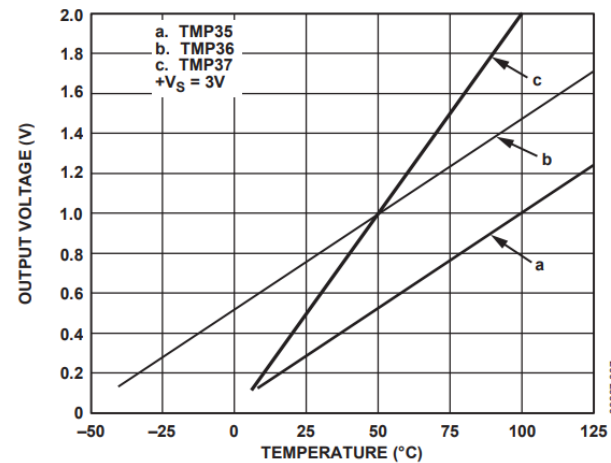


# 환경 정보 확인

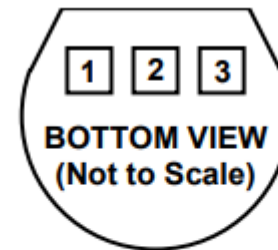
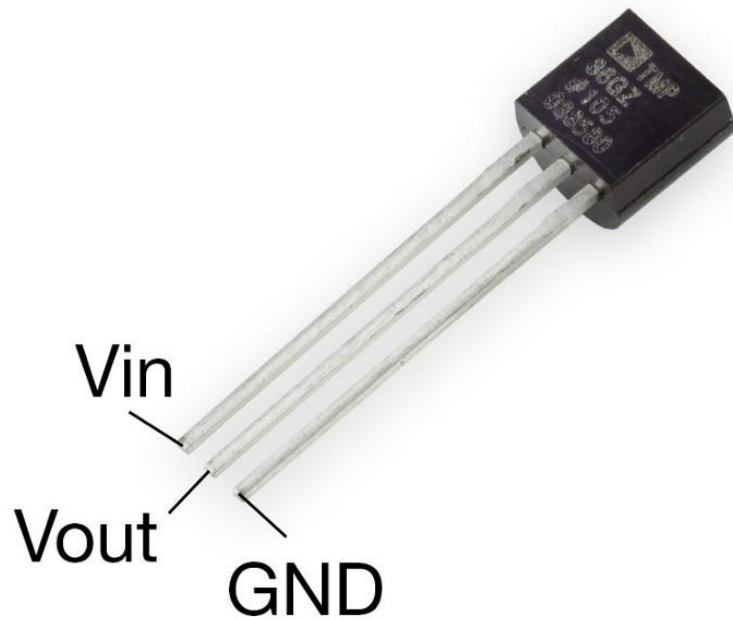


# TMP36

- 온도센서는 온도를 감지해 전기신호로 바꿔주는 센서를 의미
- TMP36
  - 상온에서 대략 750mV를 출력
  - 온도 1 °C가 변화하면 10mV의 출력 전압이 변화 함
  - 정밀도는  $\pm 1$  °C로 정밀한 온도 감지는 어려움.
  - 사용하기 쉽고 저렴하여 정밀한 온도 감지가 필요 없는 어플리케이션이 많이 사용 됨.



# TMP36 핀 연결



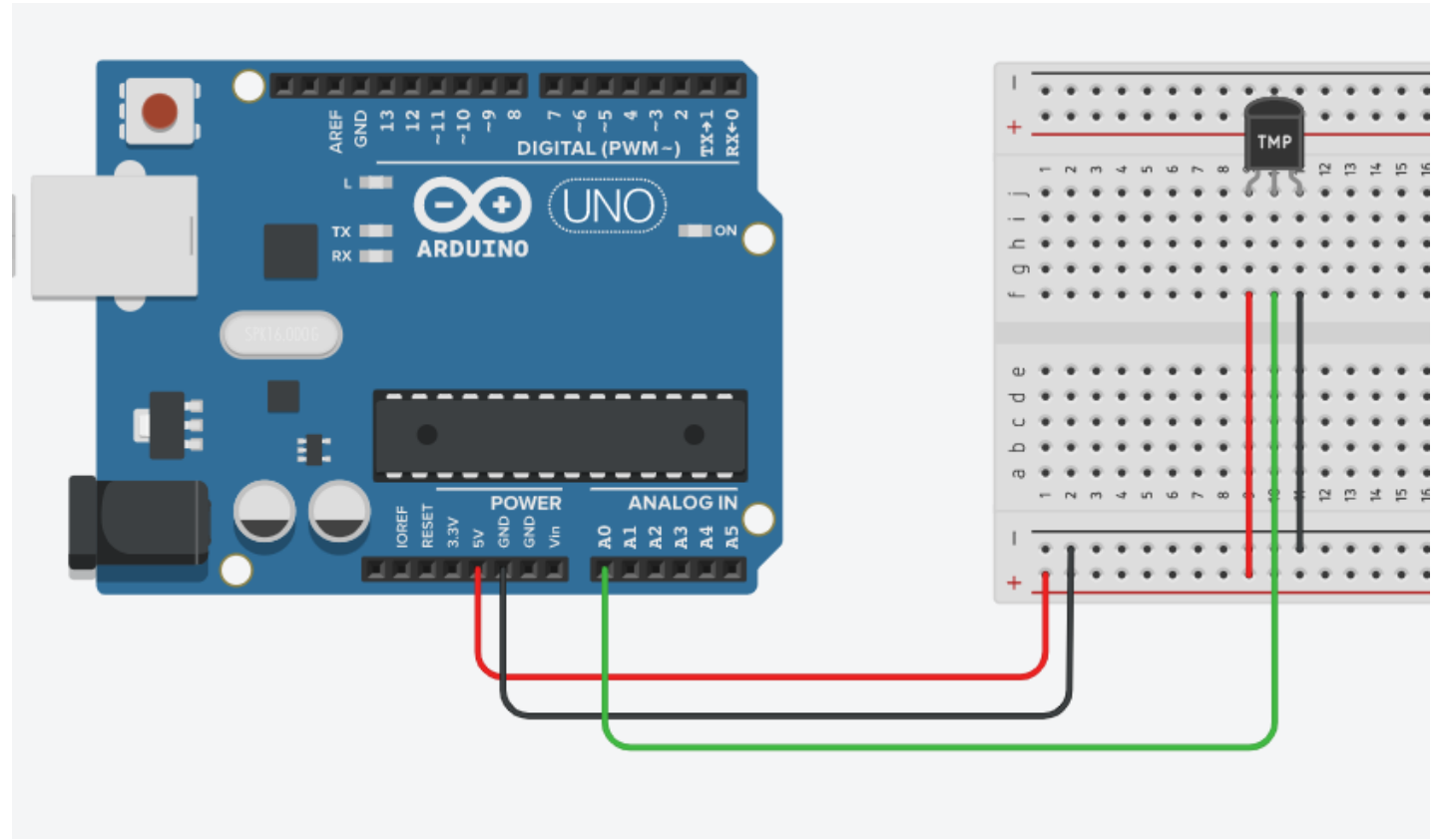
**PIN 1, +V<sub>S</sub>; PIN 2, V<sub>OUT</sub>; PIN 3, GND**

00337-004

Figure 4. T-3 (TO-92)

# TMP36 + 아두이노 실험

- TMP Vin <> 아두이노 5V
- TMP Vout <> 아두이노 A0
- TMP GND <> 아두이노 GND



# 코드 작성

**void setup()**

```
{  
  Serial.begin(9600);  
}
```

**void loop()**

```
{  
  int reading = analogRead(A0);  
  
  float voltage = (reading / 1024.0) * 5.0;  
  
  Serial.print(voltage); Serial.println(" volts");  
  
  float temperatureC = (voltage - 0.5) * 100 ;  
  Serial.print(temperatureC); Serial.println(" degrees C");  
  
  delay(1000);  
}
```

