

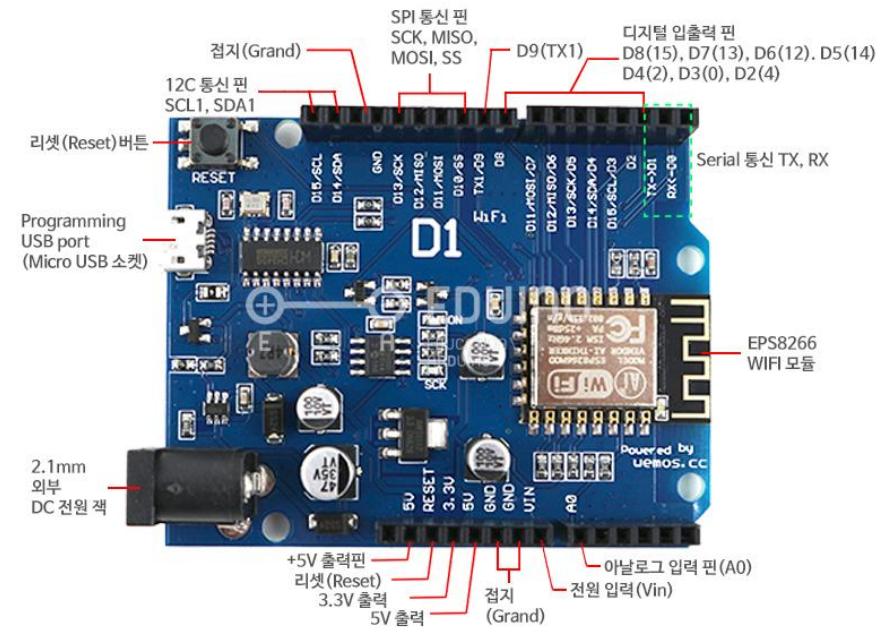
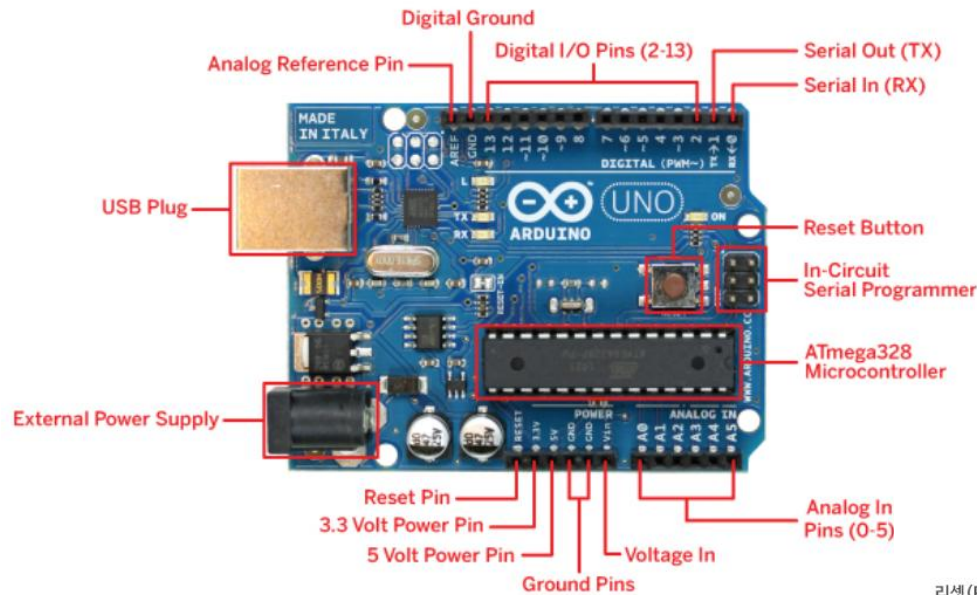
# 아두이노(Arduino) 소개



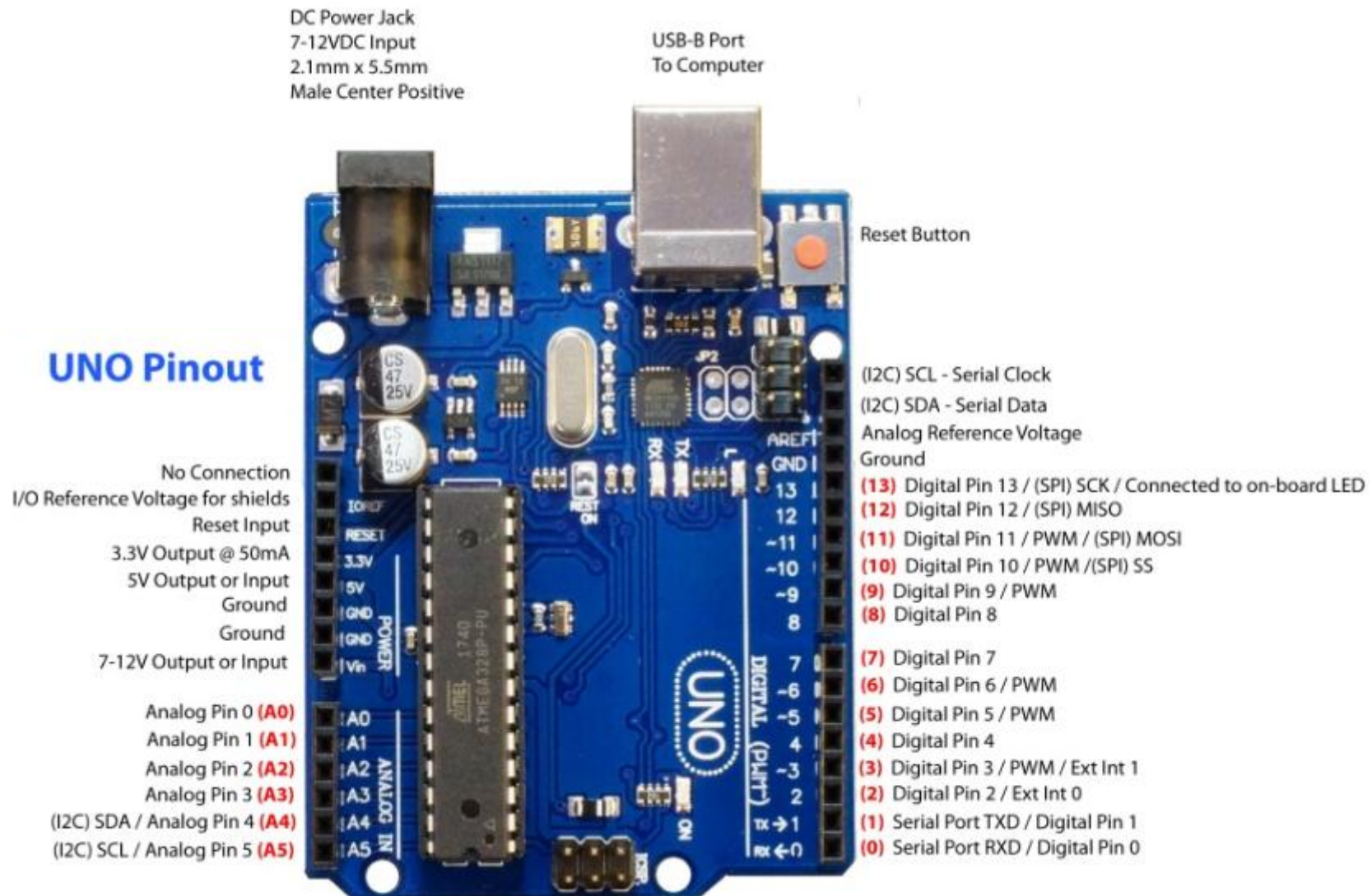
# 아두이노 소개

- **오픈소스를 기반으로 한 단일 보드 마이크로 컨트롤러**
- **작은 컴퓨터**
  - 80년대 중후반에 나온 80286PC(16MHz) 정도의 속도로 동작하는 컴퓨터
  - 메모리나 저장공간, 입출력 장치가 매우 제한적이거나 없는 대신 다른 LED, 모터, 입력장치, 출력장치 등등을 제어하는데 특화된 컴퓨터
- **설계도가 모두 공개되어 있는 오픈 하드웨어**
  - 정식 아두이노 보드와 아두이노 호환 보드

