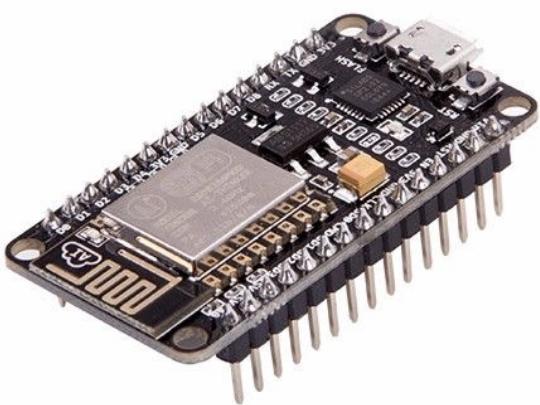
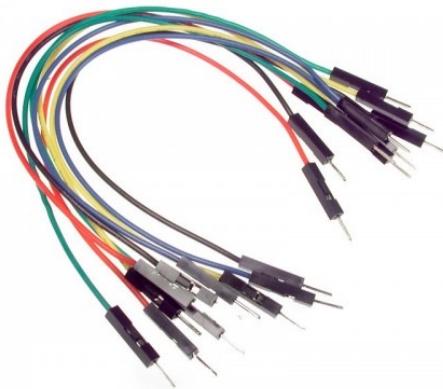