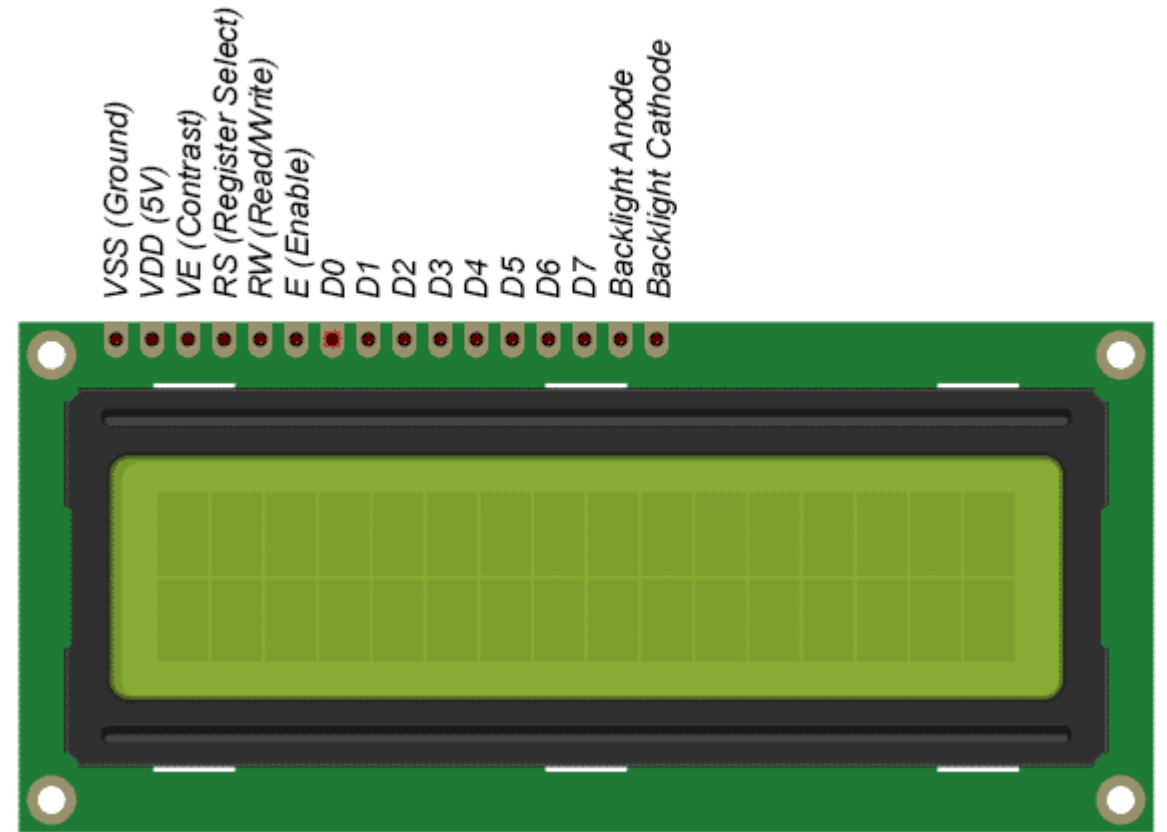
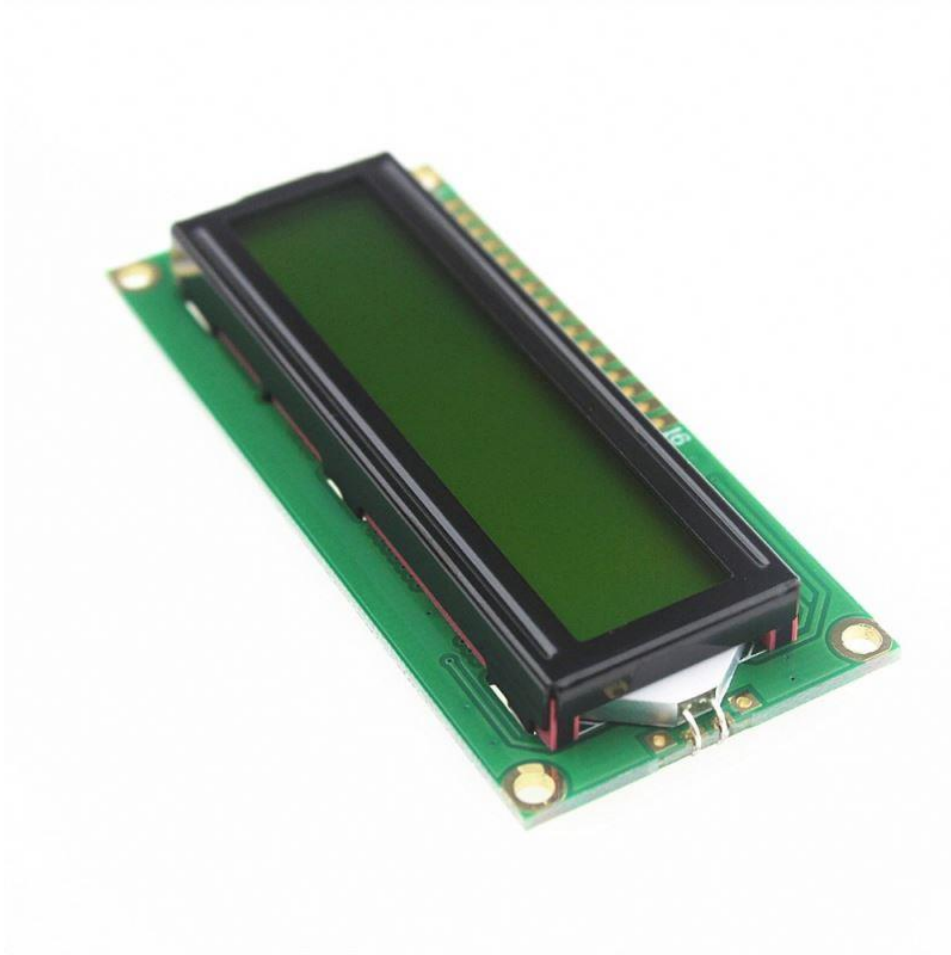
The screenshot displays an Arduino IDE interface. At the top, a blue header bar reads '온도 센서 [TMP36]'. Below it, a text box contains '이름 온도센서'. The main workspace shows a breadboard circuit with a TMP36 temperature sensor. The sensor's VCC pin is connected to a red wire leading to a 5V pin on the breadboard. Its GND pin is connected to a green wire leading to a GND pin. Its AO pin is connected to a black wire leading to an analog input pin (A0) on the breadboard. A potentiometer is also connected to the breadboard. On the right, the code editor shows the following code:

```
1 void setup()  
2 {  
3   Serial.begin(9600);  
4 }  
5  
6 void loop()  
7 {  
8   int reading = analogRead(A0);  
9  
10  float voltage = reading * 5.0;  
11  voltage /= 1024.0;  
12  
13  Serial.print(voltage); Serial.println(" volts");  
14  
15  float temperatureC = (voltage - 0.5) * 100 ;  
16  Serial.print(temperatureC); Serial.println(" degrees C");  
17  
18  float temperatureF = (temperatureC * 9.0 / 5.0) + 32.0;  
19  Serial.print(temperatureF); Serial.println(" degrees F");  
20  
21  delay(1000);  
22 }
```

Below the code editor, the '시리얼 모니터' (Serial Monitor) window is open, showing the following output:

