거리를 측정하는 것은 불필요함)

컴퓨터 사용 정보 측정하기 예제

VCC를 3.3V에 연결해서 작동이 불안정한 경우 5V에 연결(5V 사용을 권장)

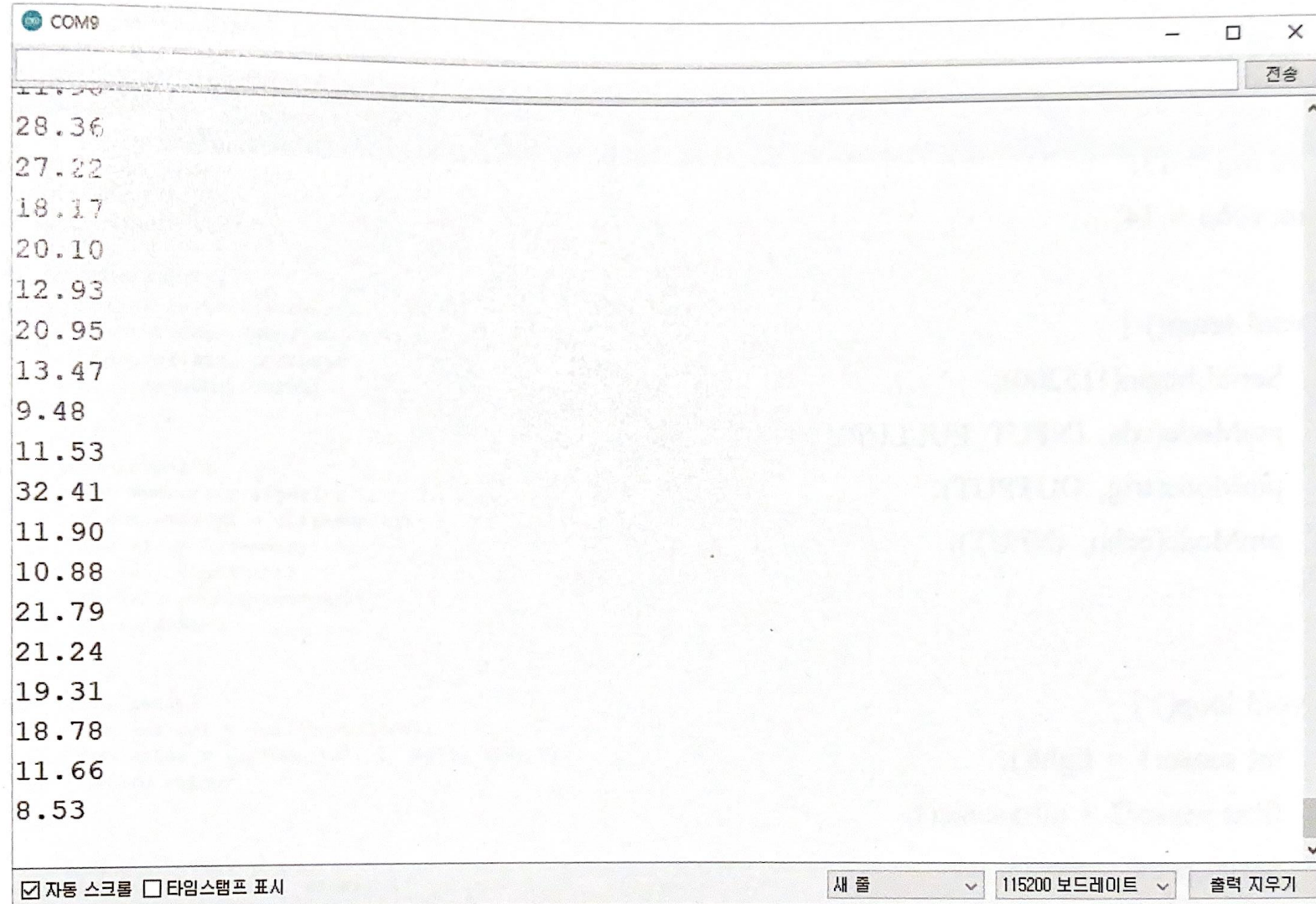


초음파센서(HC-SR04)