# 아두이노 UNO와 D1 R1(WiFi)

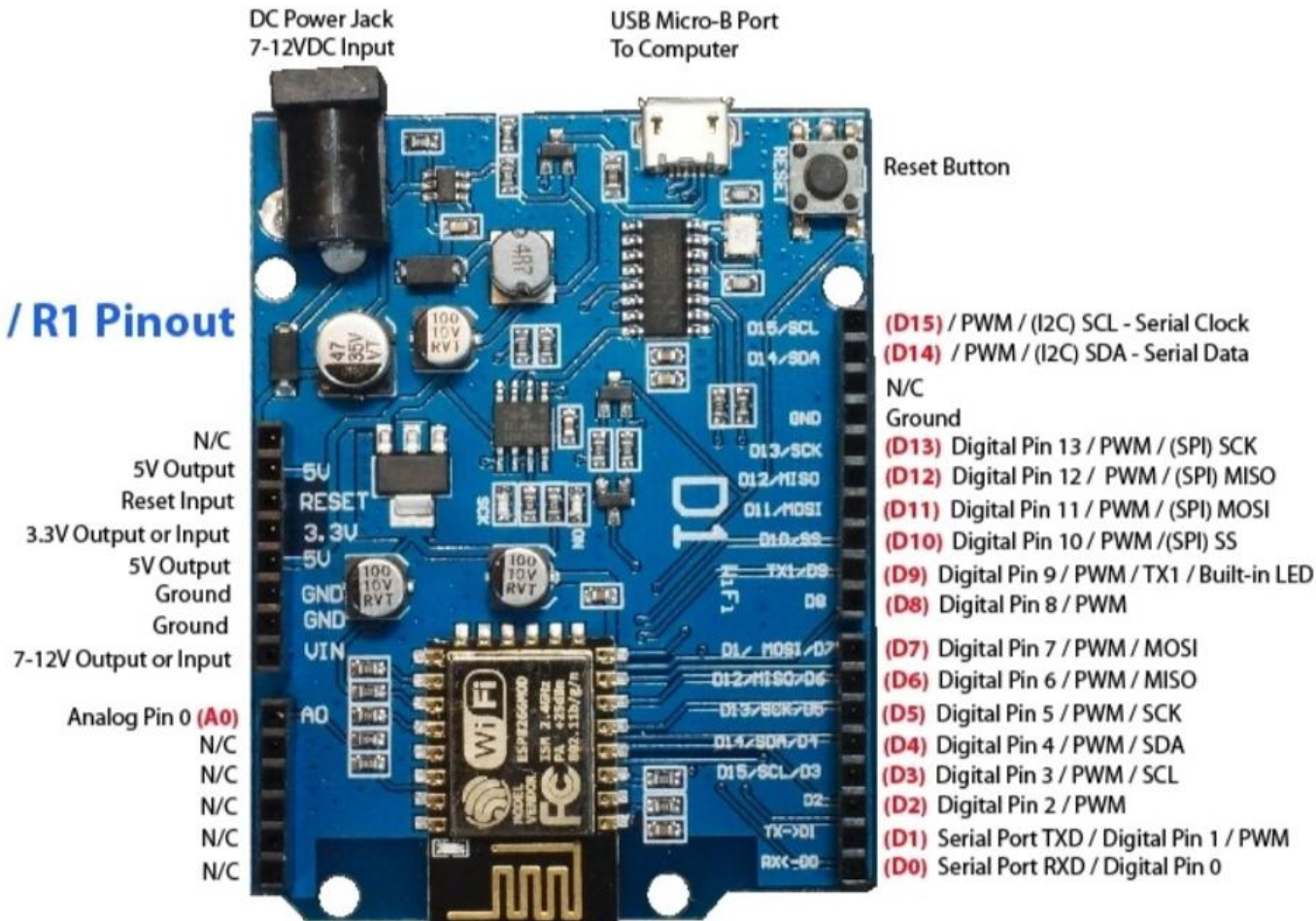


# 아두이노 UNO Pinout



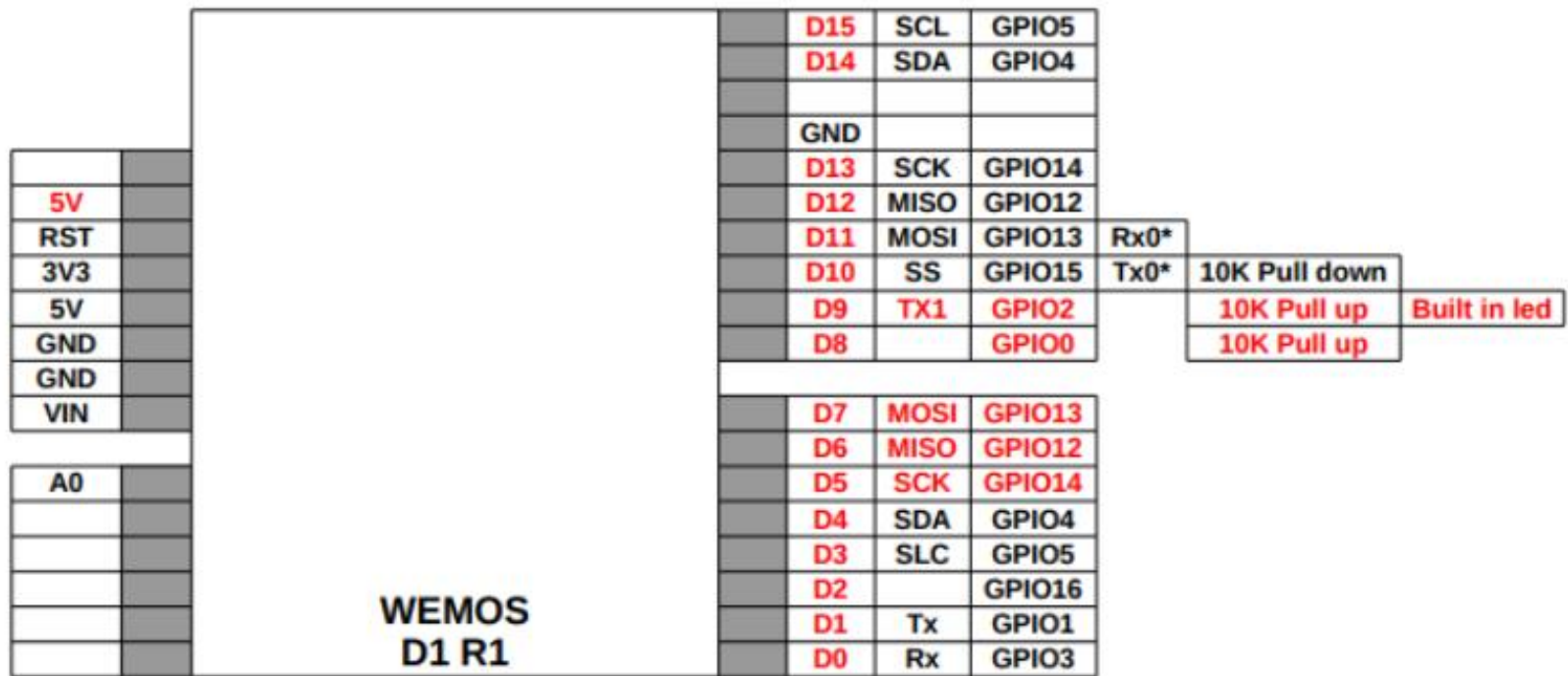
# D1 R1 Pinout

## D1 / R1 Pinout

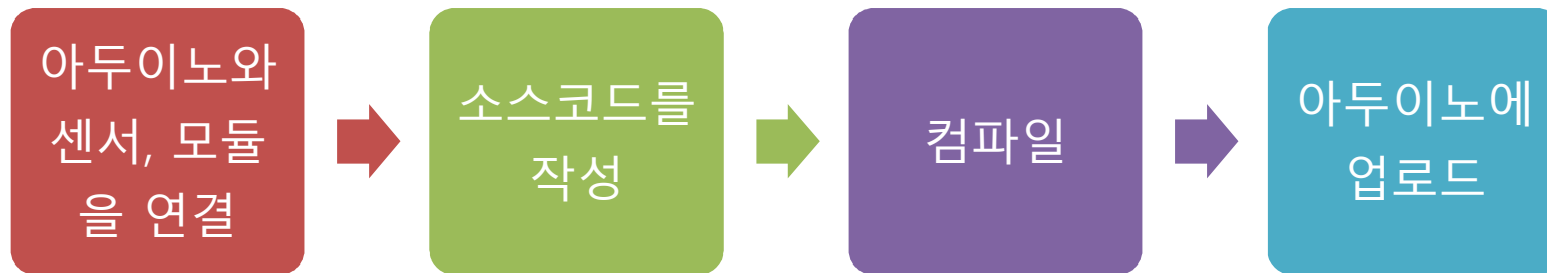




# D1 R1

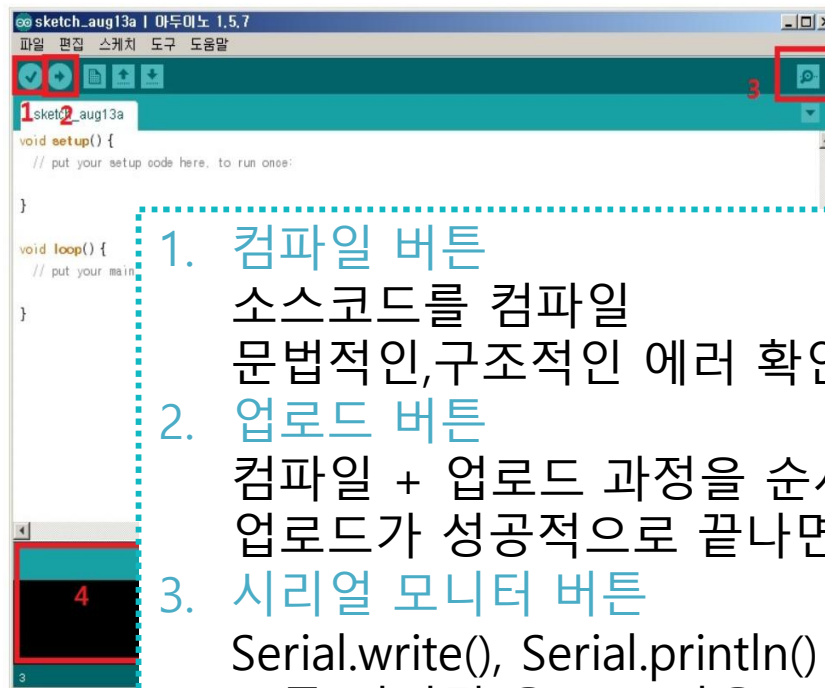


# 아두이노를 동작시키기 위한 과정



# 아두이노 개발환경

- <https://www.arduino.cc/> 다운로드



## 1. 컴파일 버튼

소스코드를 컴파일  
문법적인,구조적인 에러 확인

## 2. 업로드 버튼

컴파일 + 업로드 과정을 순서대로 실행  
업로드가 성공적으로 끝나면 4번 영역에 '업로드 완료' 메시지가 표시

## 3. 시리얼 모니터 버튼

Serial.write(), Serial.println() 등의 함수를 사용해서 serial 출력  
보통 디버깅 용도로 사용

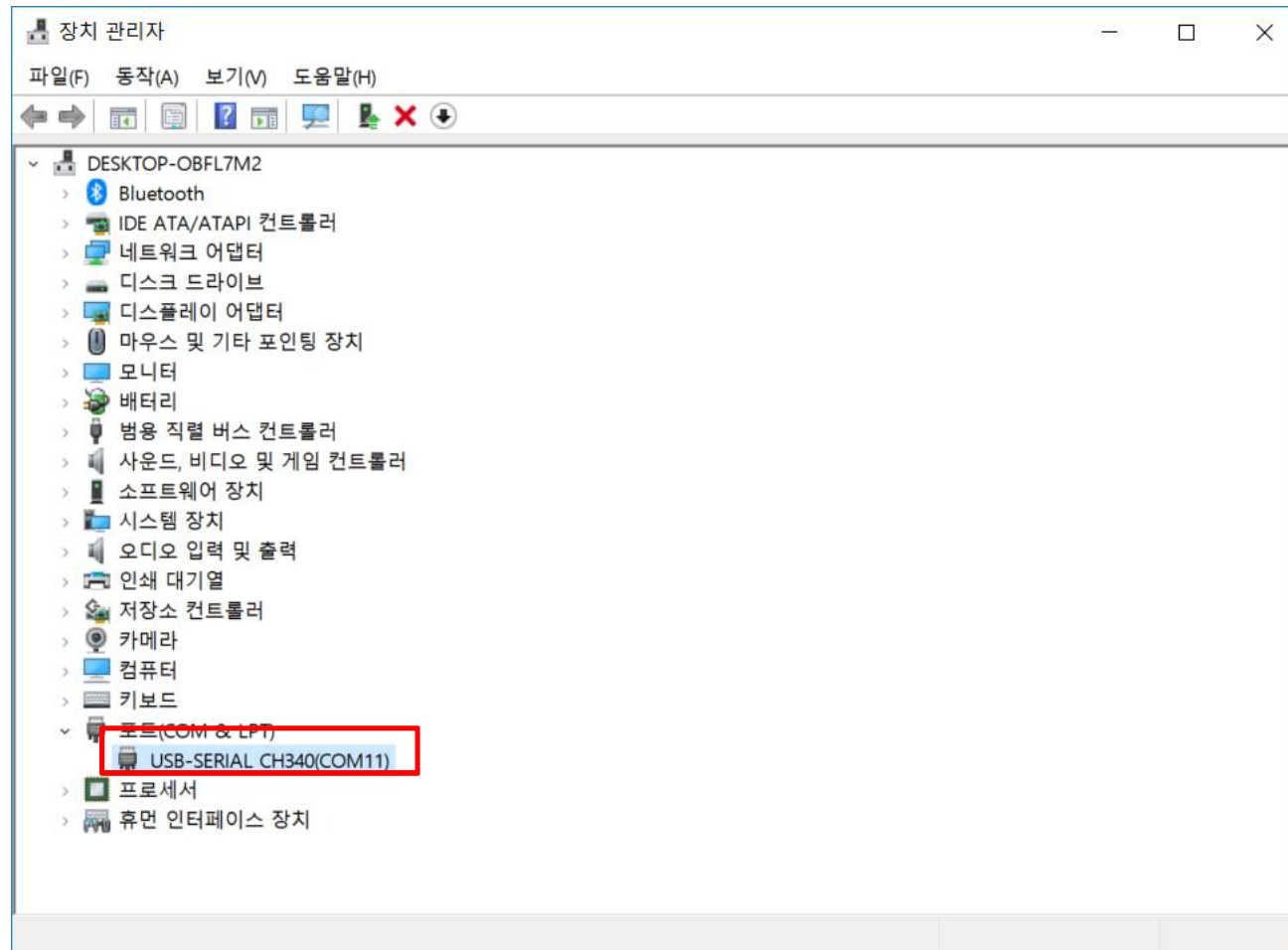
## 4. 메시지 영역

알림 메시지가나 컴파일, 업로드 결과 등등 메시지가 표시되는 영역



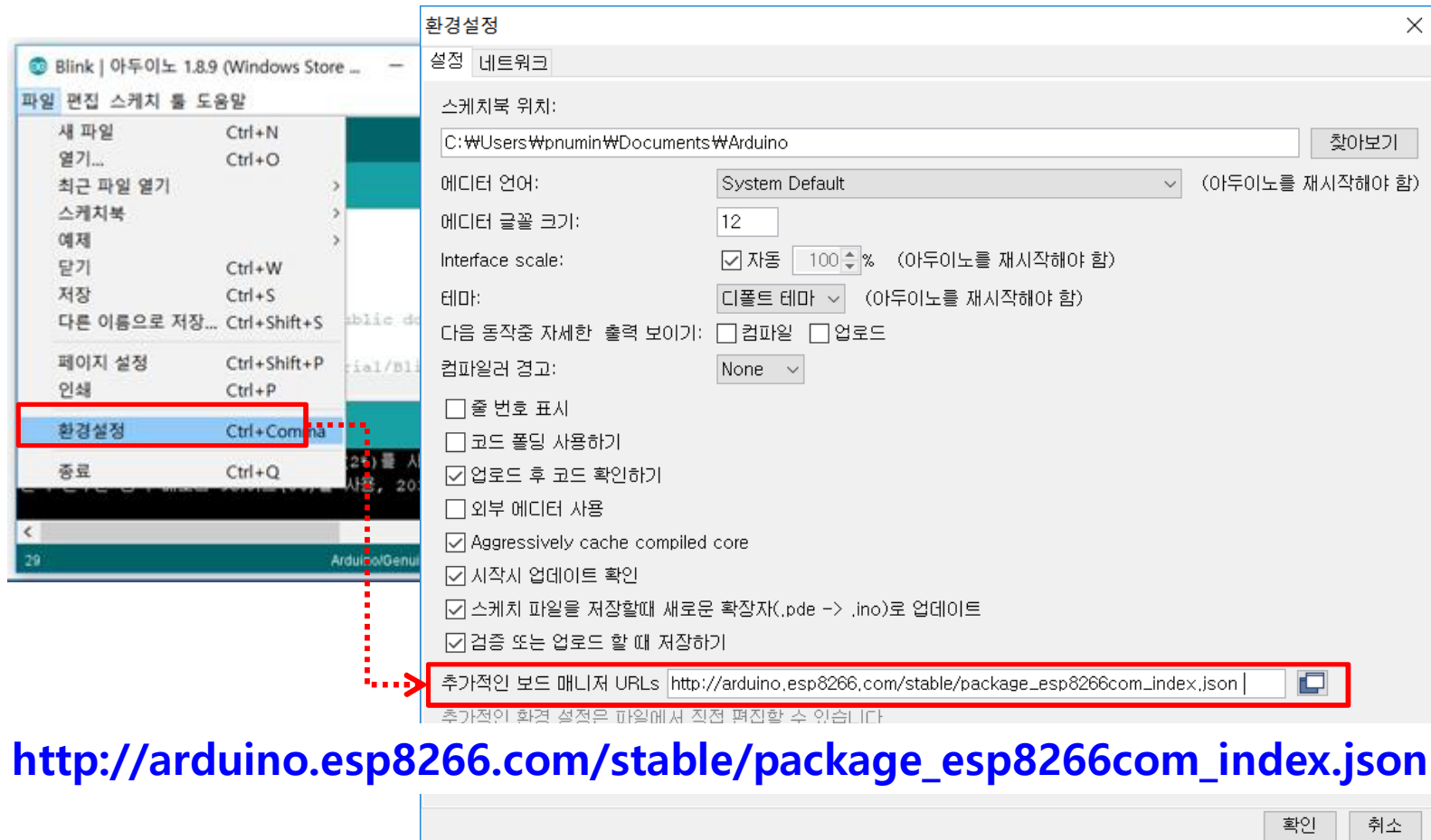
# 아두이노 UNO WiFi D1 R1 환경구축

- 장치관리자에서 확인



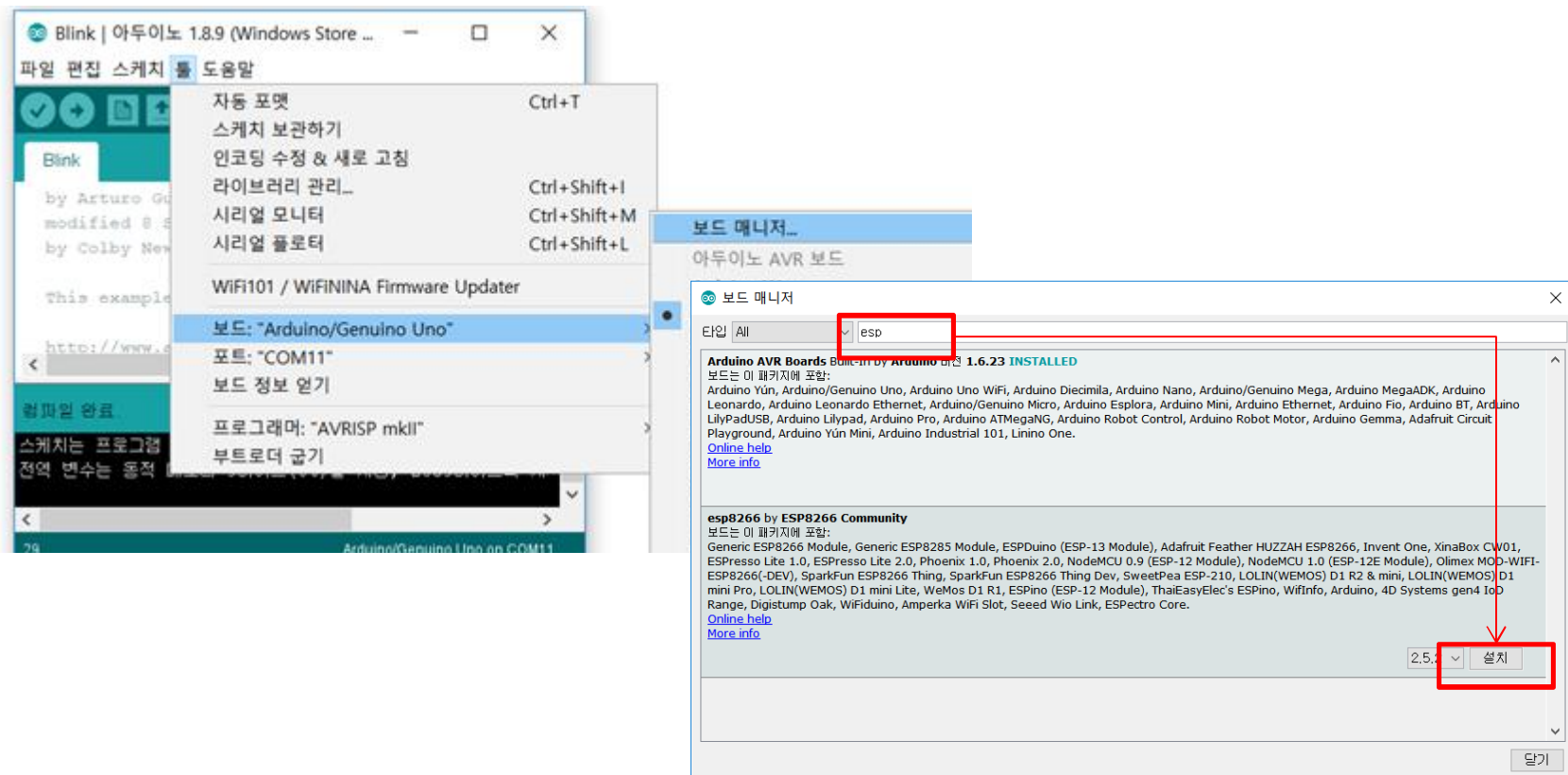
# 아두이노 UNO WiFi D1 R1 환경구축

## • 아두이노 IDE에서 보드 추가



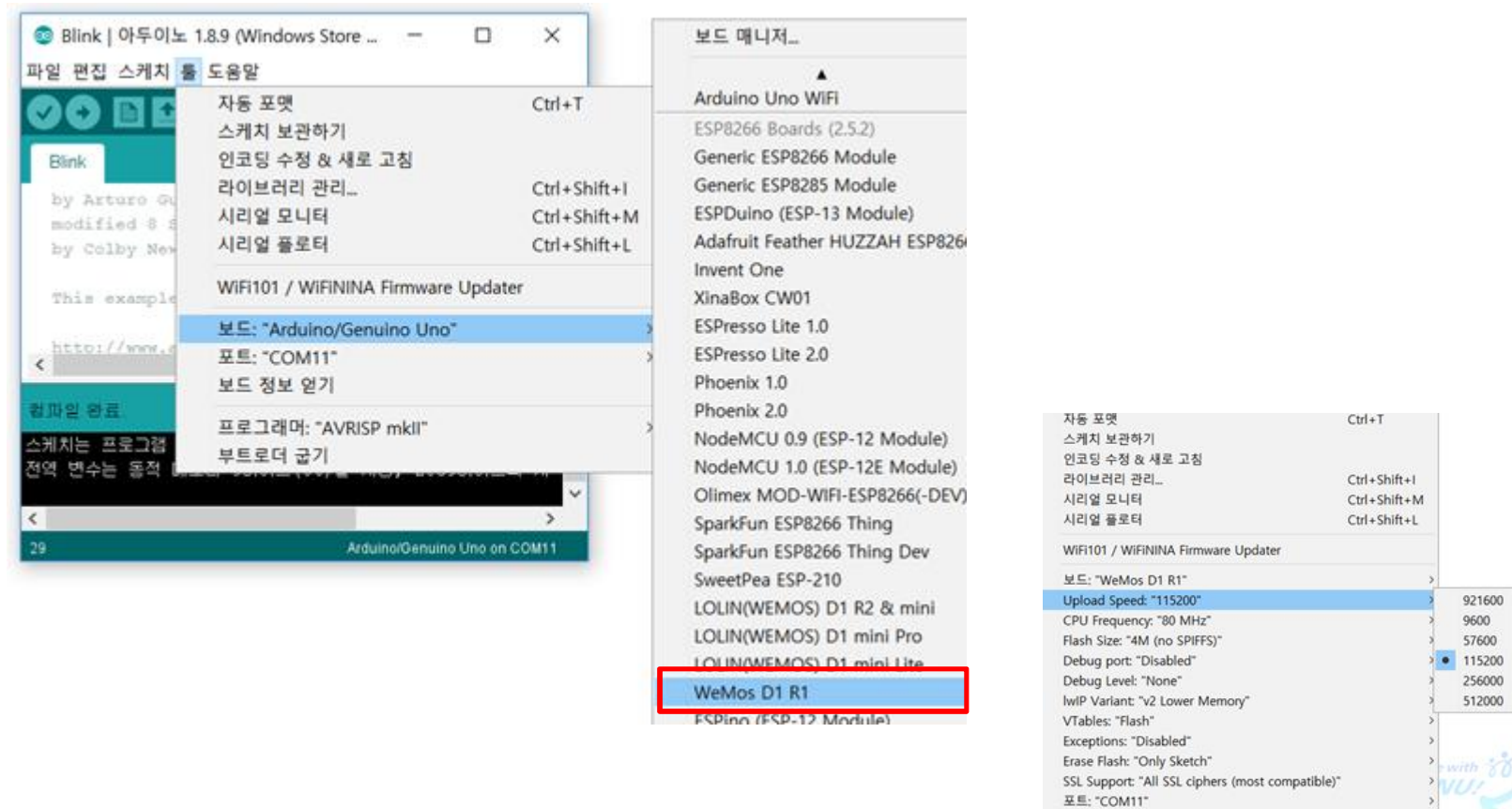