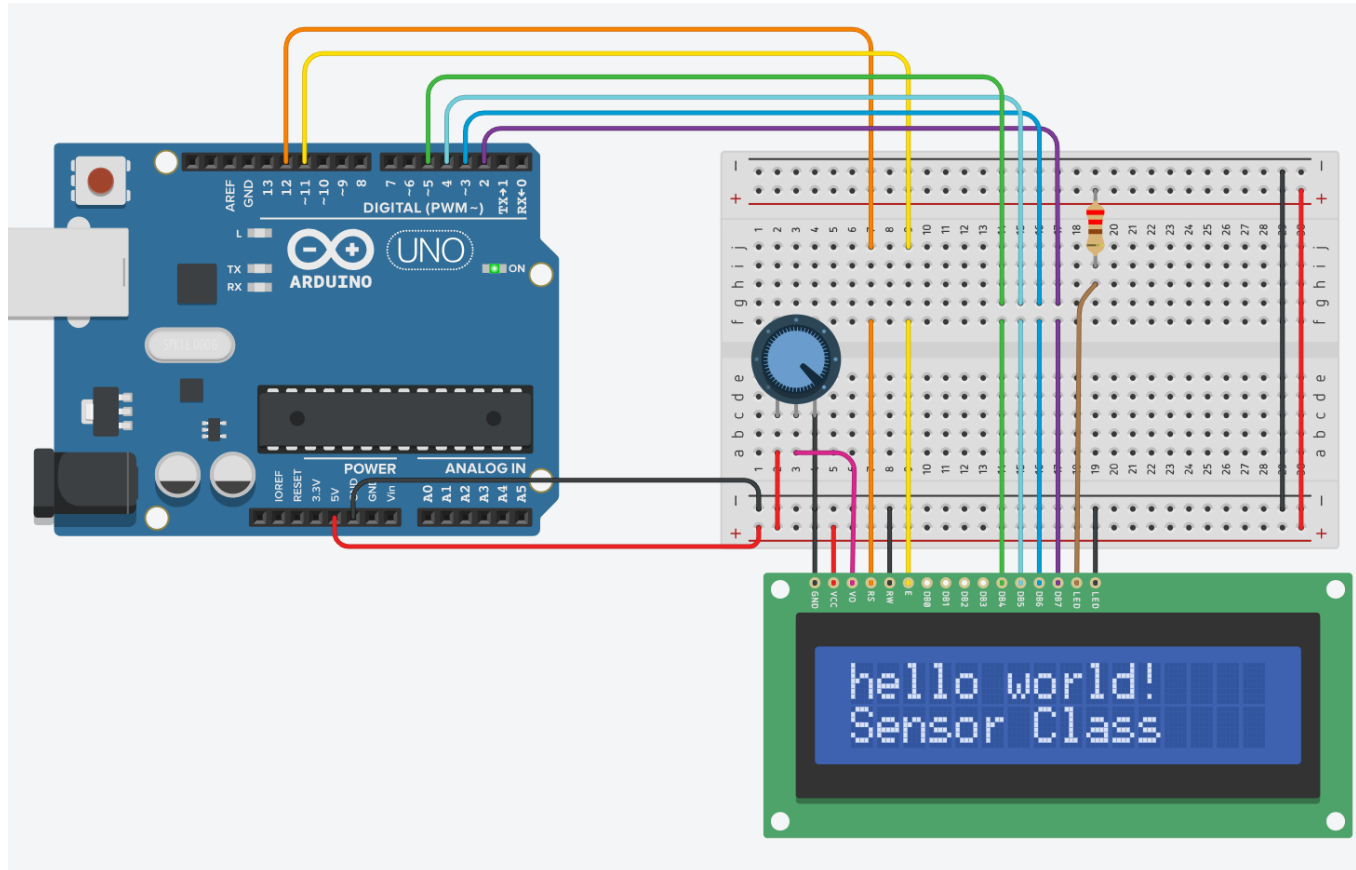
```
0.00 volts  
33.01 degrees C  
91.41 degrees F  
0.83 volts  
33.01 degrees C  
91.41 degrees F  
0.83 volts  
33.01 degrees C  
91.41 degrees F  
0.83 volts  
33.01 degrees C  
91.41 degrees F  
0.83 volts  
33.01 degrees C  
91.41 degrees F
```

# 16x2 Character LCD



# 16x2 Character LCD 실험

- 아두이노 Example을 이용



```
#include <LiquidCrystal.h>
```

```
LiquidCrystal lcd_1(12, 11, 5, 4, 3, 2);
```

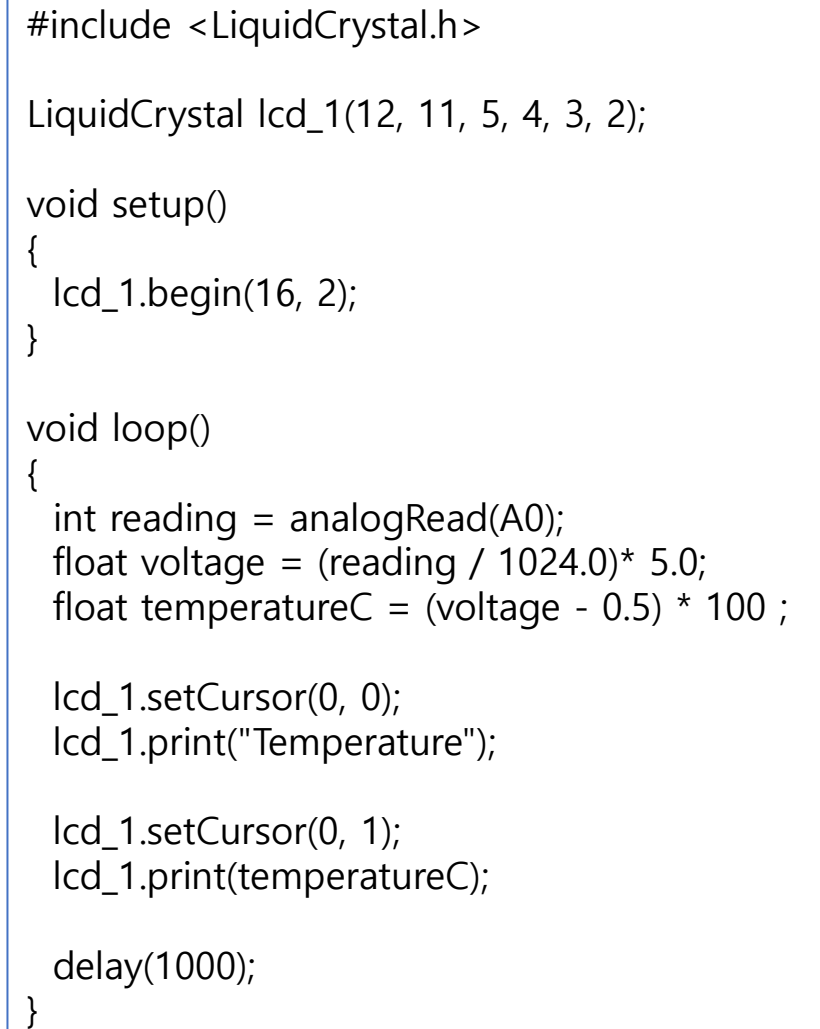
```
void setup()
```

```
{  
  lcd_1.begin(16, 2);  
}
```

```
void loop()
```

```
{  
  lcd_1.setCursor(0, 0);  
  lcd_1.print("hello world!");  
  
  lcd_1.setCursor(0, 1);  
  lcd_1.print("Sensor Class");  
  delay(1000);  
}
```

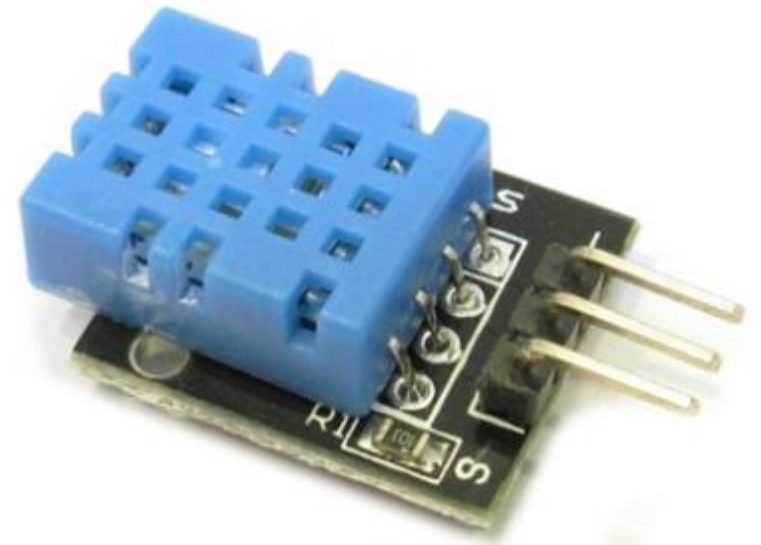
• LCD에 현재 온도(TMP36)를 표시하시오





# DHT11

- 동작 전압 (Power) 3~5 V
- 온도 측정 범위 (Temperature range) 0 ~ 50 °C ( $\pm 2$  °C)
- 습도 측정 범위 (Humidity range) 20 ~ 80 % ( $\pm 5$  %)
- 최대소비전력 (Max. current) 2.5 mA
- 데이터 주기 (sampling rate) 1 Hz