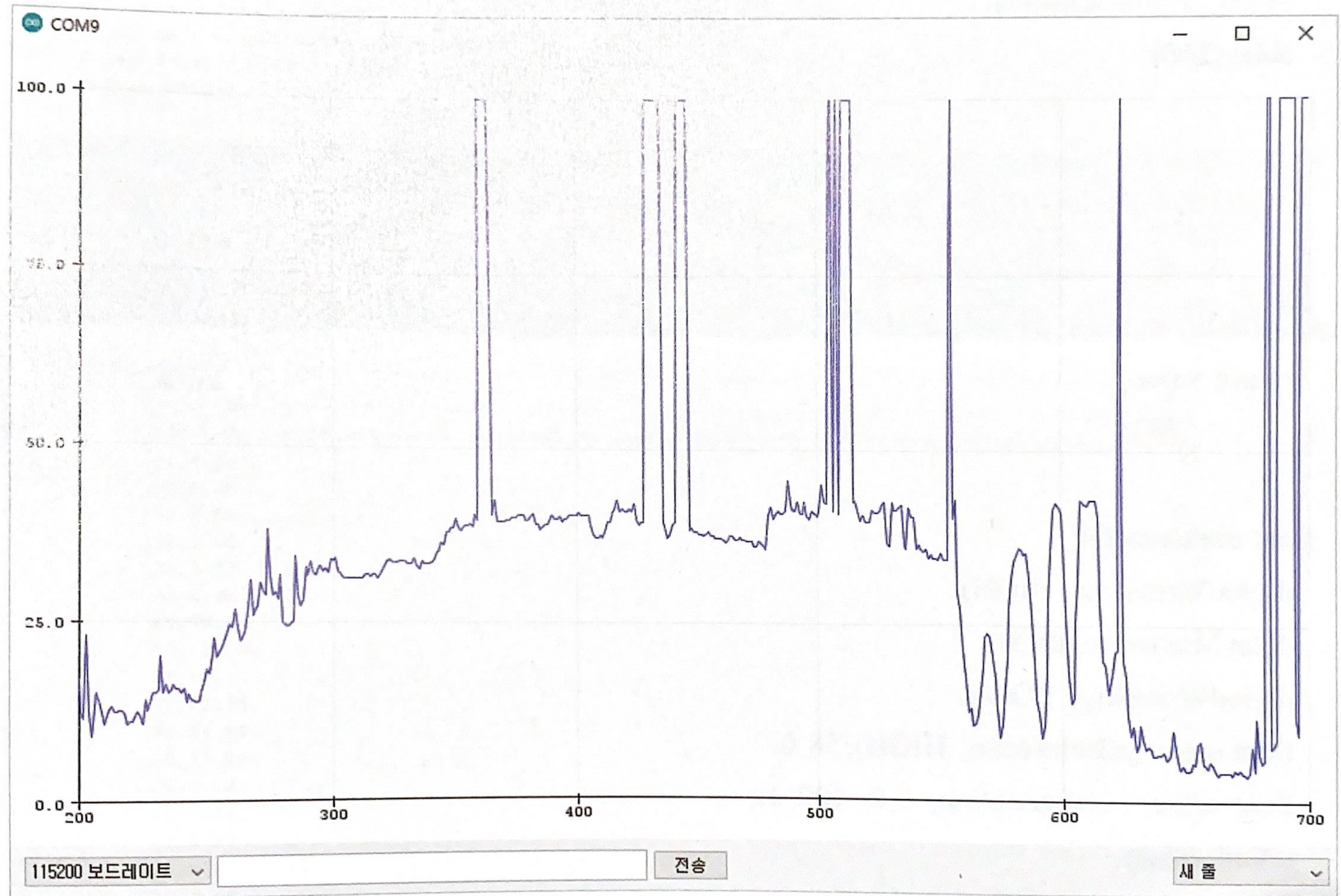


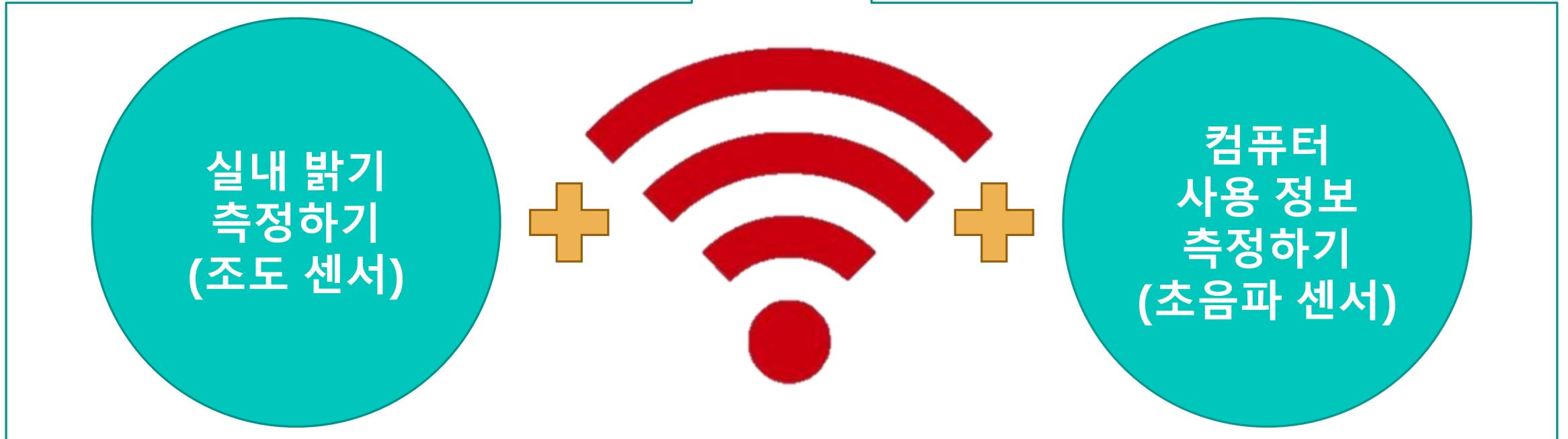
Vcc-5V / Trig-13핀 / Echo-14핀 / Gnd-GND

컴퓨터 사용 정보 측정하기 실행 결과(예)

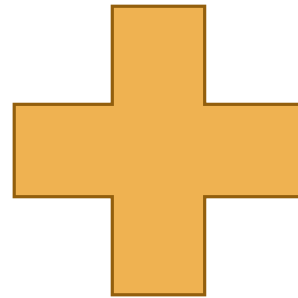


컴퓨터 사용 정보 측정하기 실행 결과(예)





실내 밝기
측정하기
(조도 센서)



컴퓨터
사용 정보
측정하기
(초음파 센서)

실내 밝기 + 컴퓨터 사용 정보 측정하기 예제

```
int cds = 34;
int trig = 13;
int echo = 14;

void setup(){
    Serial.begin(115200);
    pinMode(cds, INPUT_PULLUP);
    pinMode(trig, OUTPUT);
    pinMode(echo, INPUT);
}

...
```

실내 밝기 + 컴퓨터 사용 정보 측정하기 예제

...

```
void loop(){  
    int sensor1 = light();  
    float sensor2 = ultrasonic();  
    Serial.println(sensor1);  
    Serial.println(sensor2);  
    delay(200);  
}
```

...

실내 밝기 + 컴퓨터 사용 정보 측정하기 예제

...

```
void light(){  
    int cds_val = analogRead(cds);  
    int value = map(cds_val, 0, 4095, 100, 0);  
    return value;  
}
```

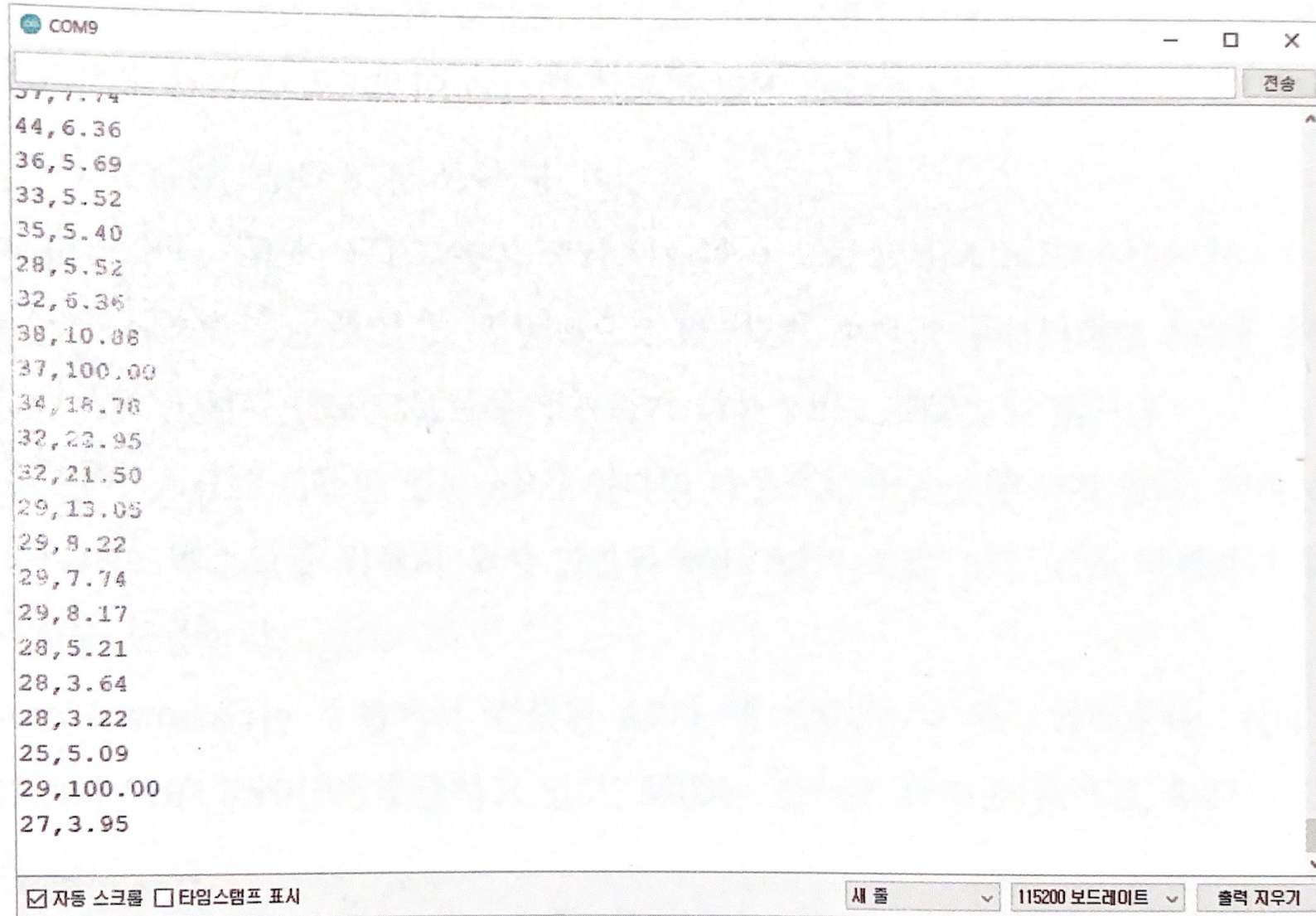
...