# 아두이노 UNO WiFi D1 R1 환경구축

## • 아두이노 IDE에서 보드 추가



# 아두이노 UNO WiFi D1 R1 환경구축

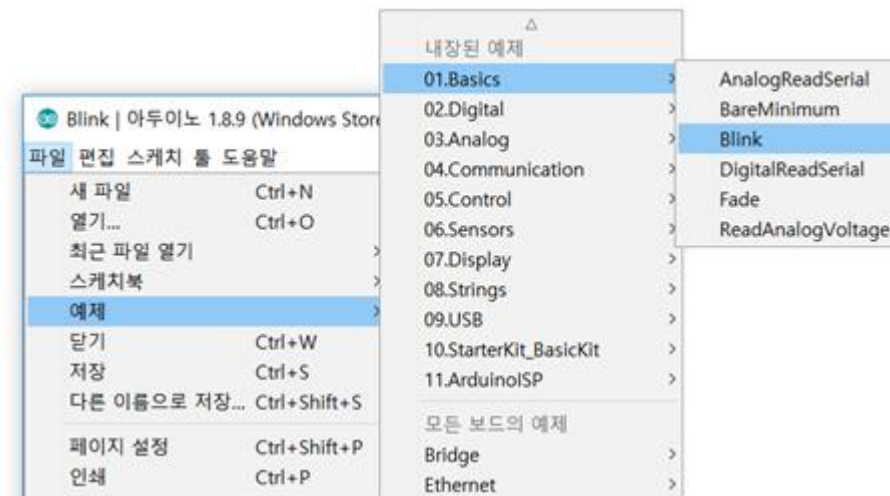
## • 아두이노 IDE에서 보드 설정



# 아두이노 UNO WiFi D1 R1 환경구축

- 설정 테스트

- 파일 > 예제 > 01.Basics > Blink
- 컴파일하고 업로드



# 아두이노 소스의 기본구조

```
sketch_nov06a §  
/*  
 * 주석  
 */  
  
int led = 13;  
  
void setup() {  
  pinMode(led, OUTPUT);  
}  
  
void loop() {  
  digitalWrite(led, HIGH);  
  delay(1000);  
  digitalWrite(led, LOW);  
  delay(1000);  
}
```

선언부

setup()

loop()

무한반복





# 시리얼 통신



# 시리얼 통신

The image shows the Arduino IDE interface with a project named "01\_serial". The code in the main editor is as follows:

```
//변수 선언
char inData ;

//시리얼 통신
void setup() {
  //1.시리얼 통신 초기화
  Serial.begin(115200);
}

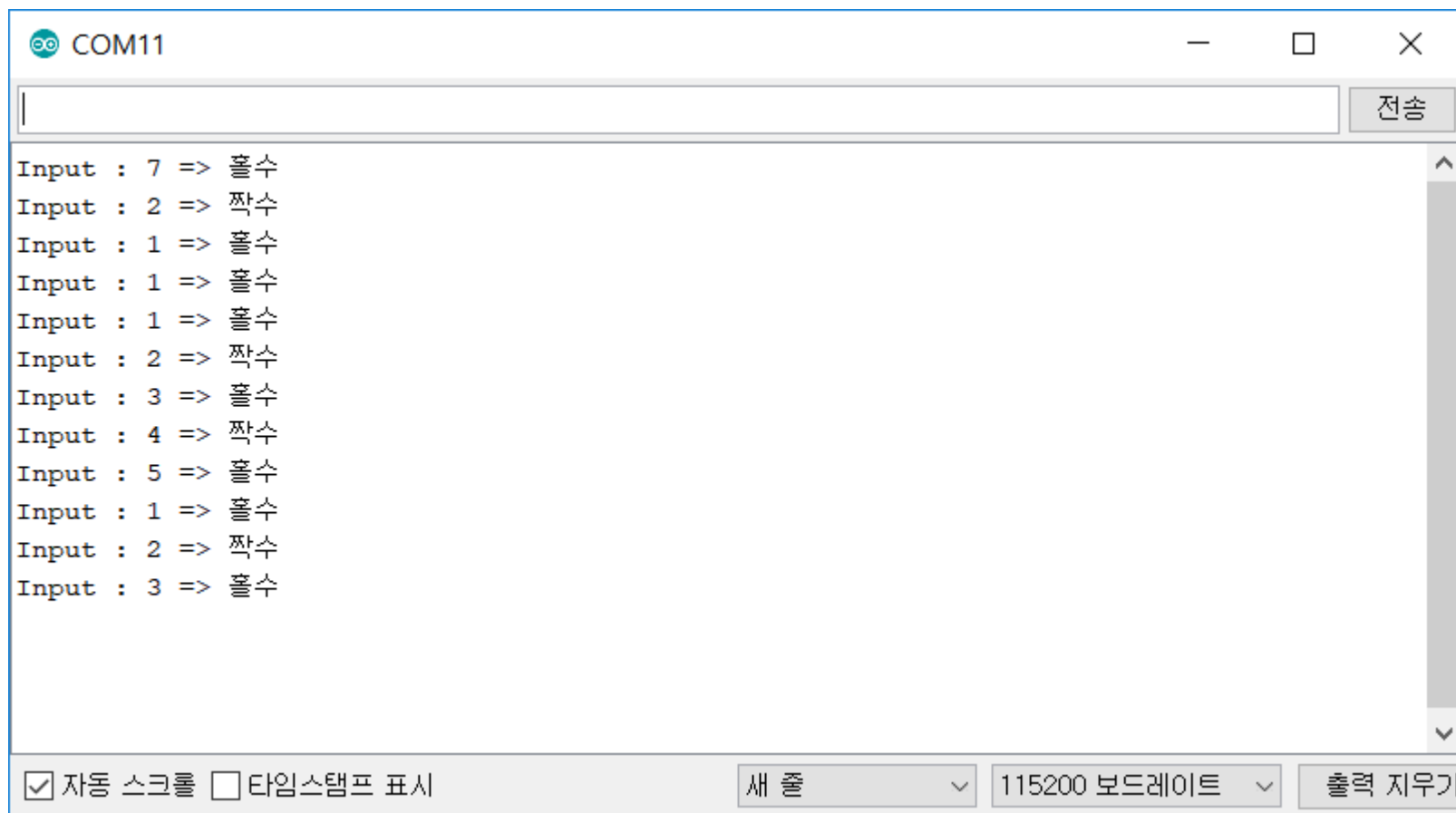