# DHT11 라이브러리 사용

- <https://github.com/adafruit/DHT-sensor-library>

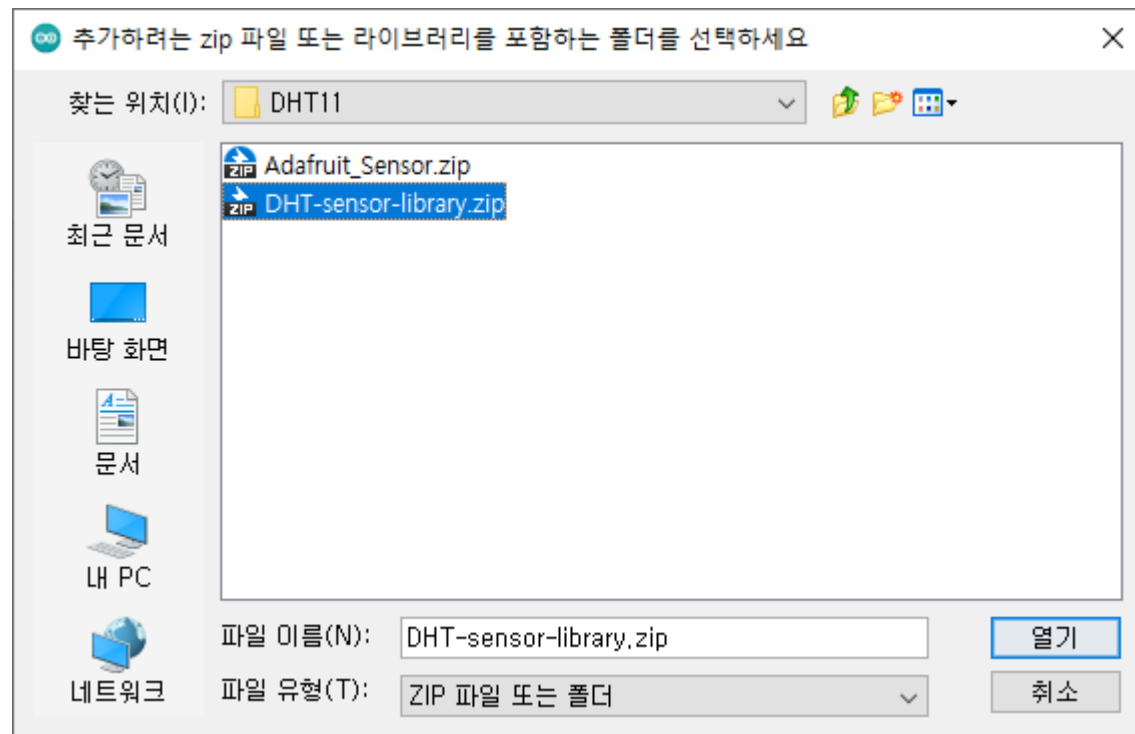
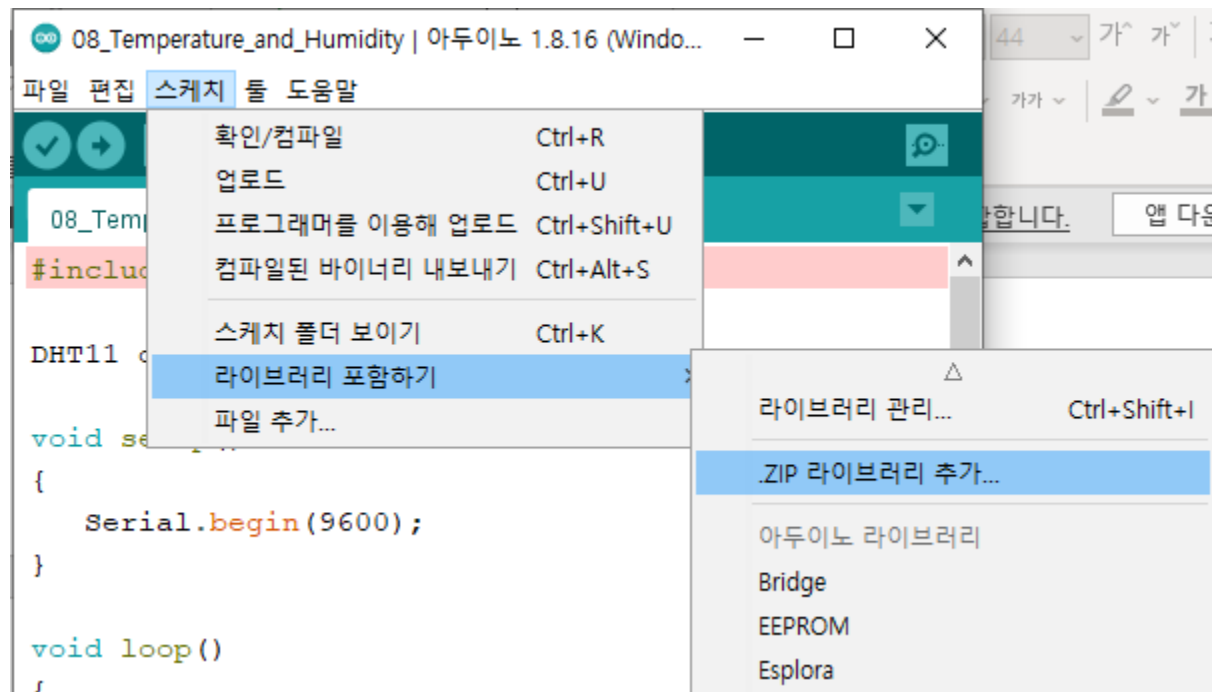
The screenshot shows the GitHub repository page for `adafruit/DHT-sensor-library`. The repository is public and has 162 watchers, 1.6k stars, and 1.2k forks. It contains 17 issues, 13 pull requests, and 142 commits. The latest commit is by `dherrada` bumping the version to 1.4.3, 29 days ago.

The file list includes:

- `.github`: actionified, formatted and doxy'd (2 years ago)
- `examples`: Updated comment on the Pin Out of the DHTxx Sensors (9 months ago)
- `.gitignore`: Add .gitignore (2 years ago)
- `CONTRIBUTING.md`: [Update URL] (2 years ago)
- `DHT.cpp`: Merge pull request #159 from Rotzbua/patch-1 (2 years ago)
- `DHT.h`: Fix comment on DHT22 and DHT21 variables (29 days ago)
- `DHT_U.cpp`: actionified, formatted and doxy'd (2 years ago)
- `DHT_U.h`: actionified, formatted and doxy'd (2 years ago)
- `README.md`: actionified, formatted and doxy'd (2 years ago)
- `code-of-conduct.md`: actionified, formatted and doxy'd (2 years ago)
- `keywords.txt`: Use correct field separator in keywords.txt (3 years ago)
- `library.properties`: Bump to 1.4.3 (29 days ago)