실내 밝기 + 컴퓨터 사용 정보 측정하기 예제

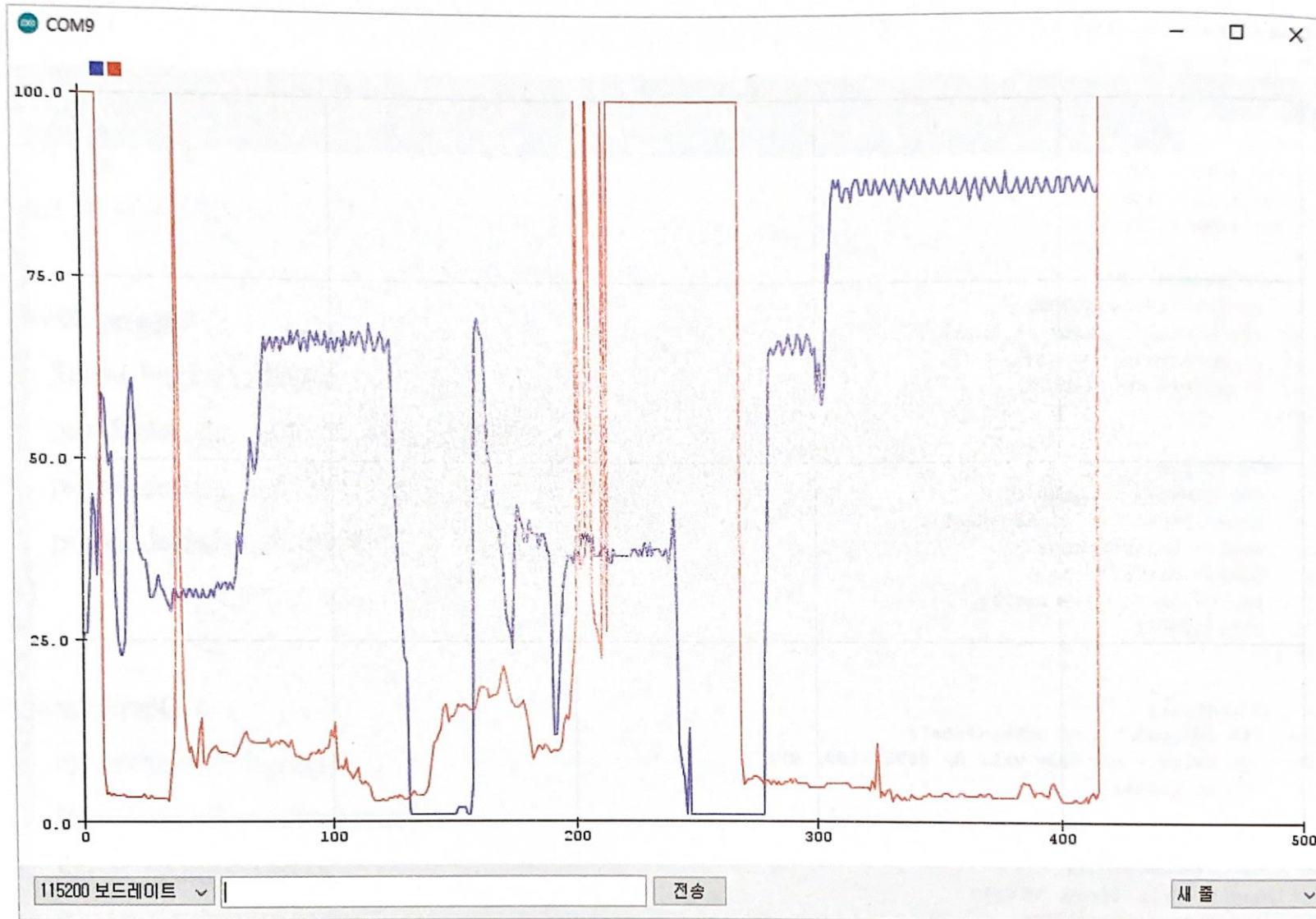
...

```
float ultrasonic(){  
    digitalWrite(trig, HIGH);  
    delayMicroseconds(5);  
    digitalWrite(trig, LOW);  
    float cm = pulseIn(echo, HIGH)/58.0;  
    float value = constrain(cm, 3.0, 100.0);  
    return value;  
}
```

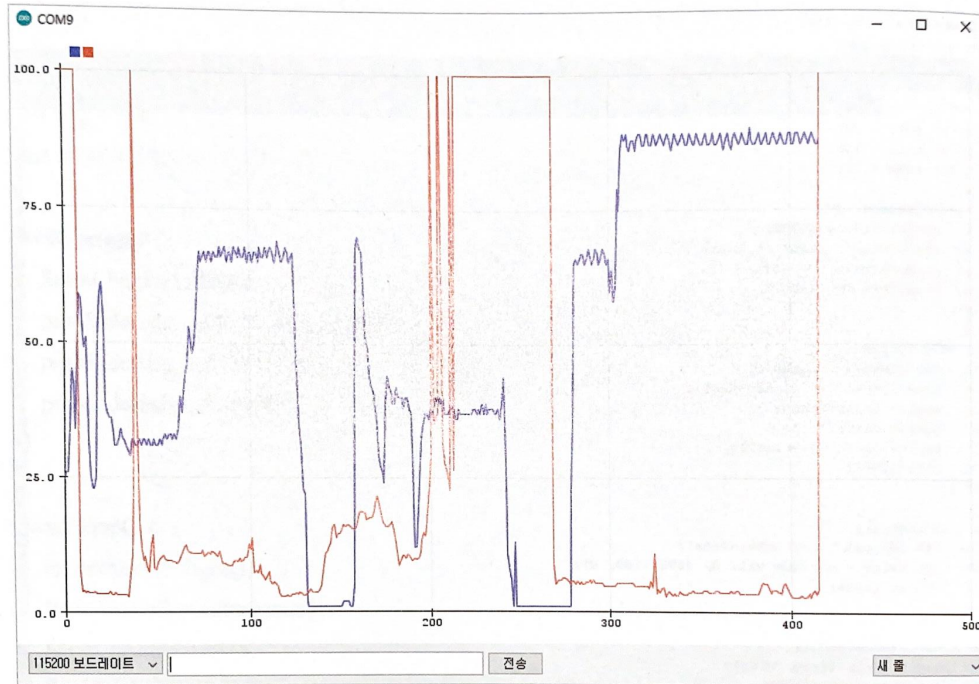
실내 밝기 + 컴퓨터 사용 정보 측정하기 실행 결과(예)



실내 밝기 + 컴퓨터 사용 정보 측정하기 실행 결과(예)



실내 밝기 + 컴퓨터 사용 정보 측정하기 실행 결과(예)



- 붉은색: 3cm~100cm의 거리
- 파란색: 0%~100%로 나타낸 빛의 세기

👉 2가지 센서를 통해 컴퓨터 앞에 장애물이 놓인 상황을 보다 정확하게 판단가능
(예) 60cm 미만에서 장애물 측정 시 빛의 밝기가 함께 어두워짐을 확인

실내 밝기 + 컴퓨터 사용 정보 측정하기 실행 결과(예)

Q. 만약 초음파는 장애물을 인식했는데 어둡기는 변화가 없거나,
밝기는 어두워졌는데 초음파가 장애물을 인식하지 못했다면?

A. 아침이 되어 주변이 밝아졌거나, 전등을 켜 상황일 수 있음

☞ 따라서 여러 개의 센서를 복합적으로 사용할수록 정확도가 높아짐



ESP32가 접속할 수 있는 Wi-Fi AP 검색하기 예제

```
#include "WiFi.h"                // 라이브러리 파일 'WiFi.h'를 포함

void setup(){
    Serial.begin(115200);
    WiFi.mode(WIFI_STA);         // 내장 Wi-Fi 모듈을 STA 모드로 사용
    WiFi.disconnect();           // 기존 연결된 Wi-Fi 해제를 위함
    delay(100);
}

...
```