void loop() {
  //2.시리얼 통신이 가능한 동안 계속 처리
  while(Serial.available() > 0){
    inData = Serial.read();
    Serial.print("Input : ");
    Serial.println(inData);
  }
}
```

A red dashed arrow points from the serial monitor icon in the top toolbar to the "COM11" serial monitor window. The serial monitor displays the following input sequence:

```
Input : H
Input : e
Input : l
Input : l
Input : o
Input : 
Input : W
Input : o
Input : r
Input : l
Input : d
Input : !
Input : !
Input :
```

At the bottom of the serial monitor window, the baud rate is set to "115200 보드레이트" (115200 baud rate), which is highlighted with a red box. The status bar at the bottom of the IDE indicates "WeMos D1 R1 on COM11".

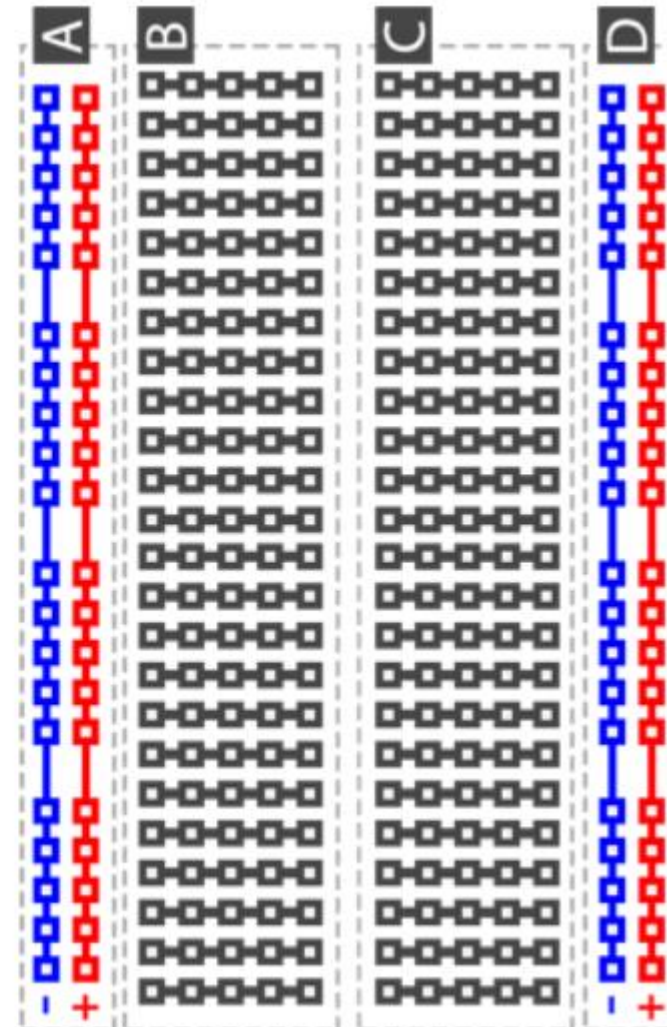
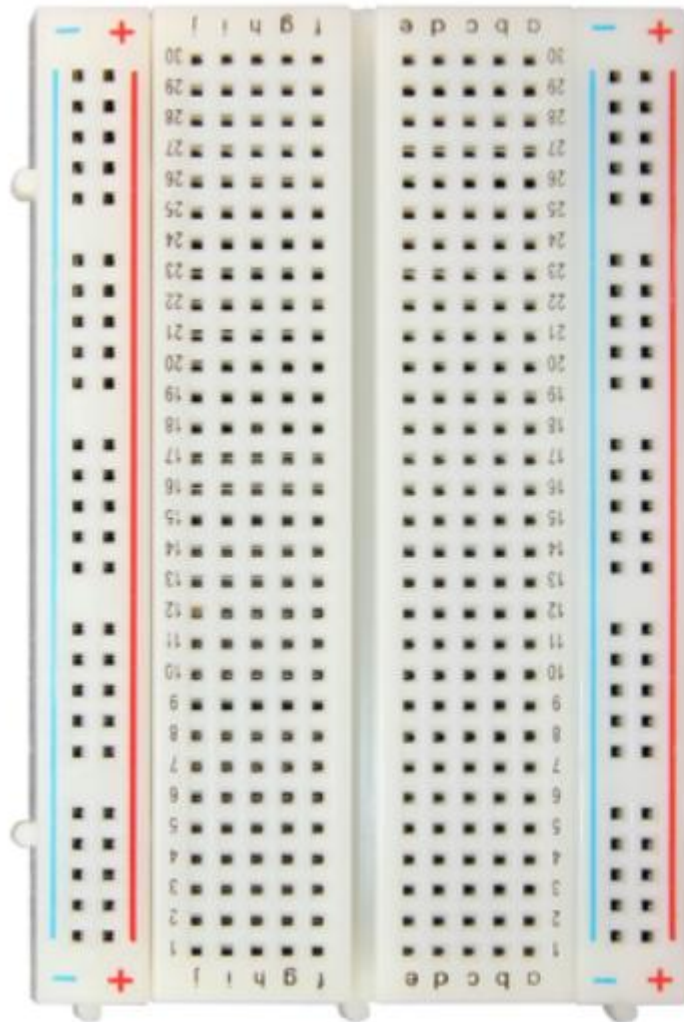
# 해결문제