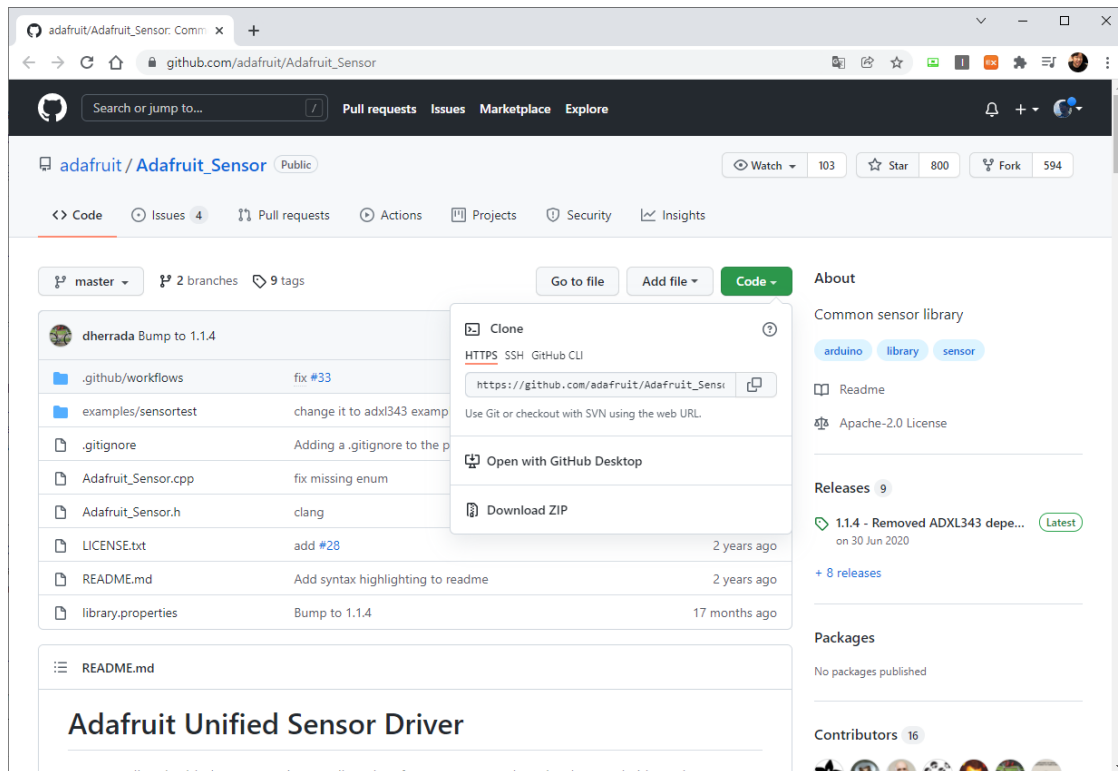
The right sidebar shows the `About` section with the description: "Arduino library for DHT11, DHT22, etc Temperature & Humidity Sensors". It also includes links to `learn.adafruit.com/dht`, `Readme`, `MIT License`, and `Code of conduct`. The `Releases` section shows 22 releases, with the latest being `1.4.3 - Switched from defines to ...` (29 days ago). The `Packages` section shows "No packages published".

# DHT11 라이브러리 사용

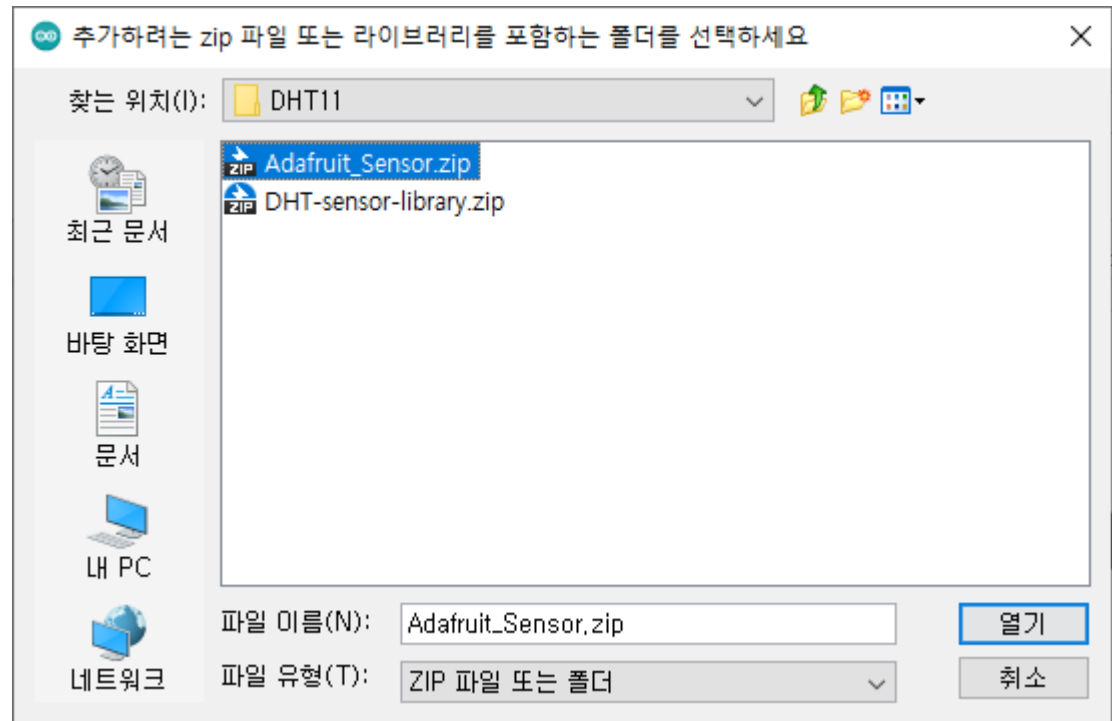
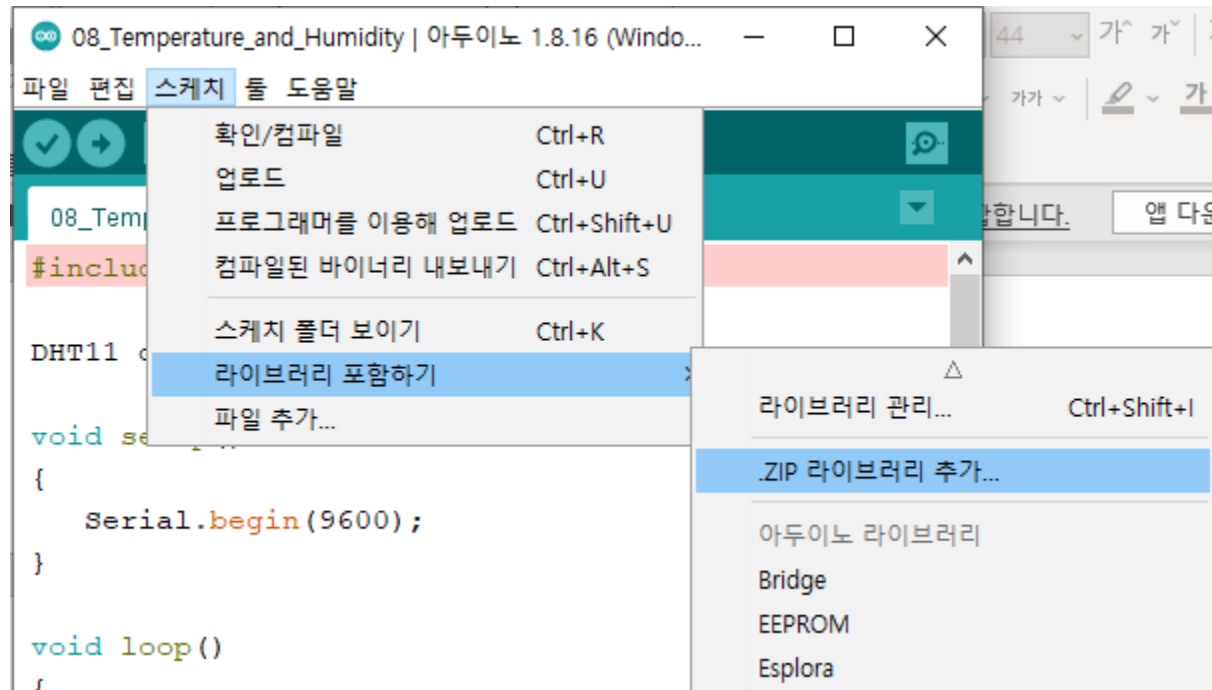


# DHT11 라이브러리 사용

- adafruit\_sensor.h no such file 에러 발생
- [https://github.com/adafruit/Adafruit\\_Sensor](https://github.com/adafruit/Adafruit_Sensor) 라이브러리 추가