NodeMCU에서 온습도 센서(DHT1) 측정하기

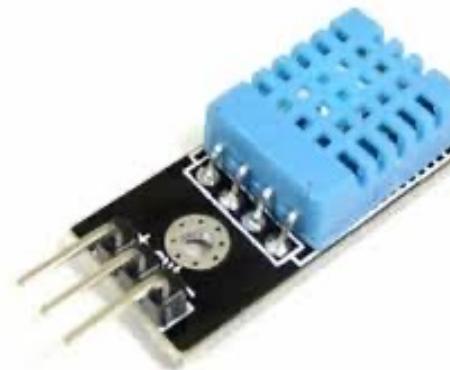
준비물



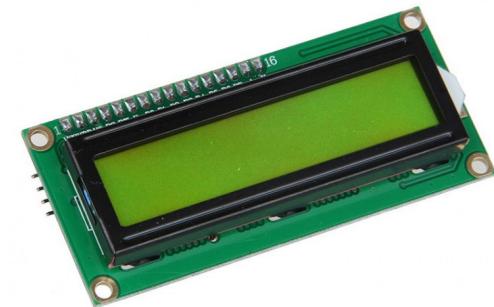
NodeMCU



점프 케이블

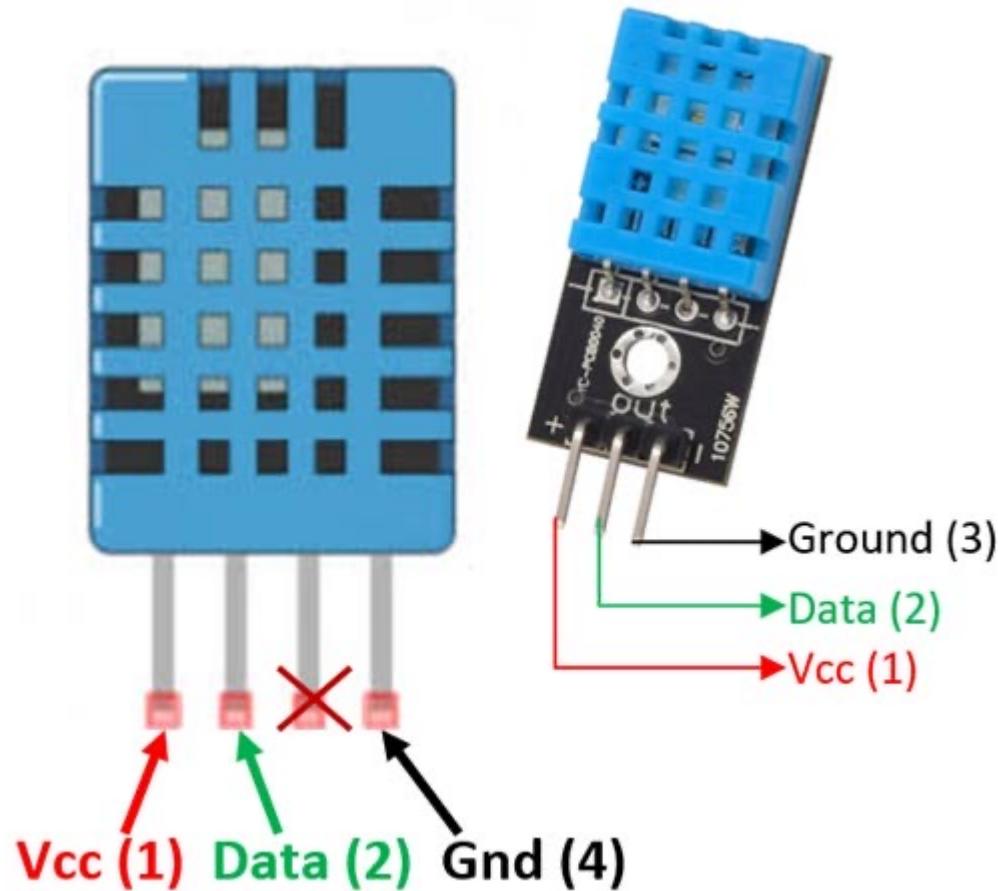


DHT11 온습도 센서

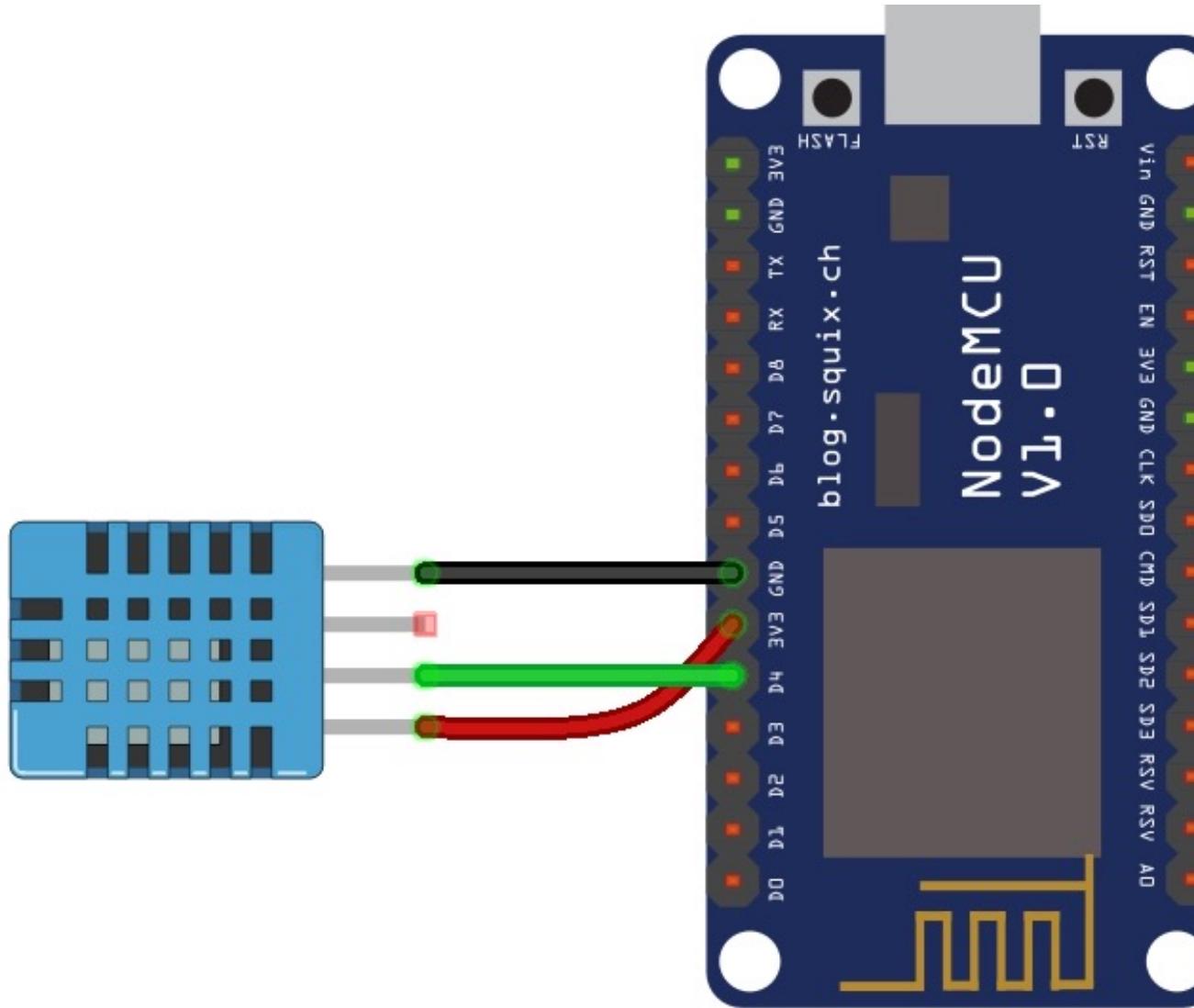


16x2 LCD 모듈

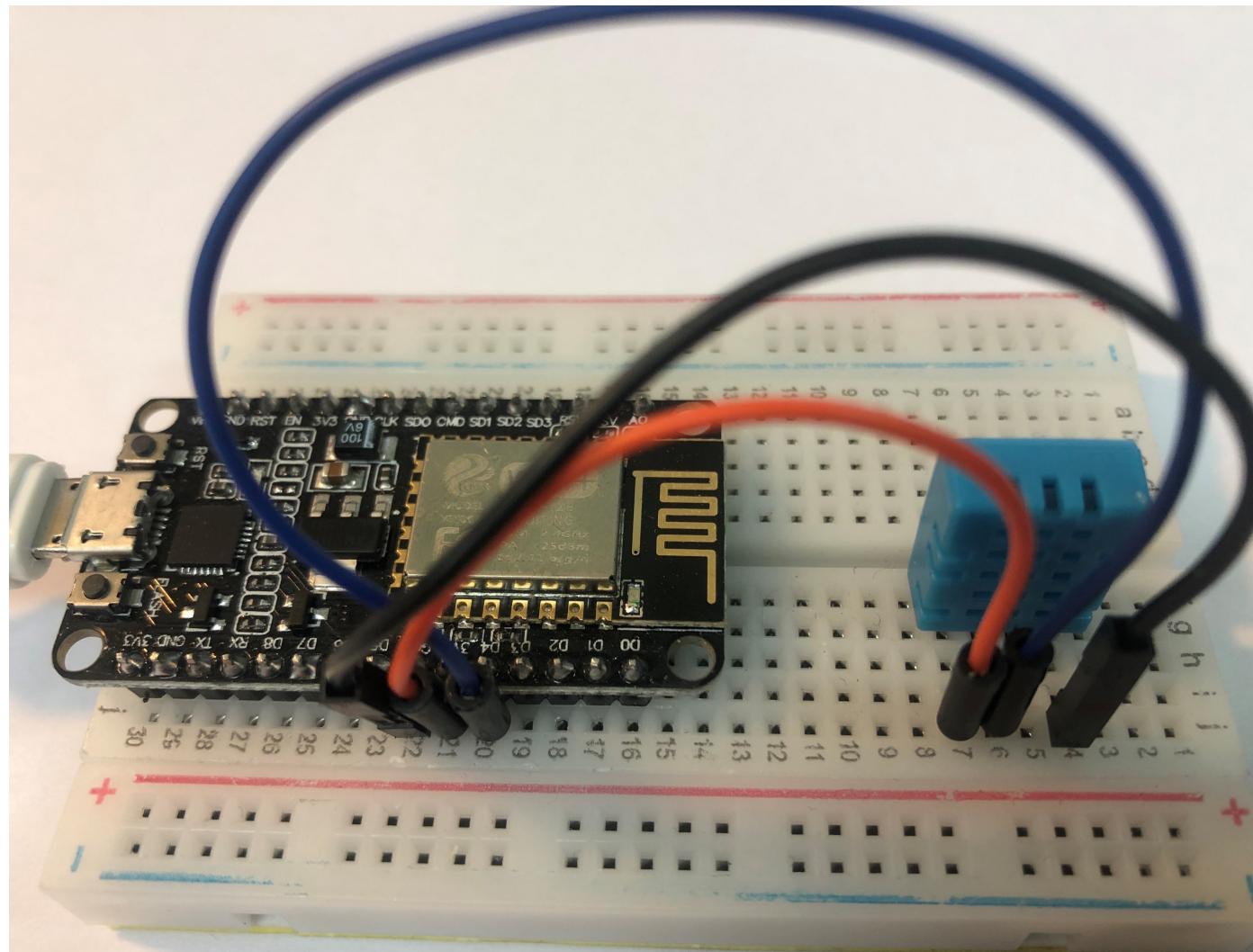