# LED 제어하기 (digital output)



# 아두이노 부품-브레드보드



# 아두이노 부품-LED



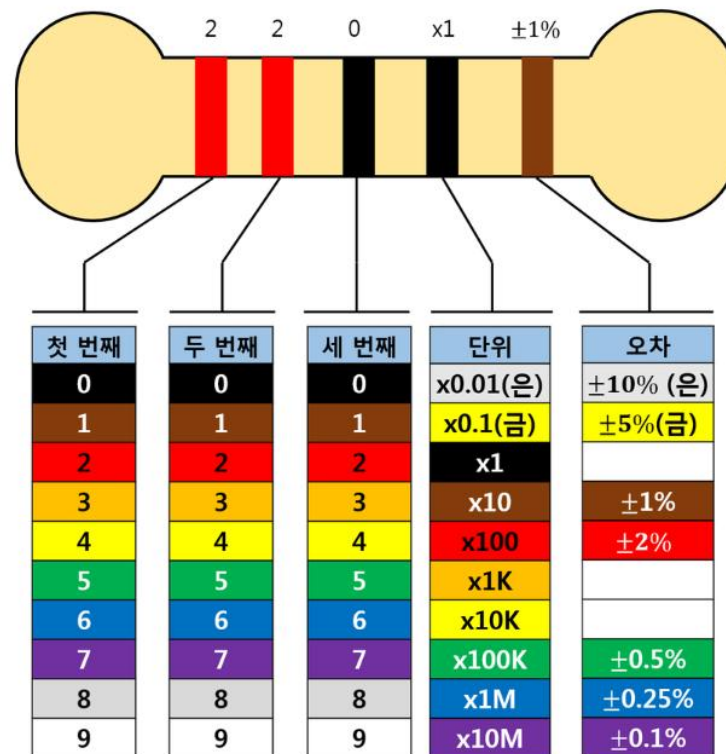
- 매우 작은 전류를 사용하며 빛을 발산하는 부품
- LED를 전원소스에 직접 연결하면 LED는 타버리게 됨

- 저항을 이용하여 LED로 들어오는 전류의 양을 제한





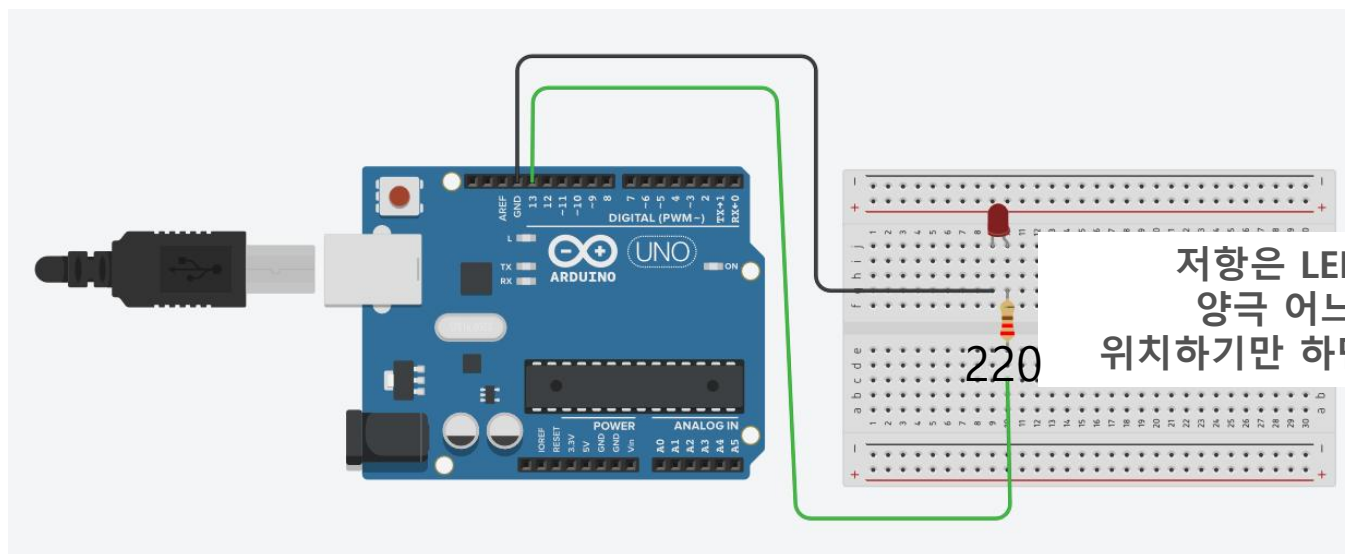
# 아두이노 부품-저항



< 220 Ω 저항 >

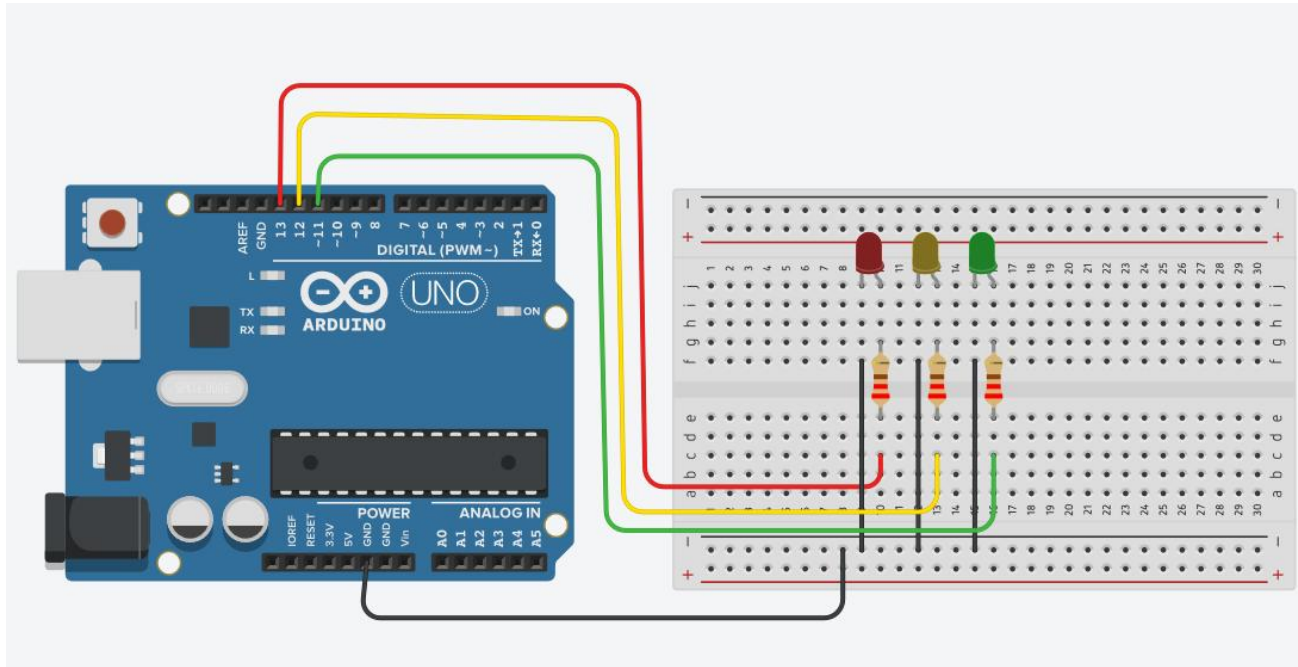
색	값
검정색	0
갈색	1
빨강색	2
주황색	3
노란색	4
초록색	5
파란색	6
보라색	7
회색	8
하얀색	9
은색	±10%
금색	±5%

# LED 불 깜빡이기



```
void setup() {  
  // initialize digital pin LED as an output.  
  pinMode(13, OUTPUT);  
}  
  
void loop() {  
  digitalWrite(13, HIGH); // turn the LED on (HIGH is the voltage level)  
  delay(1000);           // wait for a second  
  digitalWrite(13, LOW);  // turn the LED off by making the voltage LOW  
  delay(1000);           // wait for a second  
}
```

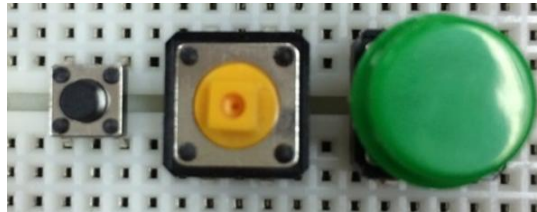
# 해결문제



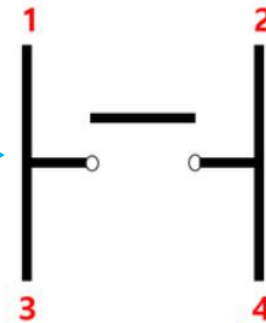
# 버튼제어하기 (digital input)