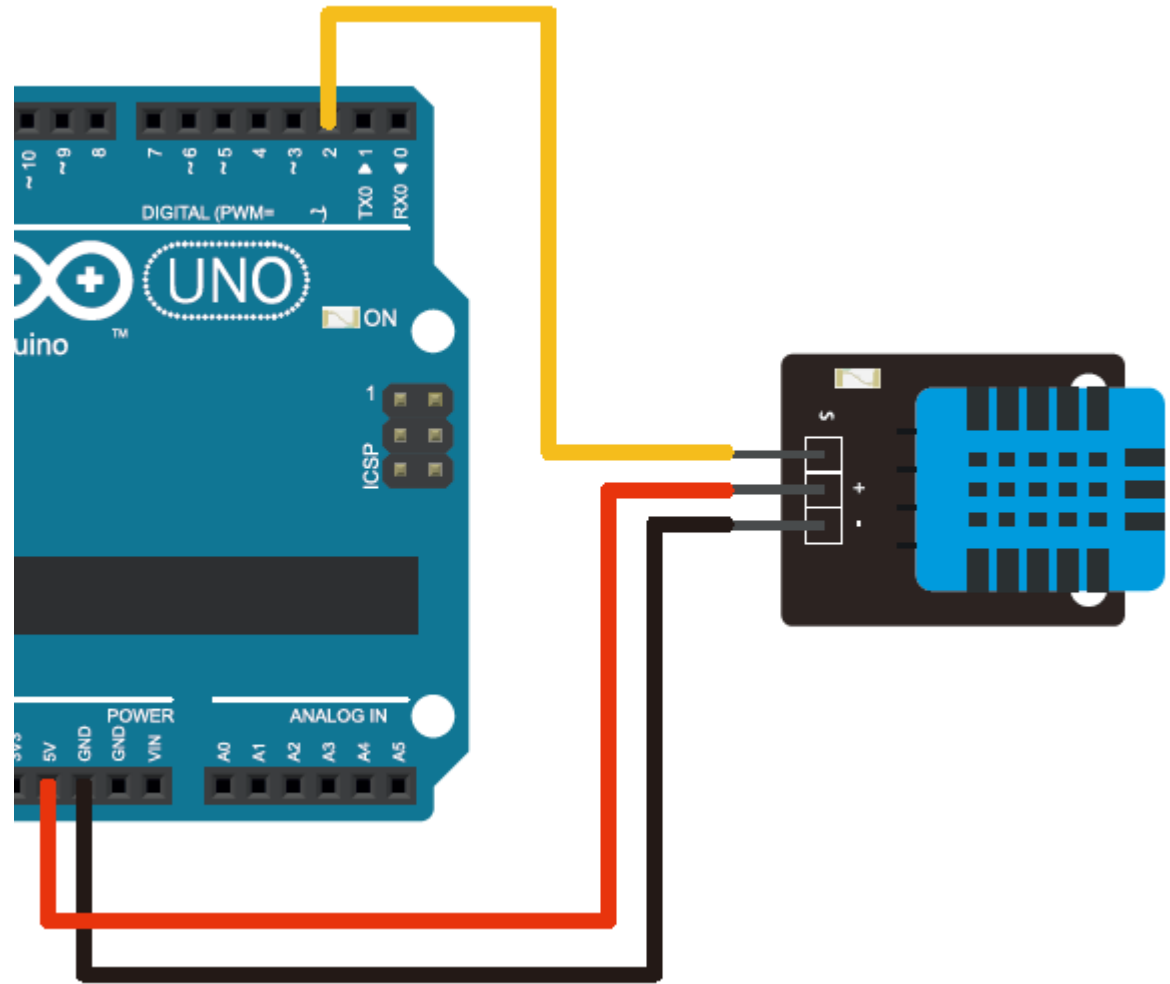


# DHT11 라이브러리 사용

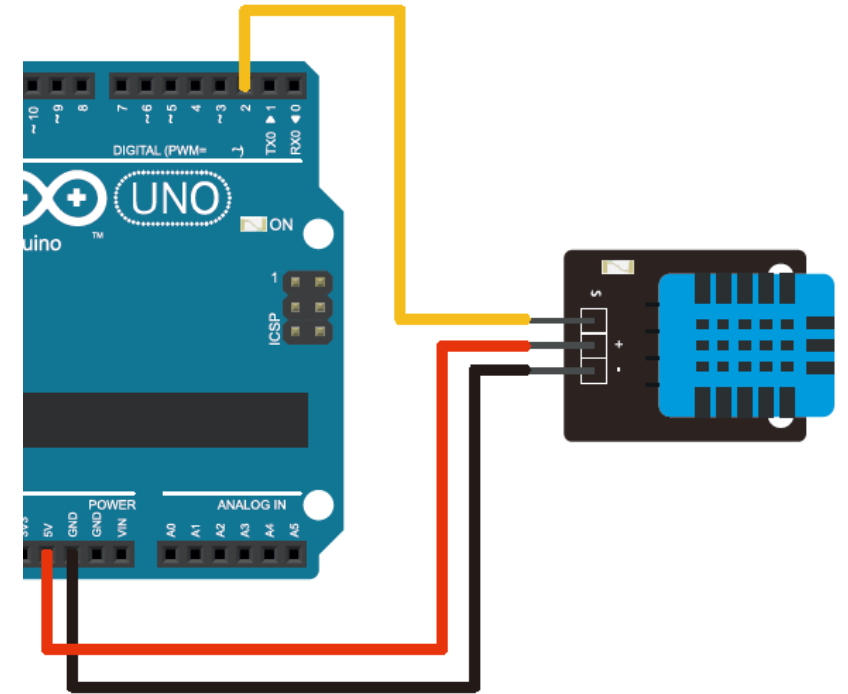
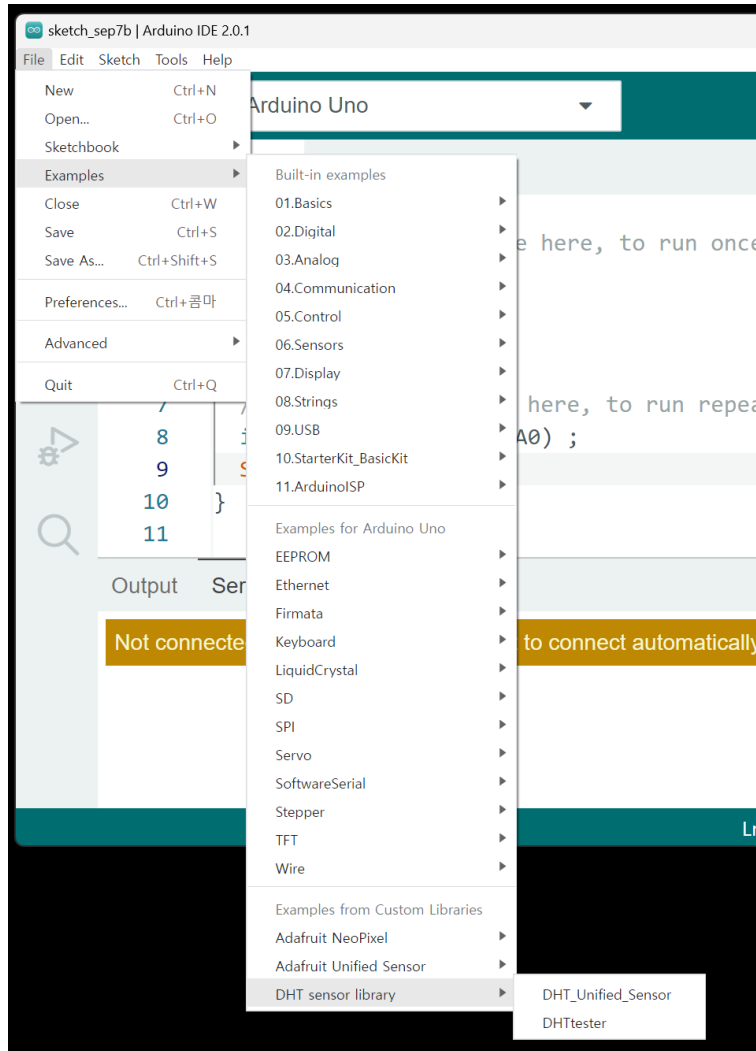


# DHT11 아두이노 테스트

- S(signal) : 아두이노 2번핀
- + : VCC(5V)
- - : GND(0V)