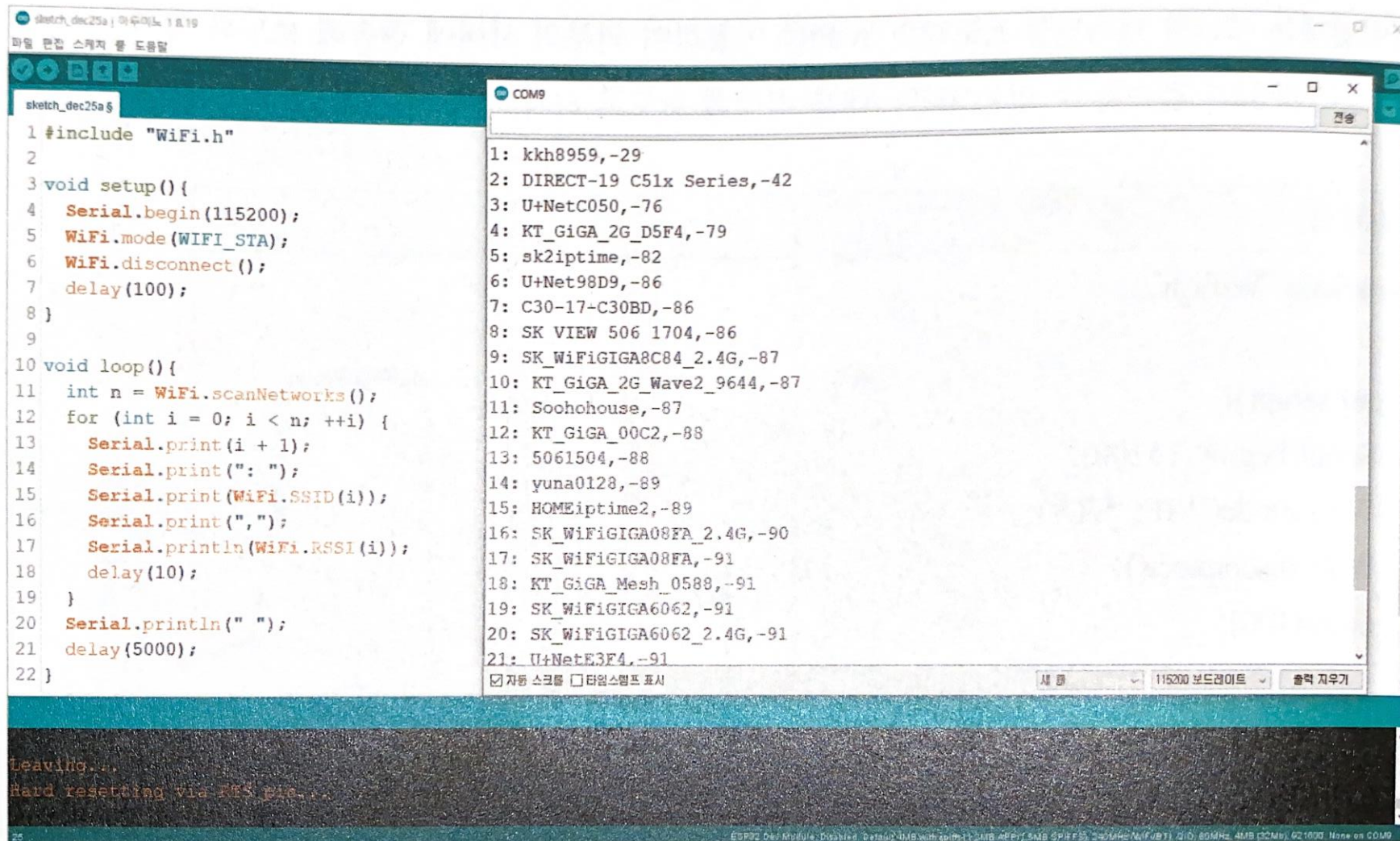
ESP32가 접속할 수 있는 Wi-Fi AP 검색하기 예제

```
...
void loop(){
    int n = WiFi.scanNetworks();           // 검색된 AP의 개수
    for(int i=0 ; i<n ; ++i){
        Serial.print(i+1);
        Serial.print(": ");
        Serial.println(WiFi.SSID(i));       // 접속할 AP의 이름
        Serial.print(",");
        Serial.println(WiFi.RSSI(i));       // 각 AP의 신호 세기
        delay(10);
    }
    ...
}
```

ESP32가 접속할 수 있는 Wi-Fi AP 검색하기 예제

```
...  
Serial.println(" ");  
delay(5000);  
}
```


ESP32가 접속할 수 있는 Wi-Fi AP 검색하기 실행 결과(예)



```
1 #include "WiFi.h"
2
3 void setup() {
4   Serial.begin(115200);
5   WiFi.mode(WIFI_STA);
6   WiFi.disconnect();
7   delay(100);
8 }
9
10 void loop() {
11   int n = WiFi.scanNetworks();
12   for (int i = 0; i < n; ++i) {
13     Serial.print(i + 1);
14     Serial.print(": ");
15     Serial.print(WiFi.SSID(i));
16     Serial.print(", ");
17     Serial.println(WiFi.RSSI(i));
18     delay(10);
19   }
20   Serial.println(" ");
21   delay(5000);
22 }
```

```
1: kkh8959,-29
2: DIRECT-19 C51x Series,-42
3: U+NetC050,-76
4: KT_GiGA_2G_D5F4,-79
5: sk2iptime,-82
6: U+Net98D9,-86
7: C30-17-C30BD,-86
8: SK VIEW 506 1704,-86
9: SK_WiFiGIGA8C84_2.4G,-87
10: KT_GiGA_2G_Wave2_9644,-87
11: Sochohouse,-87
12: KT_GiGA_00C2,-88
13: 5061504,-88
14: yuna0128,-89
15: HOMEiptime2,-89
16: SK_WiFiGIGA08FA_2.4G,-90
17: SK_WiFiGIGA08FA,-91
18: KT_GiGA_Mesh_0588,-91
19: SK_WiFiGIGA6062,-91
20: SK_WiFiGIGA6062_2.4G,-91
21: U+NetE3F4,-91
```

Leaving...
Hard resetting via RTS pin...

ESP32가 접속할 수 있는 Wi-Fi AP 검색하기 실행 결과(예)