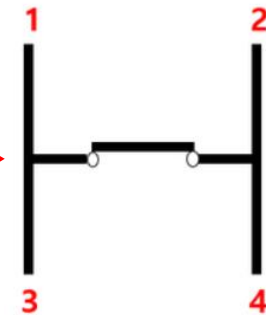
# 아두이노 부품-push 버튼



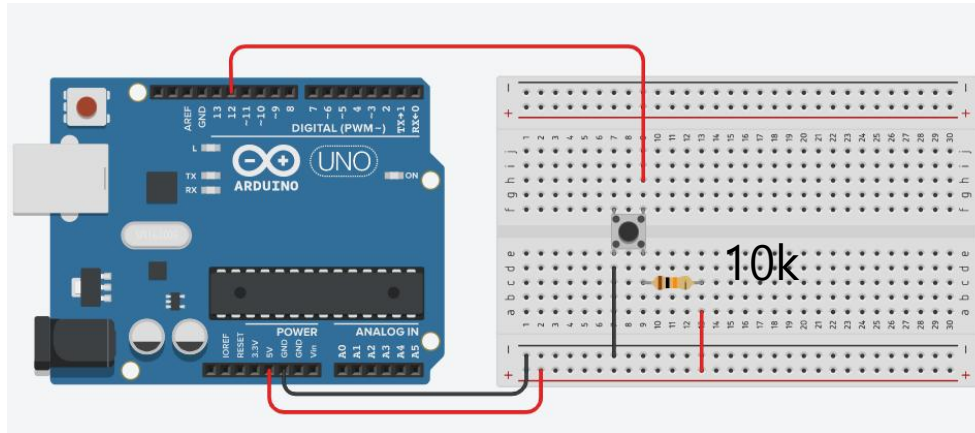
누르지 않은 경우



누른 경우



# 버튼 입력



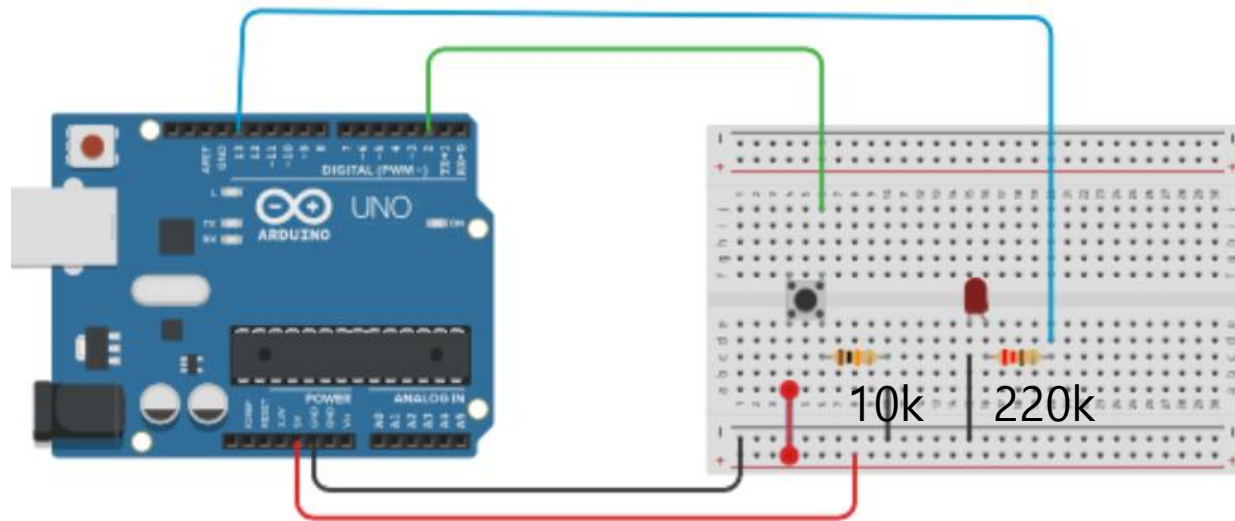
```
//디지털 입력 저장변수  
int din = 0;  
//디지털 입력 핀 번호  
int btnpin = 12;
```

```
void setup() {  
  Serial.begin(115200);  
  //입력핀 설정  
  pinMode(btnpin, INPUT);  
}
```

```
void loop() {  
  //디지털 입력  
  din = digitalRead(btnpin);  
  //입력값 시리얼 통신  
  Serial.println(din);  
  delay(100);  
}
```



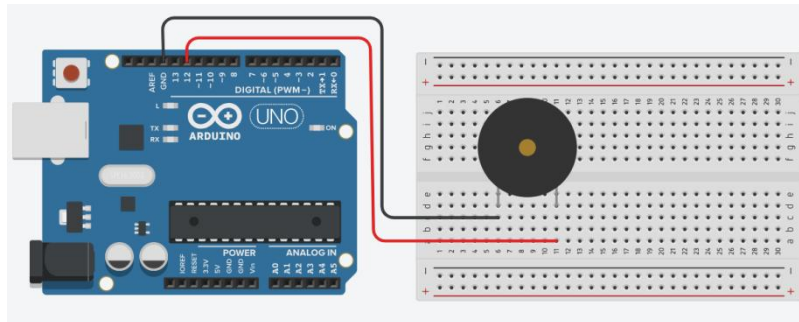
# 해결문제



# 수동부저



# 아두이노 부품 - 피에조 부저



//음정의

#define Do 262

#define Re 294

#define Mi 330

#define Fa 349

#define Sol 392

#define Ra 440

#define Si 494

#define Do2 523

//부저 핀번호

int buzzerPin = 12;

//노래

int play[] = {Sol,Sol,Ra,Ra,Sol,Sol,Mi,Sol,Sol,Mi,Mi,Re} ;

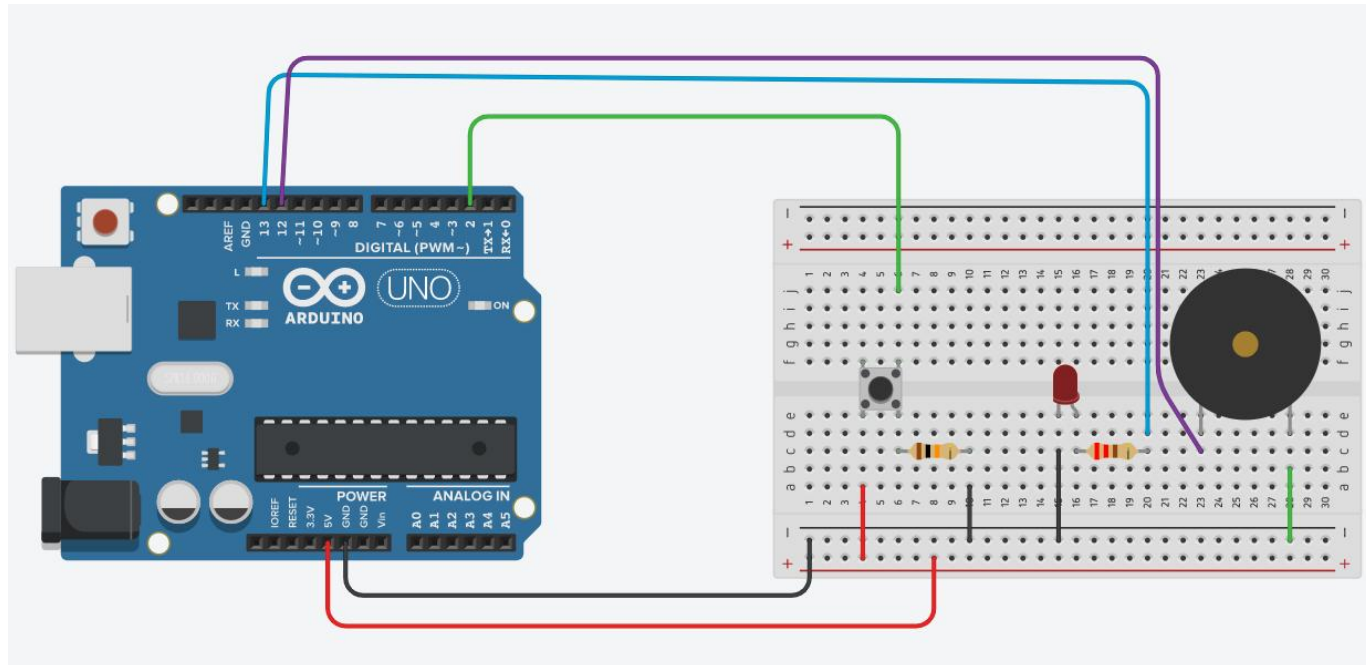
int i;

```
void setup() {  
  pinMode(buzzerPin, OUTPUT);  
}
```

```
void loop() {  
  for(i=0; i<12; i++){  
    tone(buzzerPin, play[i], 200);  
    delay(500);  
  }  
  noTone(buzzerPin);  
}
```



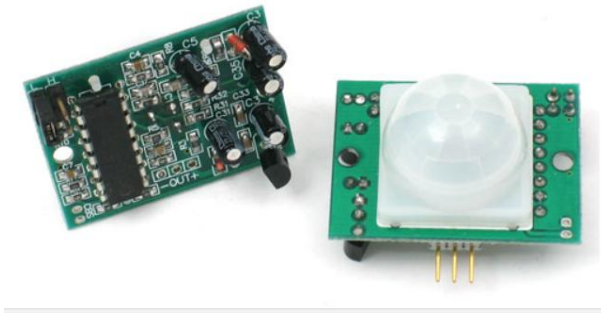
# 해결문제



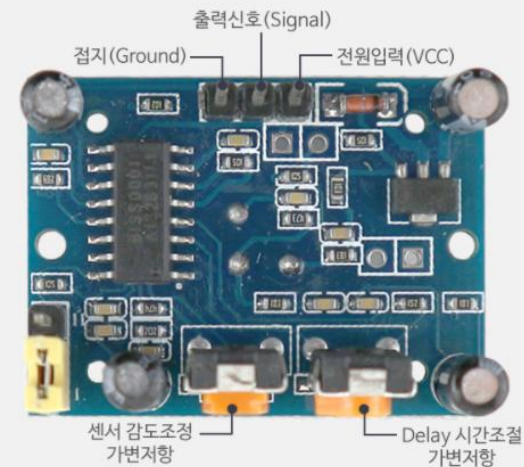
# 적외선 모션 감지 센서



# 아두이노 부품-모션감지센서

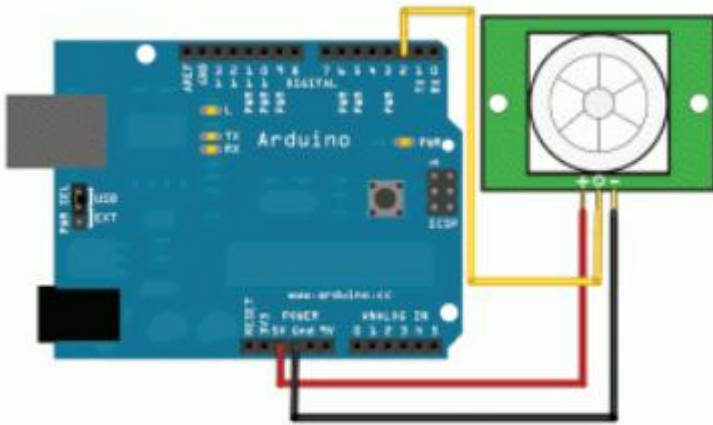


## RFID-RC522 PIN MAP





# 적외선 모션 감지 센서

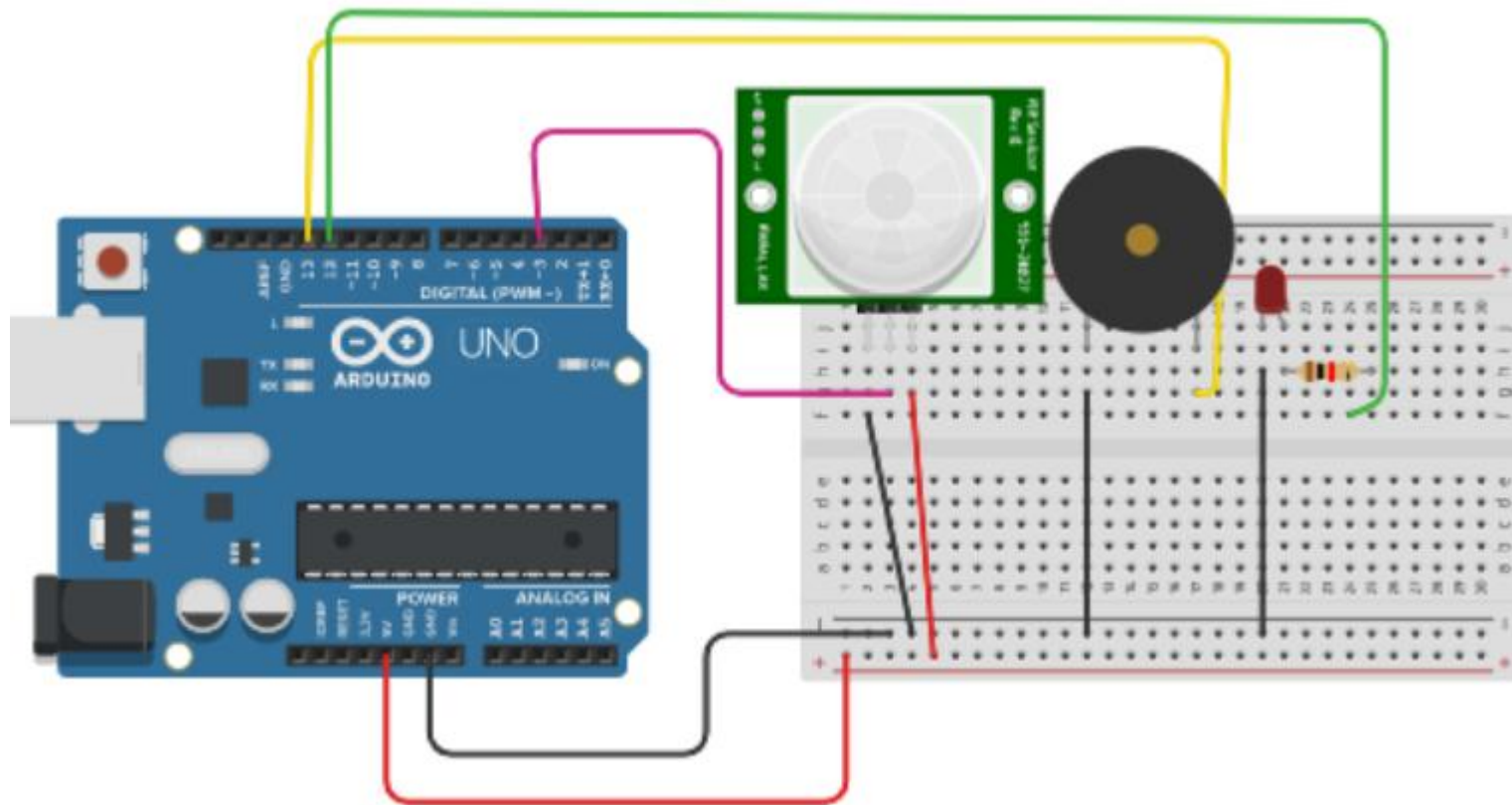


```
//핀번호  
int pirPin = 5;
```

```
//입력변수  
int din;  
void setup() {  
  Serial.begin(115200);  
  pinMode(pirPin, INPUT);  
}  
void loop() {  
  din = digitalRead(pirPin);  
  Serial.println(din);  
  delay(100);  
}
```