# DHT11 아두이노 테스트(Example 코드 활용)



# QUIZ

- LCD에 현재 온도와 습도를 표시하시오



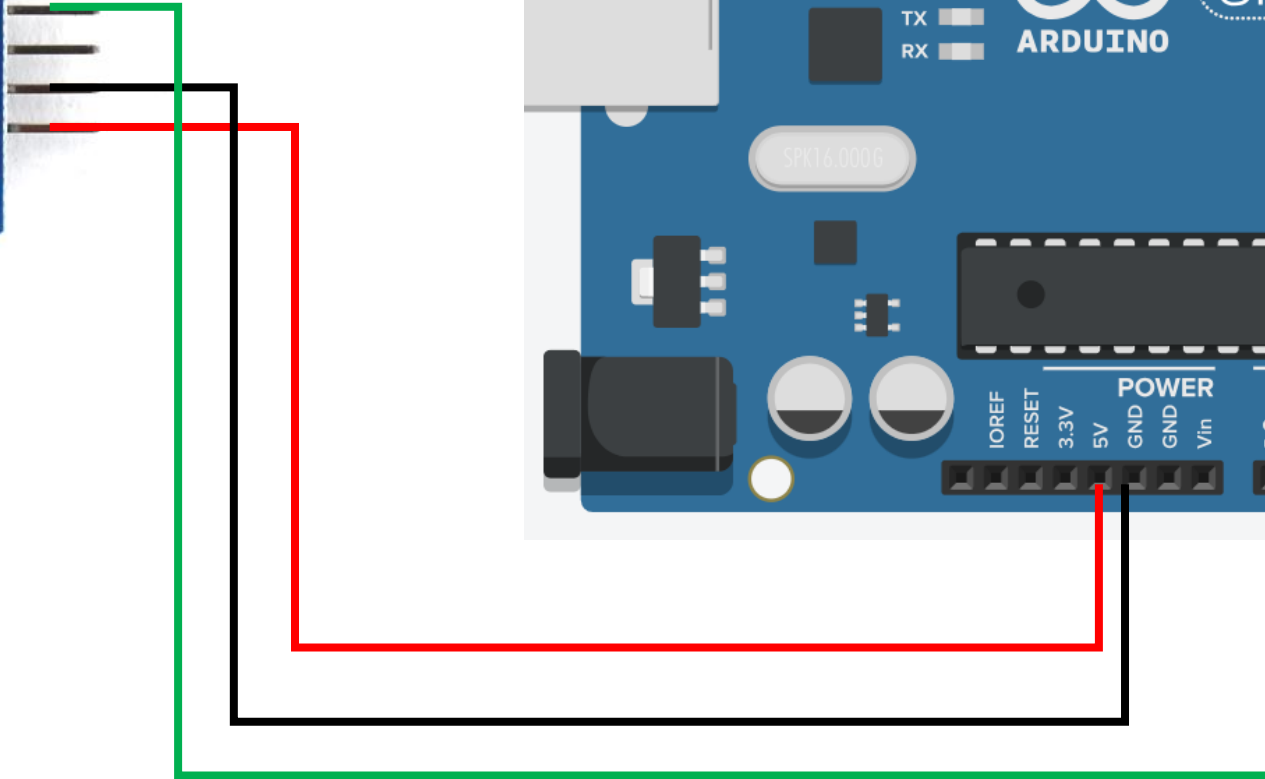
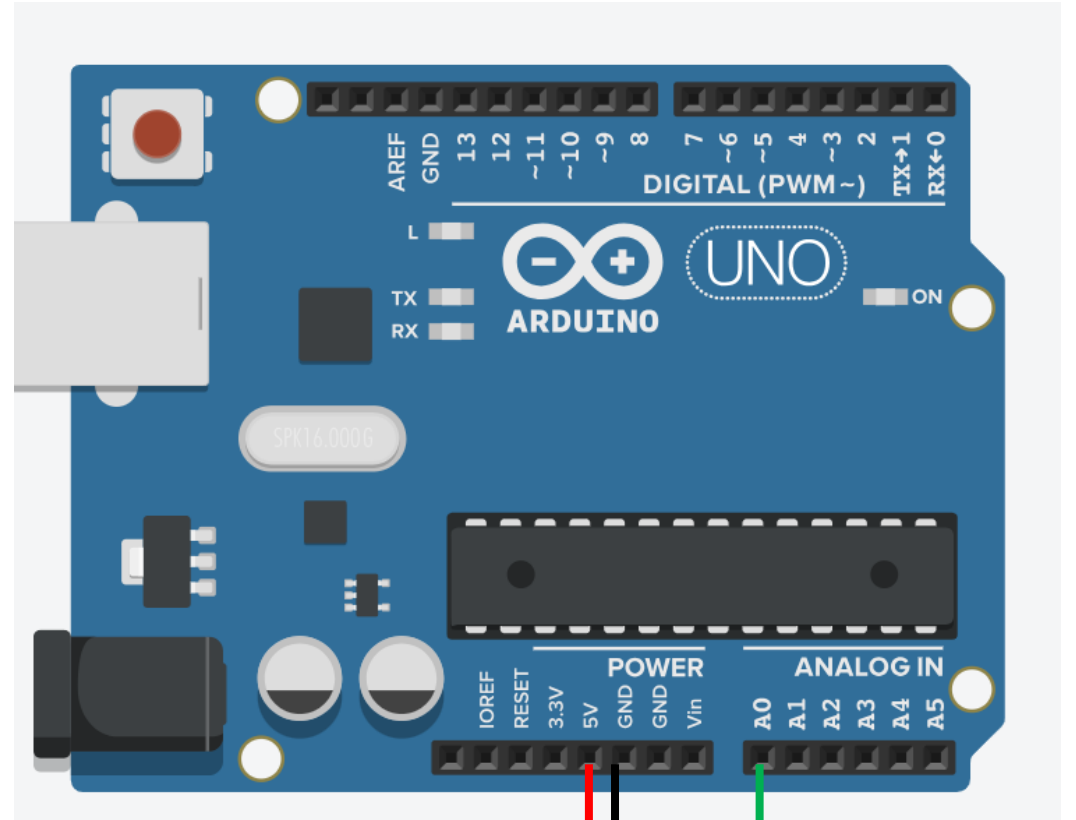


# 음주 측정기 만들기

- 알코올 센서



# 음주 측정기 만들기



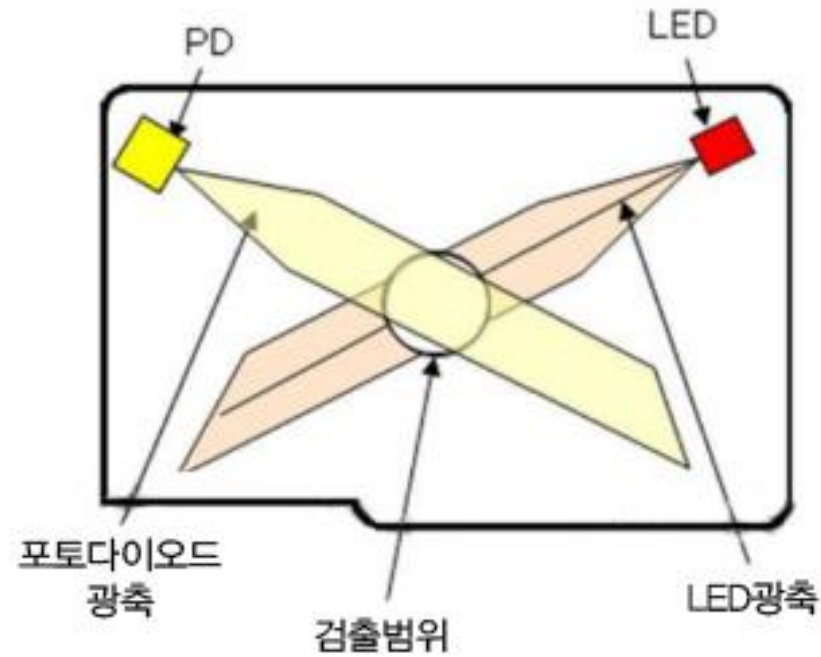
# QUIZ

- LCD에 현재 알코올 레벨을 표시하고 300이상인 경우 부저를 이용하여 소리를 출력한다.