Q. 와이파이가 2.4GHz와 5.0GHz 두 개 존재합니다. 어떤 주파수 대역의 와이파이를 사용해야 하나요?

A. ESP32에 내장된 Wi-Fi 모듈은 5.0GHz는 지원하지 않습니다. 따라서 2.4GHz를 지원하는 AP를 선택해야 합니다.

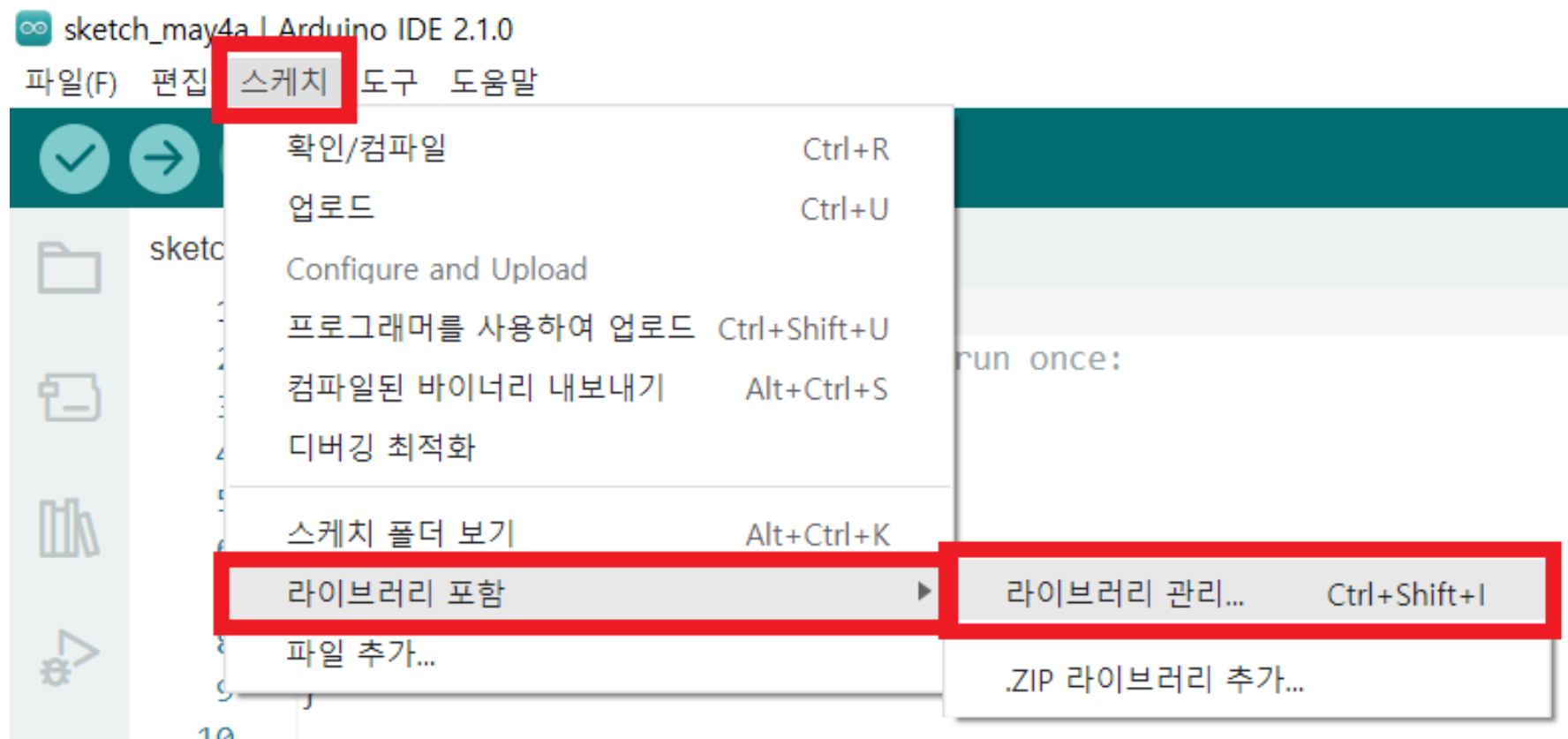
ESP32
테스트 코드



ThingSpeak
업로드

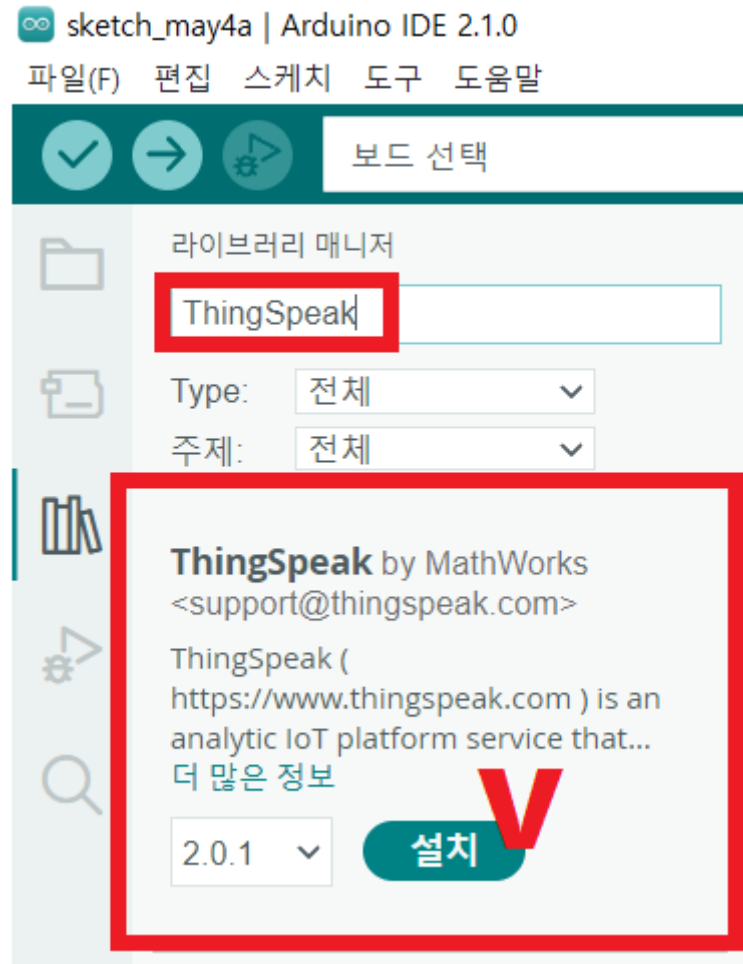
ThingSpeak용 라이브러리 설치하기

Arduino IDE > 스케치 > 라이브러리 포함 > 라이브러리 관리



ThingSpeak용 라이브러리 설치하기

라이브러리 매니저에서 'ThingSpeak' 검색 후 설치



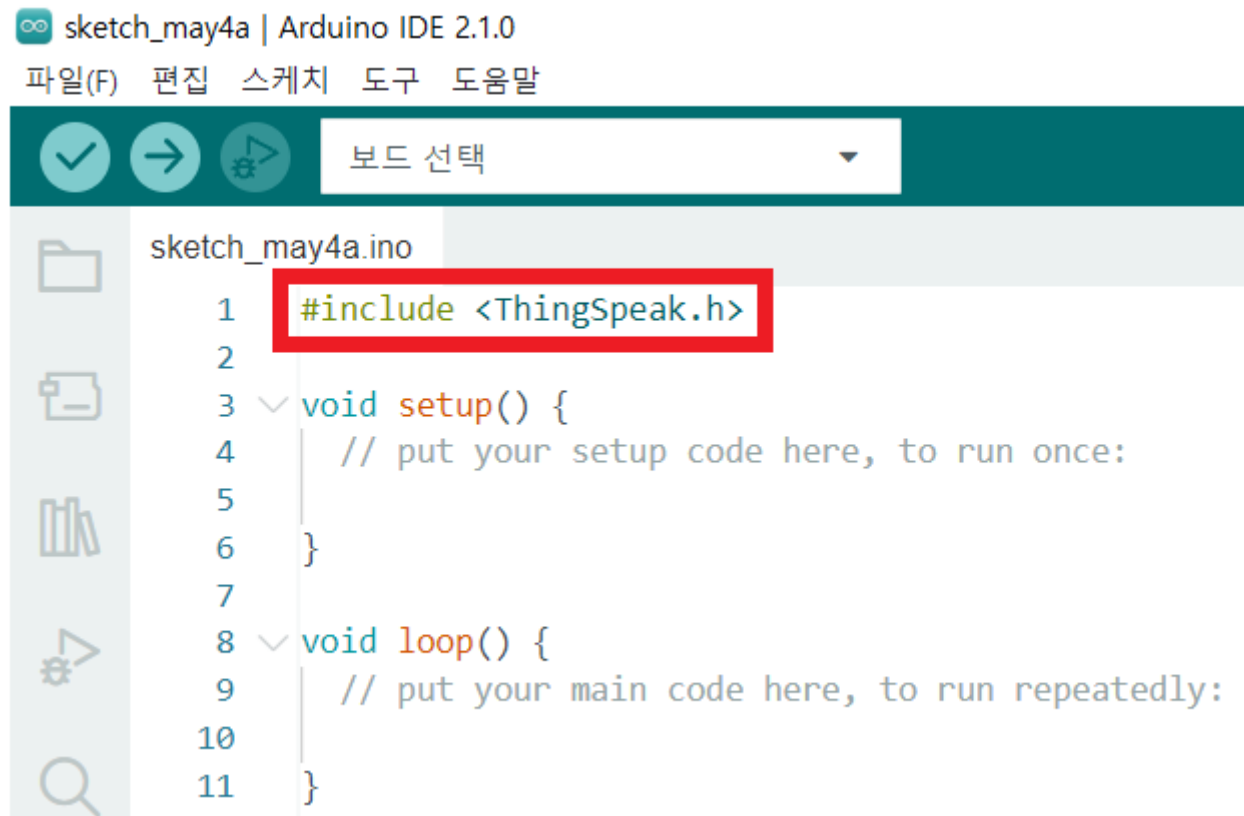
ThingSpeak용 라이브러리 설치하기

스케치 > 라이브러리 포함 > 공헌된 라이브러리들에서 'ThingSpeak' 선택



ThingSpeak용 라이브러리 설치하기

라이브러리가 정상적으로 설치되어 코드에 반영되었는지 확인



```
sketch_may4a | Arduino IDE 2.1.0
파일(F) 편집 스케치 도구 도움말

✓ → ⚙ 보드 선택

sketch_may4a.ino
1 #include <ThingSpeak.h>
2
3 void setup() {
4     // put your setup code here, to run once:
5
6 }
7
8 void loop() {
9     // put your main code here, to run repeatedly:
10
11 }
```


ThingSpeak 연결 테스트 코드 예제

```
#include <WiFi.h>
#include "ThingSpeak.h"
WiFiClient esp32;

void setup(){
    Serial.begin(115200);
    WiFi.mode(WIFI_STA);
    ThingSpeak.begin(esp32);
}

...
```