# 해결문제



# 온도 감지 센서




# 라이브러리 추가

- 라이브러리

- 특정 목적을 위해 사용되는 파일들을 모아놓은 집합
- 새로운 센서를 쓸 때 우리가 직접 컨트롤 하기에는 너무 복잡하기 때문에 이런 일을 쉽게 할 수 있도록 해주는 라이브러리들을 사용

- 라이브러리 설치 방법

- [스케치 - 라이브러리 가져오기] 메뉴로 설치
  - [스케치 - 라이브러리 가져오기 - 라이브러리 추가...] 메뉴를 선택합니다. 그리고 다운로드 받은 라이브러리 ZIP 파일의 위치를 지정
- 직접 설치
  - 아두이노 개발환경을 종료
  - [아두이노 설치폴더\libraries] 폴더로 이동
    - 라이브러리 폴더를 만들어 파일을 넣으면 아두이노 개발환경이 시작할 때 인식
  - 아두이노 개발환경을 실행하고 [스케치 - 라이브러리 가져오기] 메뉴를 확인
  - 예제는 [파일 - 예제] 메뉴를 확인

Come with 

- libraries\ArduinoTest (폴더)
- libraries\ArduinoTest\ArduinoTest.cpp
- libraries\ArduinoTest\ArduinoTest.h
- libraries\ArduinoTest\examples (폴더)

# DHT11 라이브러리 추가

내 PC > 문서 > Arduino > libraries


이름

arduino\_43861

arduino\_633772

arduino\_872104

Arduinolson

 DHT



» [내 PC](#) » [문서](#) » [Arduino](#) » [libraries](#) » [DHT](#)

이름


 .github

### examples

Adafruit\_Sensor.h

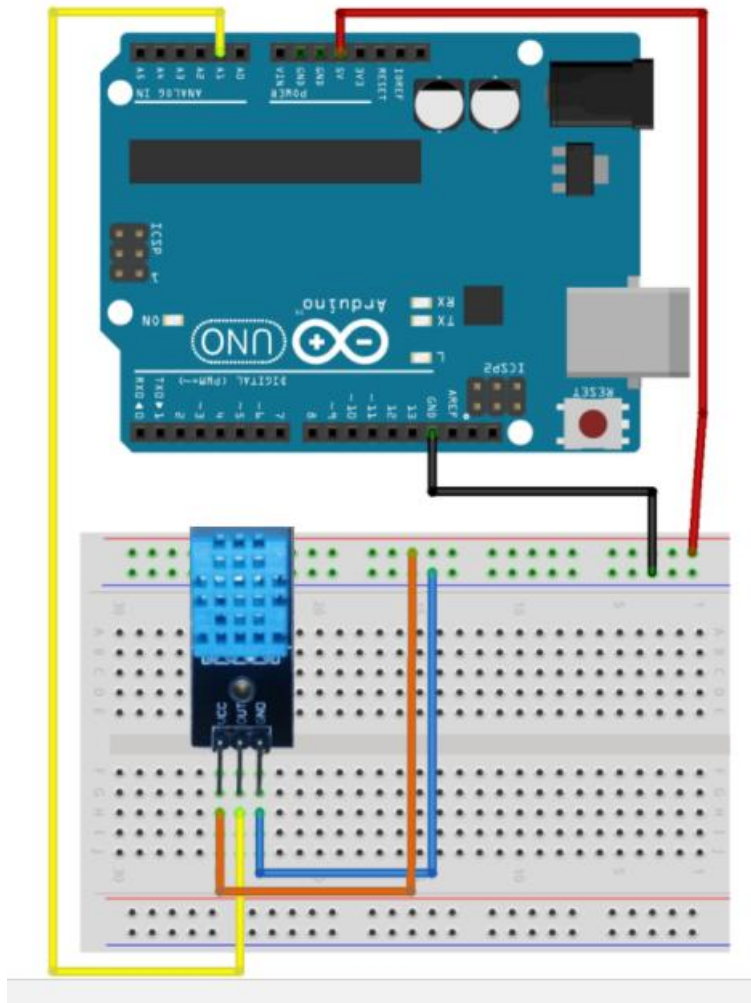
 DHT.cpp DHT.h DHT\_U.cpp DHT\_U.h

 keywords.txt

 library.properties

 README.md

# 온도 습도 측정



```
#include "DHT.h"
#define DHTTYPE DHT11 // DHT 11
```

```
// DHT Sensor
const int DHTPin = 5;
DHT dht(DHTPin, DHTTYPE);
```

```
void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  delay(10);
```

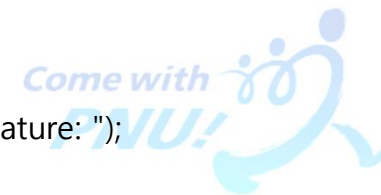
```
  dht.begin();
}
```

```
void loop() {
  // put your main code here, to run repeatedly:
  float h = dht.readHumidity();
  // Read temperature as Celsius (the default)
  float t = dht.readTemperature();
```

```
  if (isnan(h) || isnan(t)) {
    Serial.println("Failed to read from DHT sensor!");
  }
```

```
  else{
    Serial.print("Humidity: ");
    Serial.print(h);
    Serial.print(" %Wt Temperature: ");
    Serial.print(t);
    Serial.print(" %Wn");
```

```
  }
}
```



# 아두이노(Arduino) WiFi



# D1 R1 WiFi 확인

```
#include <ESP8266WiFi.h>

#ifndef STASSID
#define STASSID "your-ssid"
#define STAPSK "your-password"
#endif

const char* ssid = STASSID;
const char* password = STAPSK;

const char* host = "djxmmx.net";
const uint16_t port = 17;

void setup() {
  Serial.begin(115200);

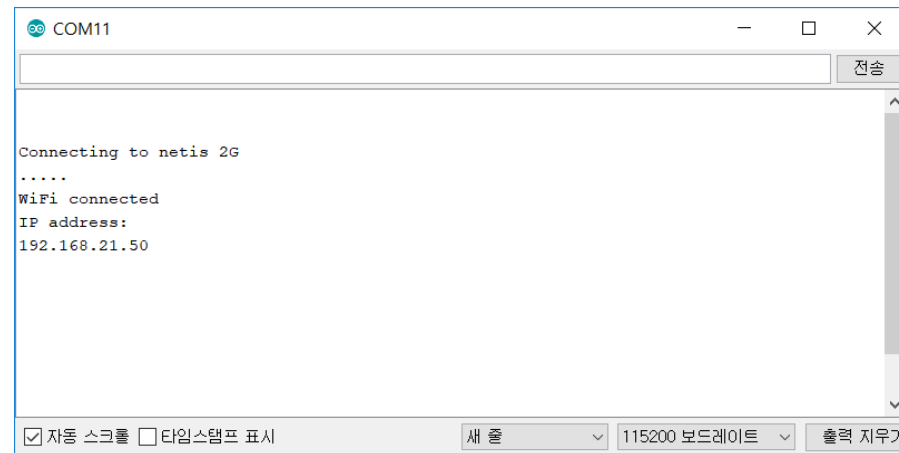
  // We start by connecting to a WiFi network
  Serial.println();
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);

  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}

void loop() {
}
```