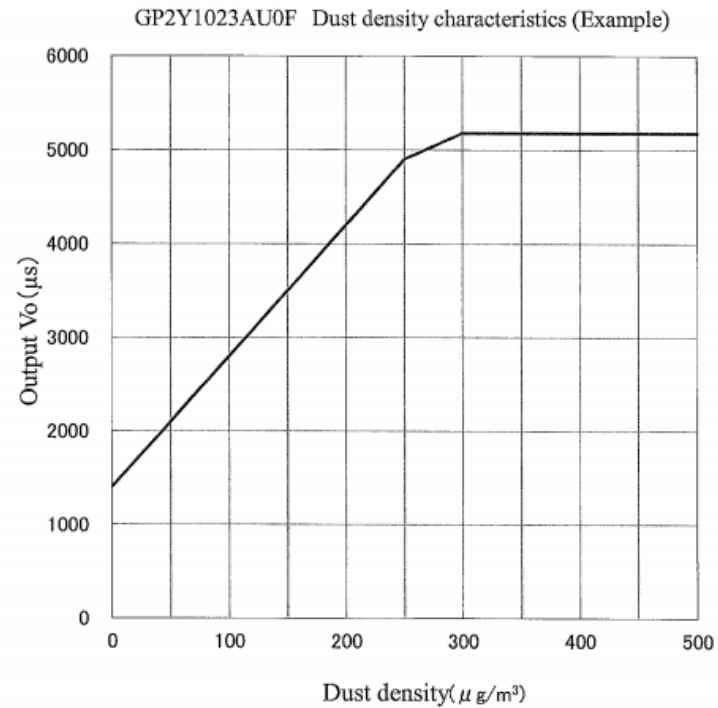
# PM2.5 GP2Y1023AU0F 먼지센서

- 빛을 이용하여 먼지가 얼마나 많은지를 측정

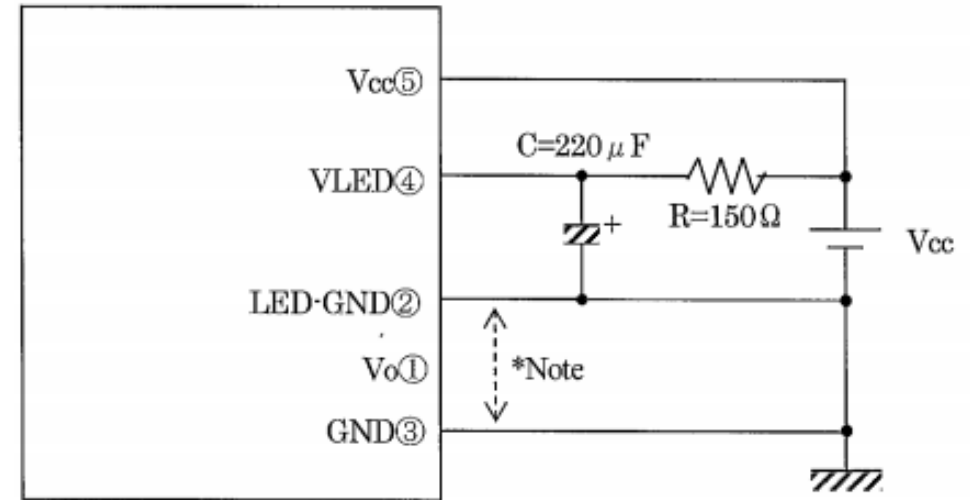
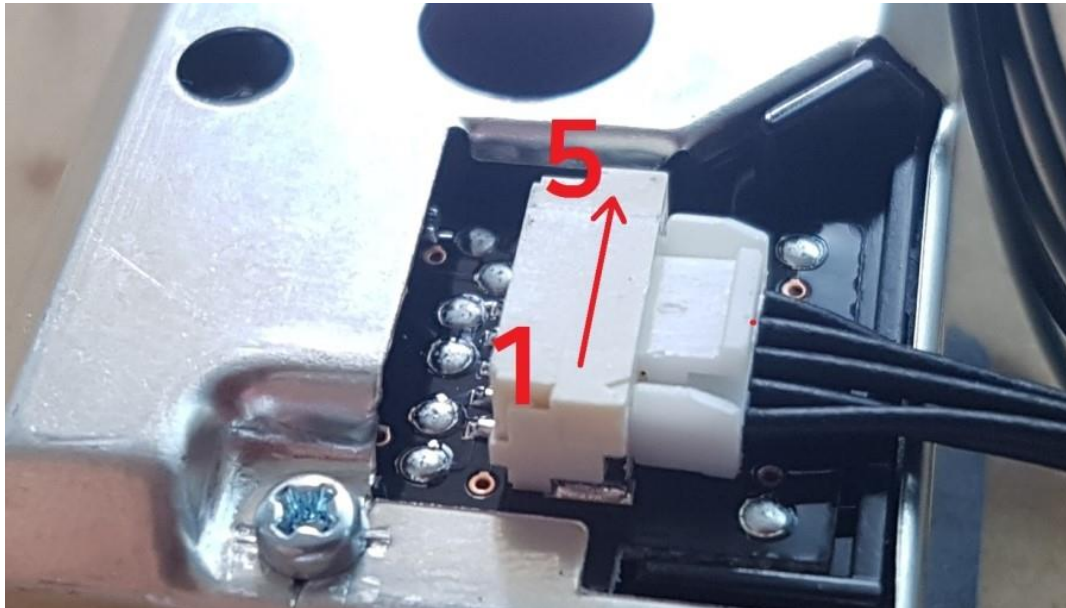


# PM2.5 GP2Y1023AU0F 먼지 센서

- 빛을 이용하여 먼지가 얼마나 많은지를 측정



# 미세먼지 센서 회로 구성

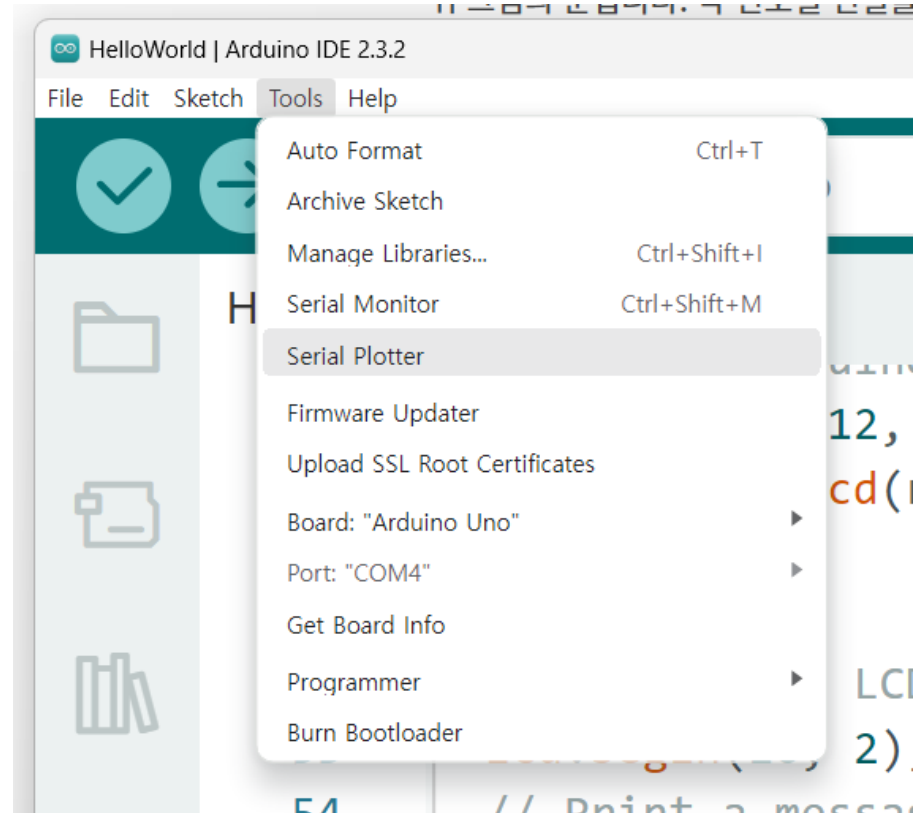


\*Note : LED-GND line and GND line must be separated.

# 미세먼지 센서의 데이터 확인

```
void setup()
{
  pinMode(8, INPUT);
  Serial.begin(9600);
}

void loop()
{
  int value = digitalRead(8) ;
  Serial.println(value);
}
```



# 미세먼지 센서의 데이터 확인

시리얼 플로터를 활용하여 PWM 신호 확인