ThingSpeak 연결 테스트 코드 예제

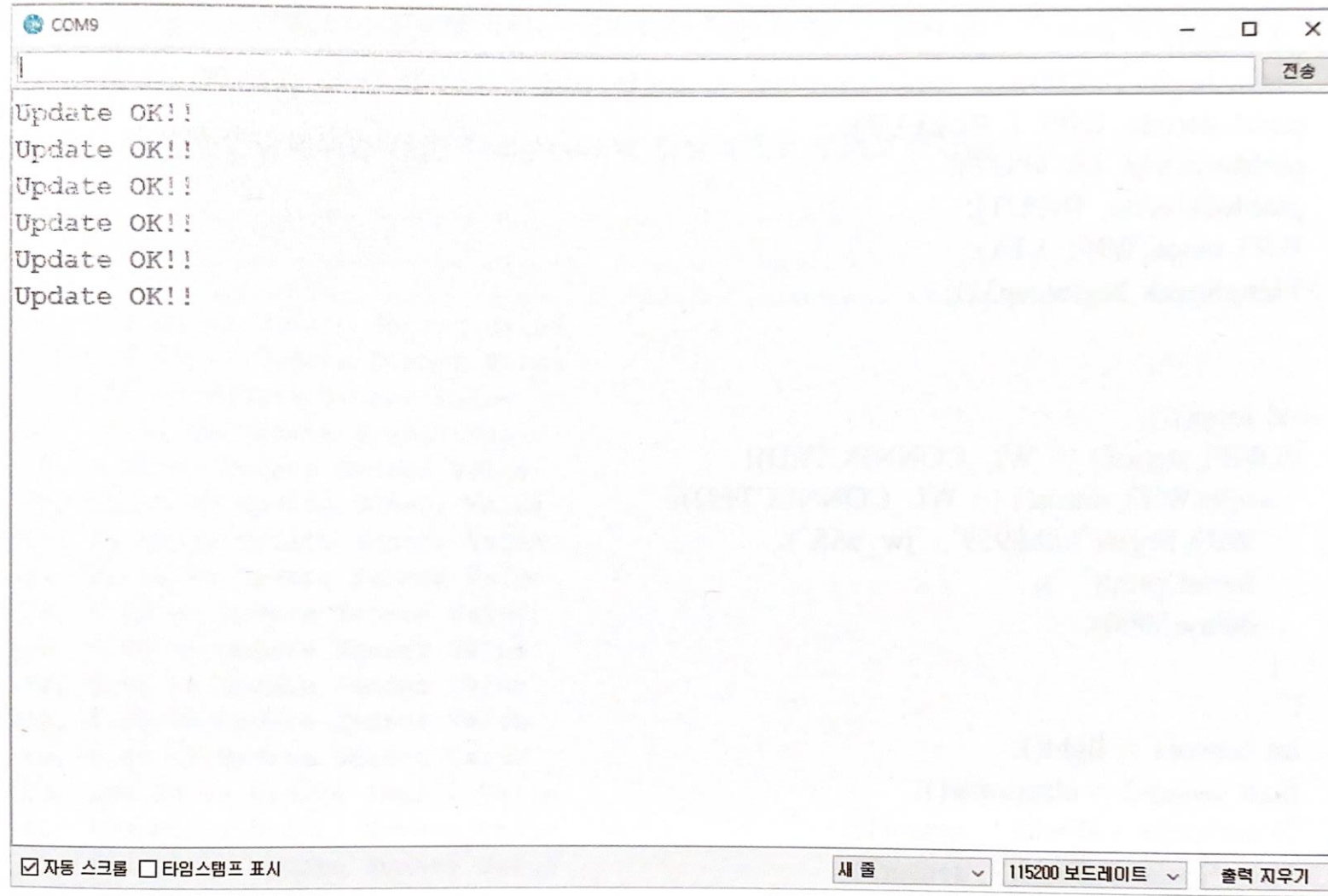
```
...  
  
void loop(){  
    if(WiFi.status() != WL_CONNECTED){           // AP 연결  
        while(WiFi.status() != WL_CONNECTED){  
            WiFi.begin("iptime", "qwerty");      // AP의 이름과 암호  
            Serial.print(".");  
            delay(5000);                          // 5초 당 점(.)을 찍으며 접속 시도  
        }                                         // 점(.)이 찍히지 않으면 접속 성공  
    }  
  
    ...
```

ThingSpeak 연결 테스트 코드 예제

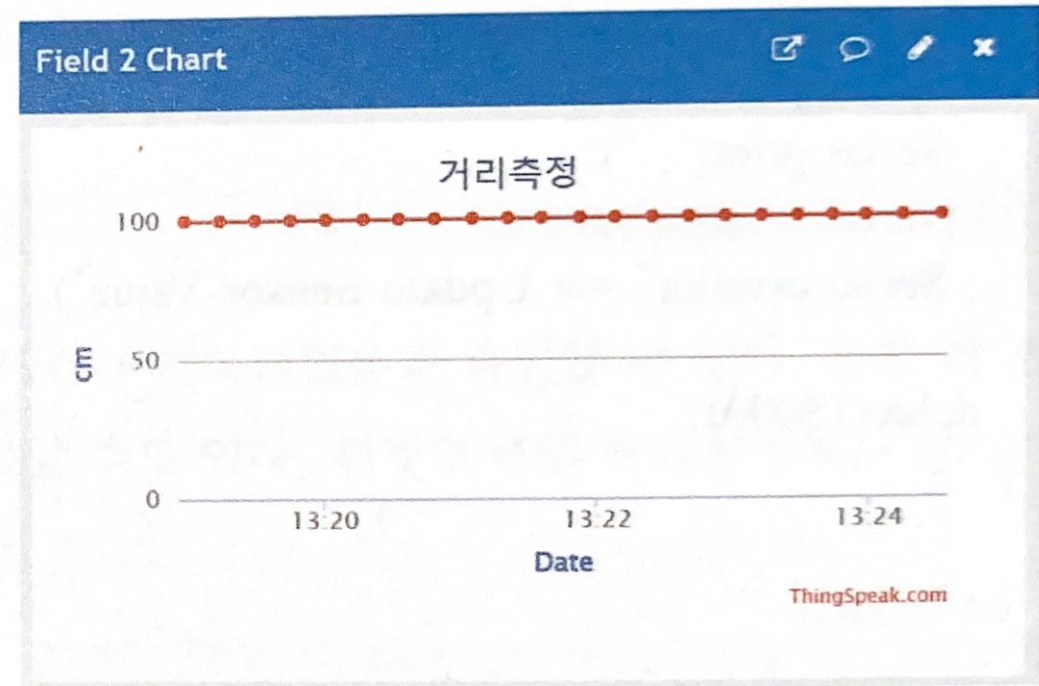
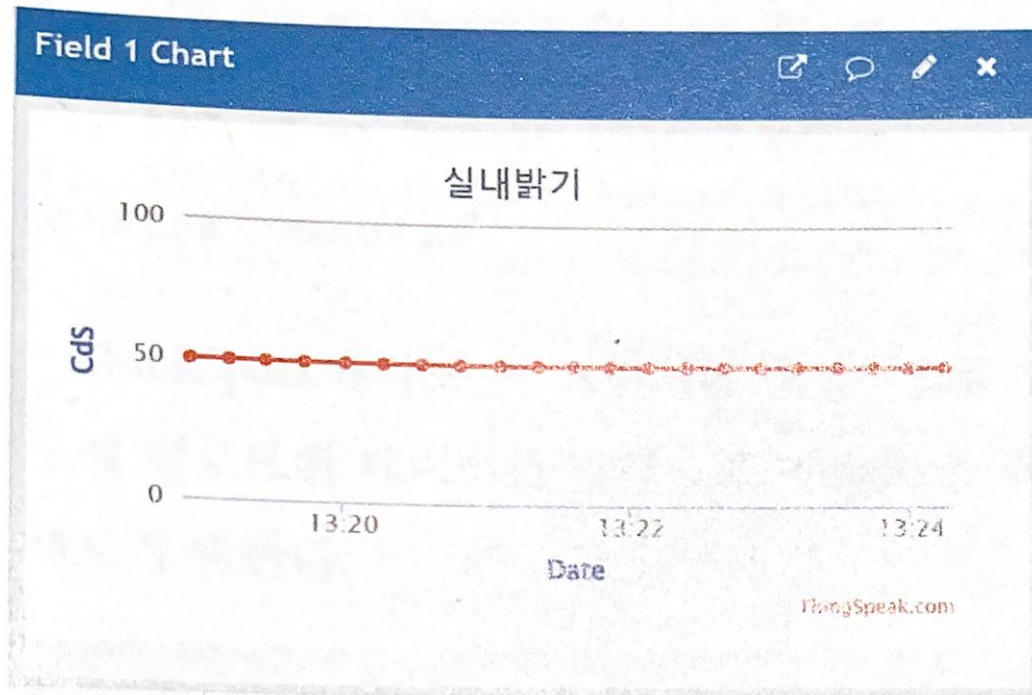
```
...                                     // AP 연결 후 ThingSpeak로 데이터 전송

int sensor1 = 50;
float sensor2 = 100.0;
ThingSpeak.setField(1, sensor1);
ThingSpeak.setField(2, sensor2);
int x = ThingSpeak.writeFields(1992542, "FKH5MJF570NS2FSX");
if(x==200){                           Channel ID                API KEY
    Serial.println("Update OK!");    // 정상적인 데이터전송
}
delay(15000);                        // ThingSpeak는 15초 간격으로 데이터를 업데이트
}
```