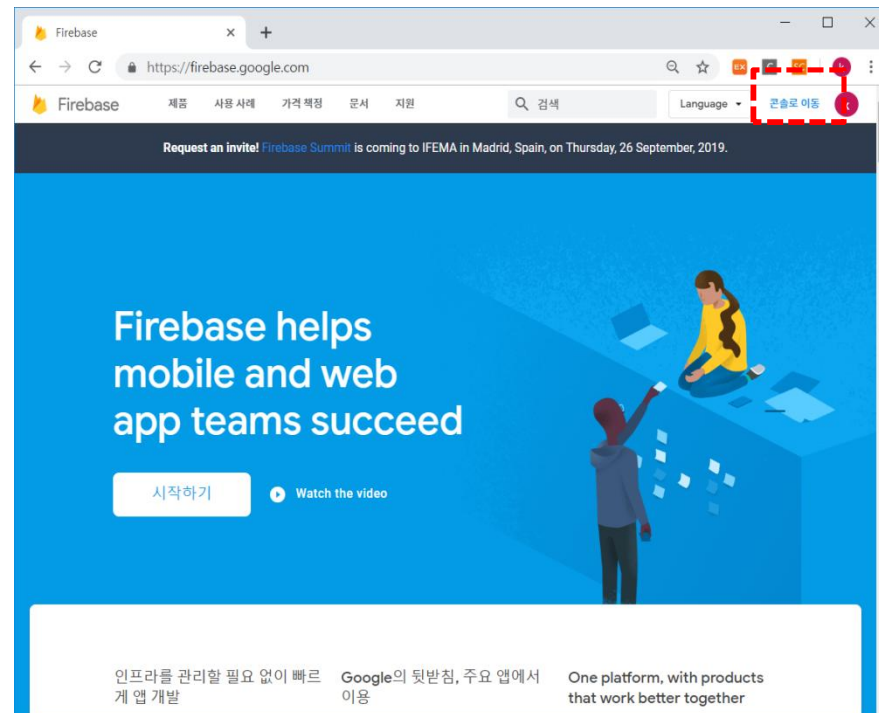


# 아두이노와 파이어베이스 연결

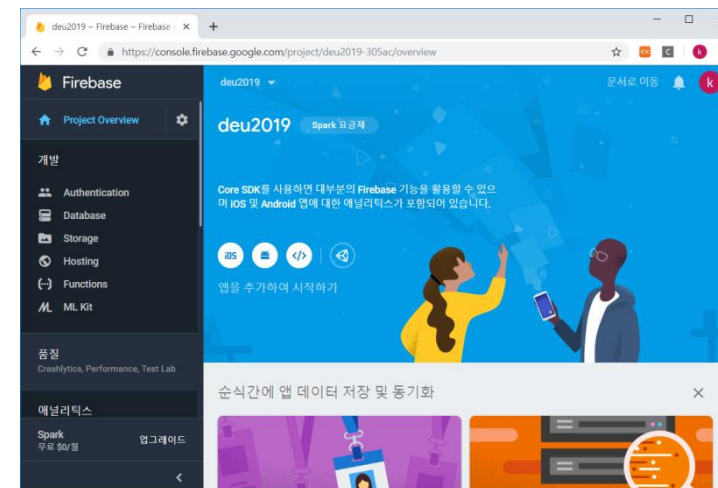
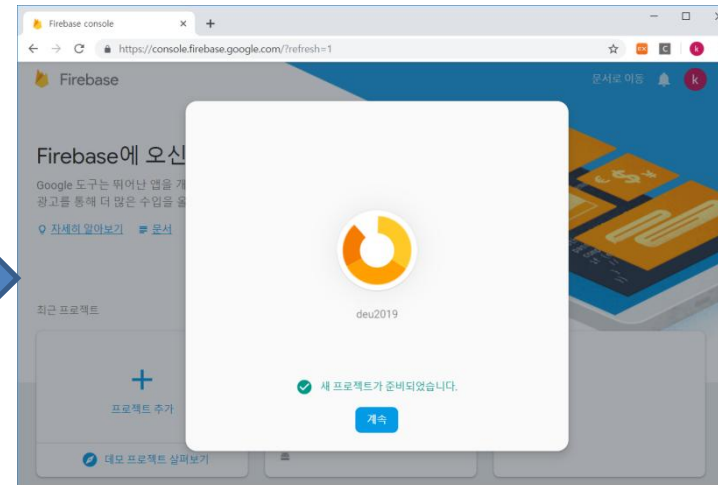
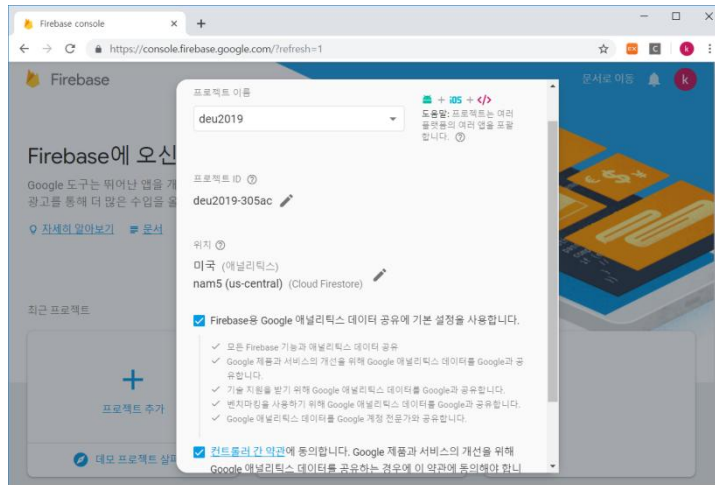


# Firebase 환경설정하기

- <https://firebase.google.com/>
- 구글 계정으로 로그인
- Console로 이동



# 프로젝트 생성



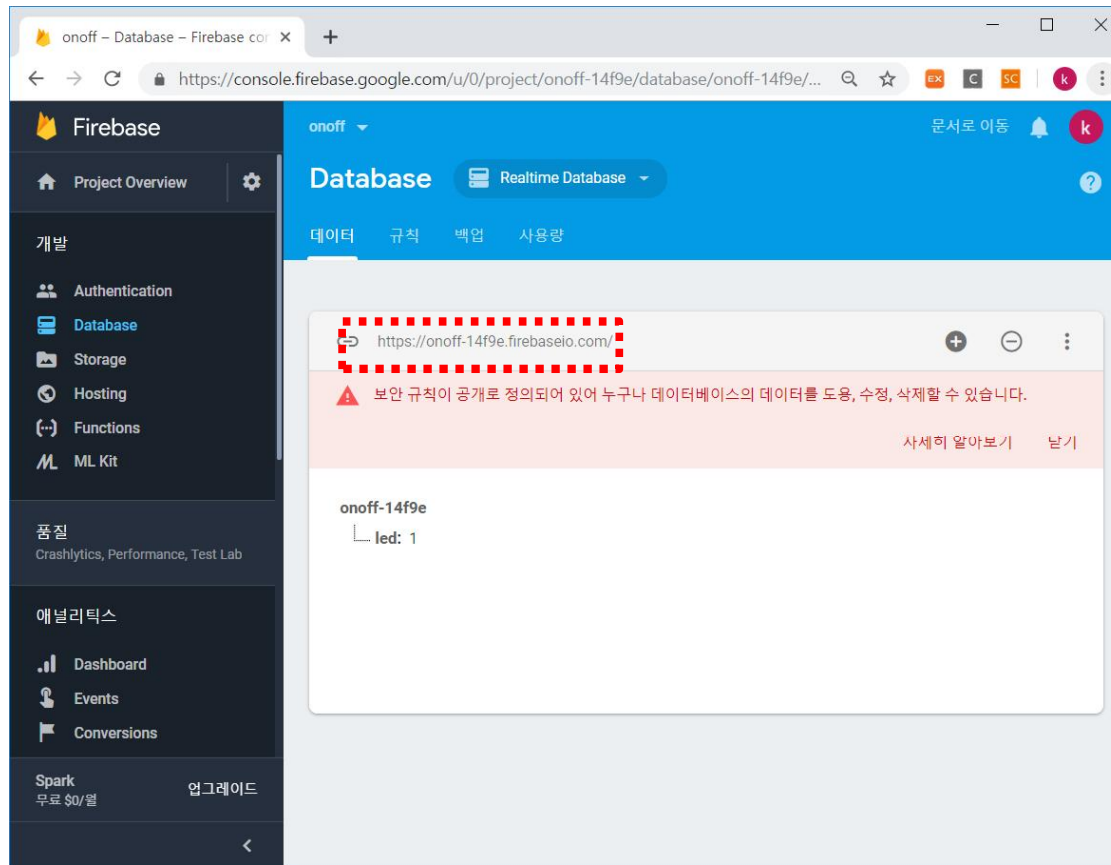
# 아두이노와 파이어베이스 연결

- [https://github.com/firebase/firebase-arduino/tree/master/examples/FirebaseDemo\\_ESP8266](https://github.com/firebase/firebase-arduino/tree/master/examples/FirebaseDemo_ESP8266)
  - Download [FirebaseArduino library](#)
  - Start Arduino
  - Click Sketch > Include Library > Add .ZIP Library...
  - Choose firebase-arduino-master.zip downloaded

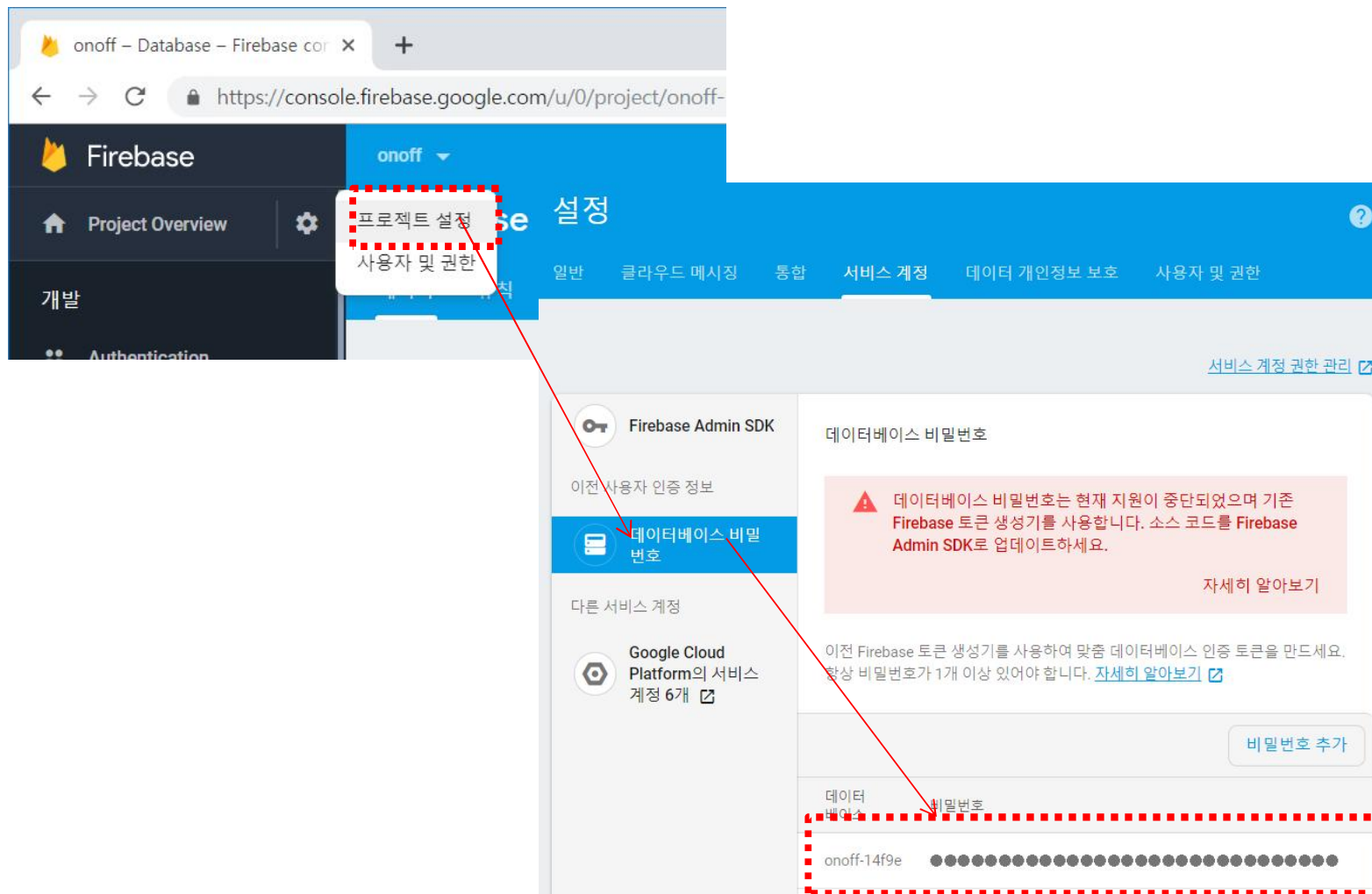


# 파이어베이스 – Realtime Database

- <https://console.firebase.google.com/>



# 파이어베이스-데이터베이스 비번



# ESP8266 WiFi 설정 및 firebase 설정

```
#include <ESP8266WiFi.h>
#include <FirebaseArduino.h>

// Set these to run example.
#define FIREBASE_HOST ""
#define FIREBASE_AUTH ""
#define WIFI_SSID ""
#define WIFI_PASSWORD "11110000"

void setup() {
  Serial.begin(115200);

  // connect to wifi.
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("connecting");
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }
  Serial.println();
  Serial.print("connected: ");
  Serial.println(WiFi.localIP());

  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
}
```

```
float n = 0;

void loop() {
  // set value
  Firebase.setFloat("temp", n);
  // handle error
  if (Firebase.failed()) {
    Serial.print("setting /number failed:");
    Serial.println(Firebase.error());
    return;
  }
  n = n + 0.1 ;
  delay(1000);
}
```



# 해결문제

```
int ledck ;  
void loop() {  
    ledck = Firebase.getInt("led") ;  
  
    if ( ledck == 1 ) {  
        digitalWrite(ledPin, HIGH);  
    }  
    else {  
        digitalWrite(ledPin, LOW);  
    }  
  
    Serial.println(ledck);  
    delay(1000);  
}
```