ThingSpeak 연결 테스트 코드 실행 결과(예)



ThingSpeak 연결 테스트 코드 실행 결과(예)



실제 데이터
측정



ThingSpeak
업로드

실제 데이터 전송 코드 예제

```
#include <WiFi.h>
#include "ThingSpeak.h"
WiFiClient esp32;
```

```
int cds = 34;
int trig = 13;
int echo = 14;
```

```
...
```


실제 데이터 전송 코드 예제

...

```
void setup(){  
  Serial.begin(115200);  
  pinMode(cds, INPUT_PULLUP);  
  pinMode(trig, OUTPUT);  
  pinMode(echo, INPUT);  
  WiFi.mode(WIFI_STA);  
  ThingSpeak.begin(esp32);  
}
```

...

실제 데이터 전송 코드 예제

```
...  
  
void loop(){  
    if(WiFi.status() != WL_CONNECTED){  
        while(WiFi.status() != WL_CONNECTED){  
            WiFi.begin("iptime", "qwerty");  
            Serial.print(".");  
            delay(5000);  
        }  
    }  
  
    ...  
}
```

실제 데이터 전송 코드 예제

```
...
int sensor1 = light();
float sensor2 = ultrasonic();
ThingSpeak.setField(1, sensor1);
ThingSpeak.setField(2, sensor2);
int x = ThingSpeak.writeFields(1992542, "FKH5MJF570NS2FSX");
if(x==200){
    Serial.print(sensor1);
    Serial.print(", ");
    Serial.print(sensor2);
    Serial.println(" => Update Sensor Value");
}
delay(15000);
}
```

...

실제 데이터 전송 코드 예제

...

```
int light(){  
    int cds_val = analogRead(cds);  
    int value = map(cds_val, 0, 4095, 100, 0);  
    return value;  
}
```

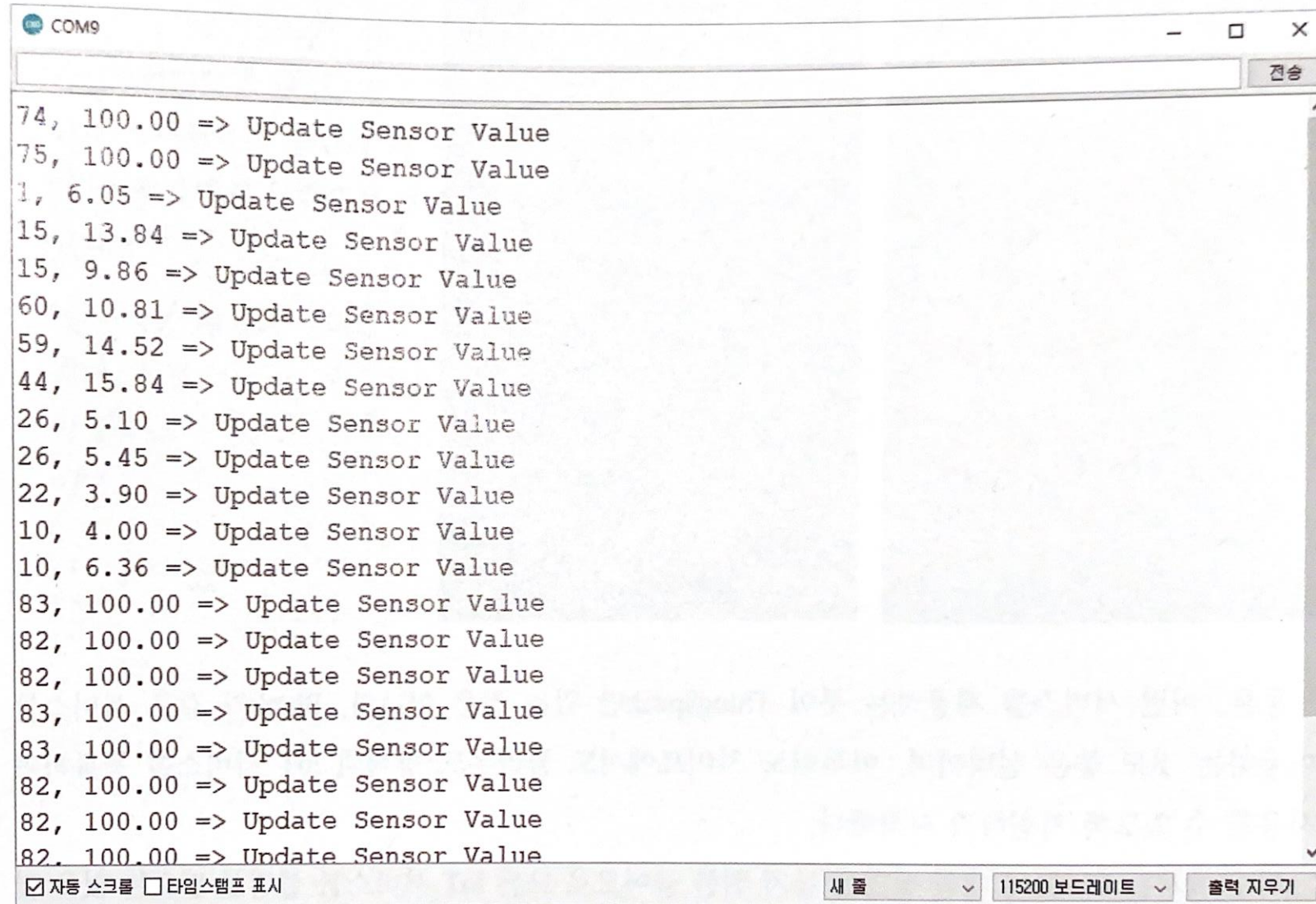
...

실제 데이터 전송 코드 예제

...

```
float ultrasonic(){  
    digitalWrite(trig, HIGH);  
    delayMicroseconds(5);  
    digitalWrite(trig, LOW);  
    float cm = pulseIn(echo, HIGH)/58.0;  
    float value = constrain(cm, 3.0, 100.0);  
    return value;  
}
```

실제 데이터 전송 코드 실행 결과(예)

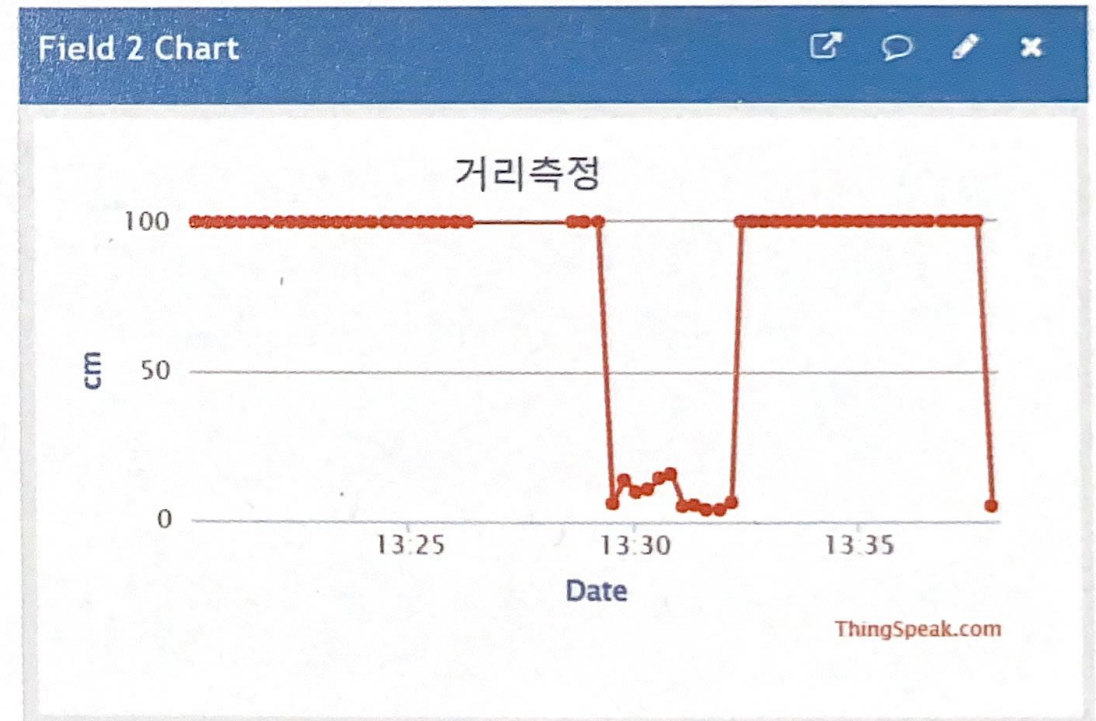
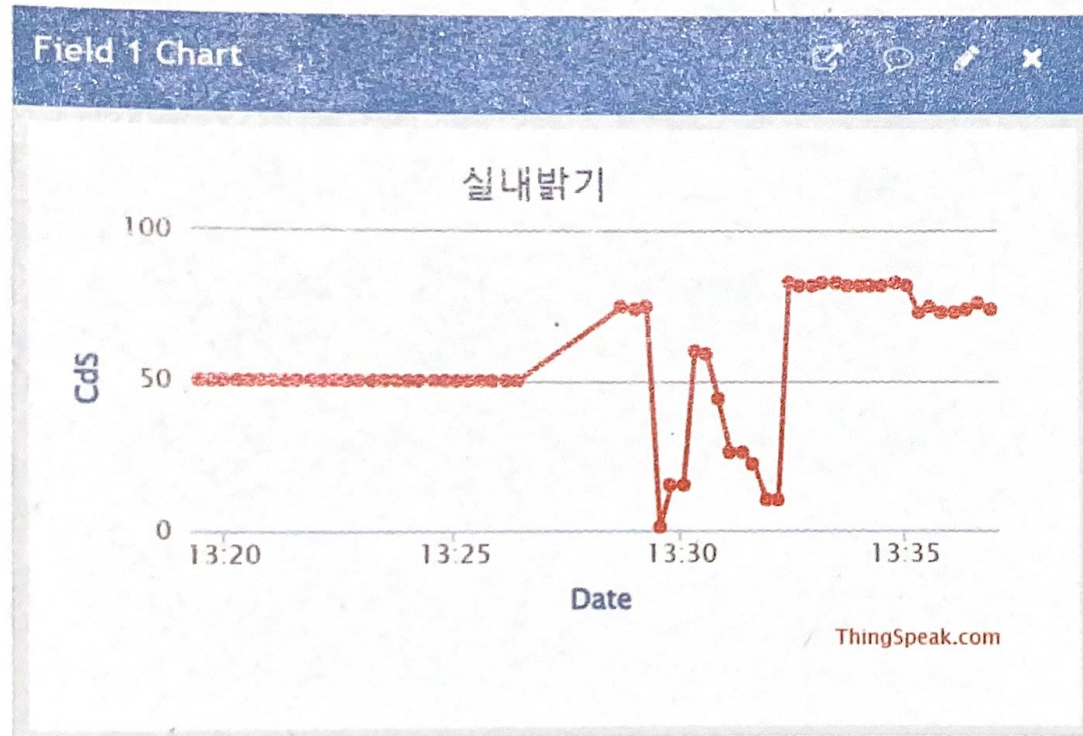


A screenshot of a Windows terminal window titled 'COM9'. The window displays a list of sensor data updates, each consisting of an ID, a value, and the text '=> Update Sensor Value'. The data is as follows:

ID	Value	Action
74	100.00	=> Update Sensor Value
75	100.00	=> Update Sensor Value
1	6.05	=> Update Sensor Value
15	13.84	=> Update Sensor Value
15	9.86	=> Update Sensor Value
60	10.81	=> Update Sensor Value
59	14.52	=> Update Sensor Value
44	15.84	=> Update Sensor Value
26	5.10	=> Update Sensor Value
26	5.45	=> Update Sensor Value
22	3.90	=> Update Sensor Value
10	4.00	=> Update Sensor Value
10	6.36	=> Update Sensor Value
83	100.00	=> Update Sensor Value
82	100.00	=> Update Sensor Value
82	100.00	=> Update Sensor Value
83	100.00	=> Update Sensor Value
83	100.00	=> Update Sensor Value
82	100.00	=> Update Sensor Value
82	100.00	=> Update Sensor Value
82	100.00	=> Update Sensor Value

At the bottom of the window, there are several controls: a checkbox for '자동 스크롤' (checked), a checkbox for '타임스탬프 표시' (unchecked), a dropdown menu for '세 줄' (set to '세 줄'), a dropdown menu for '115200 보드레이트' (set to '115200 보드레이트'), and a button labeled '출력 지우기'.

실제 데이터 전송 코드 실행 결과(예)



프로젝트 안내 및 진행

프로젝트 안내 및 진행

**지금까지 배운 내용을 바탕으로
ESP32를 활용하여 자신만의 프로그램을
만들어봅시다.**

프로젝트 안내 및 진행

주제
선정



역할
분배



프로
그램
작성



오류
수정



결과
발표

프로젝트 안내 및 진행

- **팀명:** _____
- **팀원:** _____

프로젝트 안내 및 진행

- 주제: _____
- 역할분배: _____

프로젝트 안내 및 진행

- **제작 계획:**

프로젝트 결과 발표

프로젝트 결과 발표 보고서

◆ ESP32 프로젝트 결과 보고서 ◆	
팀명	
팀원	
제작시기 (날짜, 시간)	

작품 제목	
기능	
유용성	

제작 시
어려웠던 점

느낀 점 &
앞으로의 다짐

프로젝트 결과 발표

**프로그램을 만든 과정과 결과를
친구들 앞에서 발표해봅시다.**

수고하셨습니다.

참고문헌 및 자료 출처

- 하썸보드로 ESP32 고수되기(하주원, 현우사)
- <https://lastminuteengineers.com/esp32-i2c-lcd-tutorial/>
- <https://korean.alibaba.com/product-detail/KY-022-infrared-sensor-receiving-module-1600661644500.html>
- http://www.ktword.co.kr/test/view/view.php?m_temp1=2709
- <https://www.funshop.co.kr/goods/detail/60341>
- <http://m.vctec.co.kr/product/esp32-c3s-24ghz-wifi%EB%B8%94%EB%A3%A8%ED%88%AC%EC%8A%A4-ble-risc-v-%EB%B3%B4%EB%93%9C-4mb-%ED%94%8C%EB%9E%98%EC%89%AC-esp32-c3s-24ghz-wifi-bo/19169/>
- <https://www.youtube.com/watch?v=X-ozIRMRYZ4>
- https://kr.freepik.com/free-photo/front-view-of-hand-holding-smartphone_11372498.htm#query=%EC%8A%A4%EB%A7%88%ED%8A%B8%ED%8F%B0&position=1&from_view=keyword&track=sph

참고문헌 및 자료 출처

- <https://www.itworld.co.kr/howto/148445>
- <https://www.earlyadopter.co.kr/155097>
- <https://thingspeak.com/>
- <https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&blogId=jf102&logNo=220993294363>