DHT11 온습도 센서 Pinout



Fritzing 회로 제작

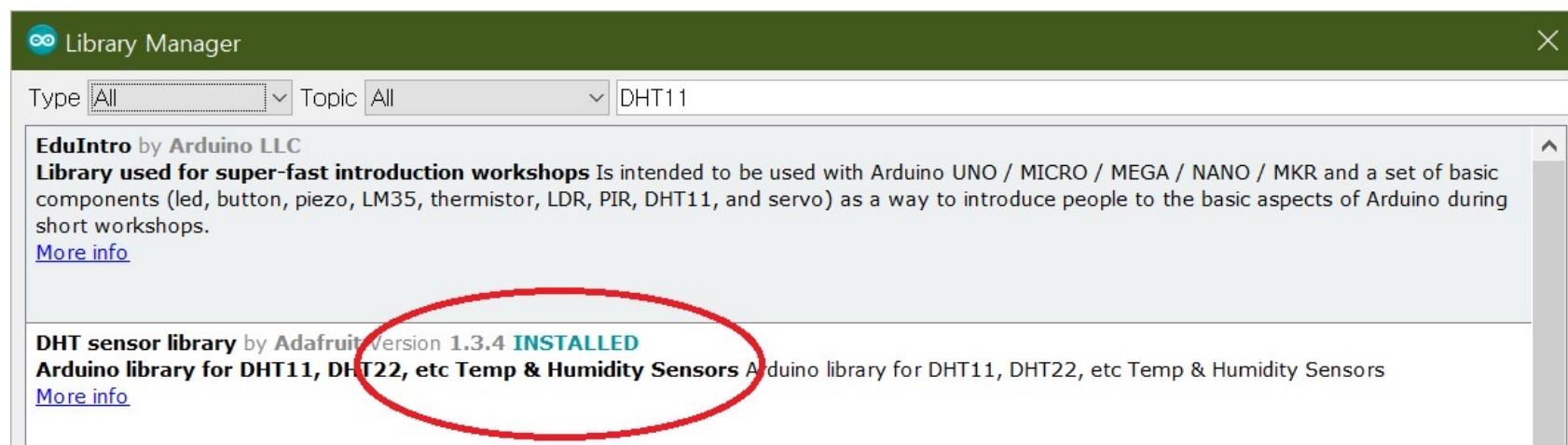
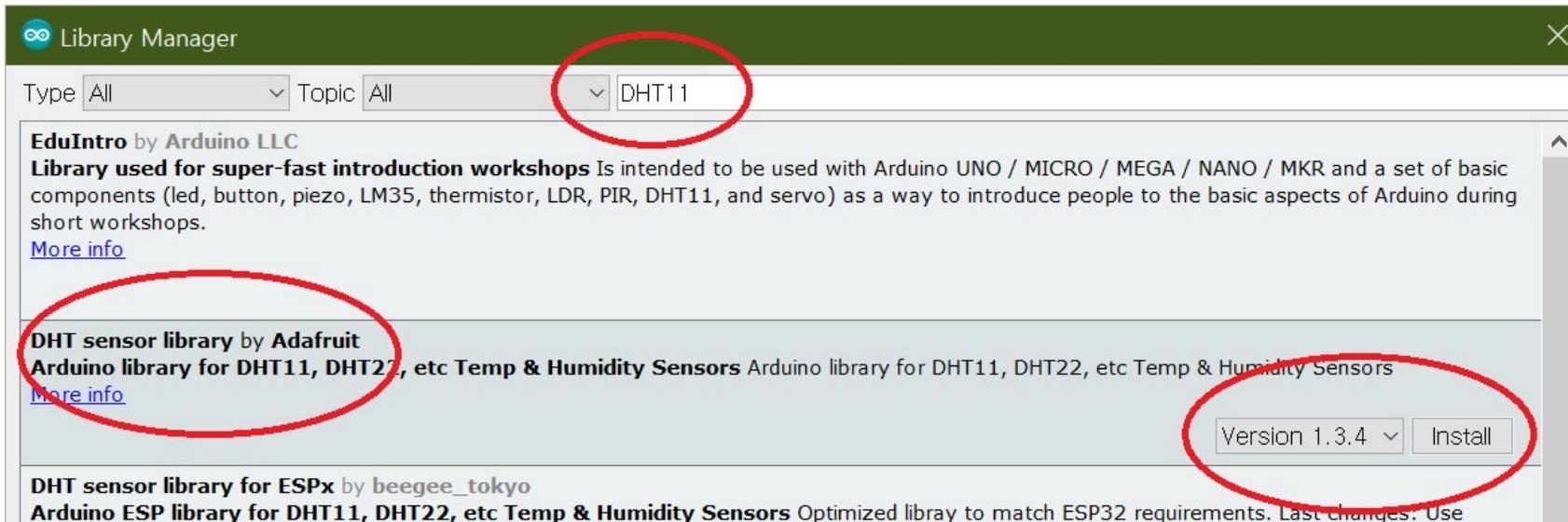


실제 회로 구성



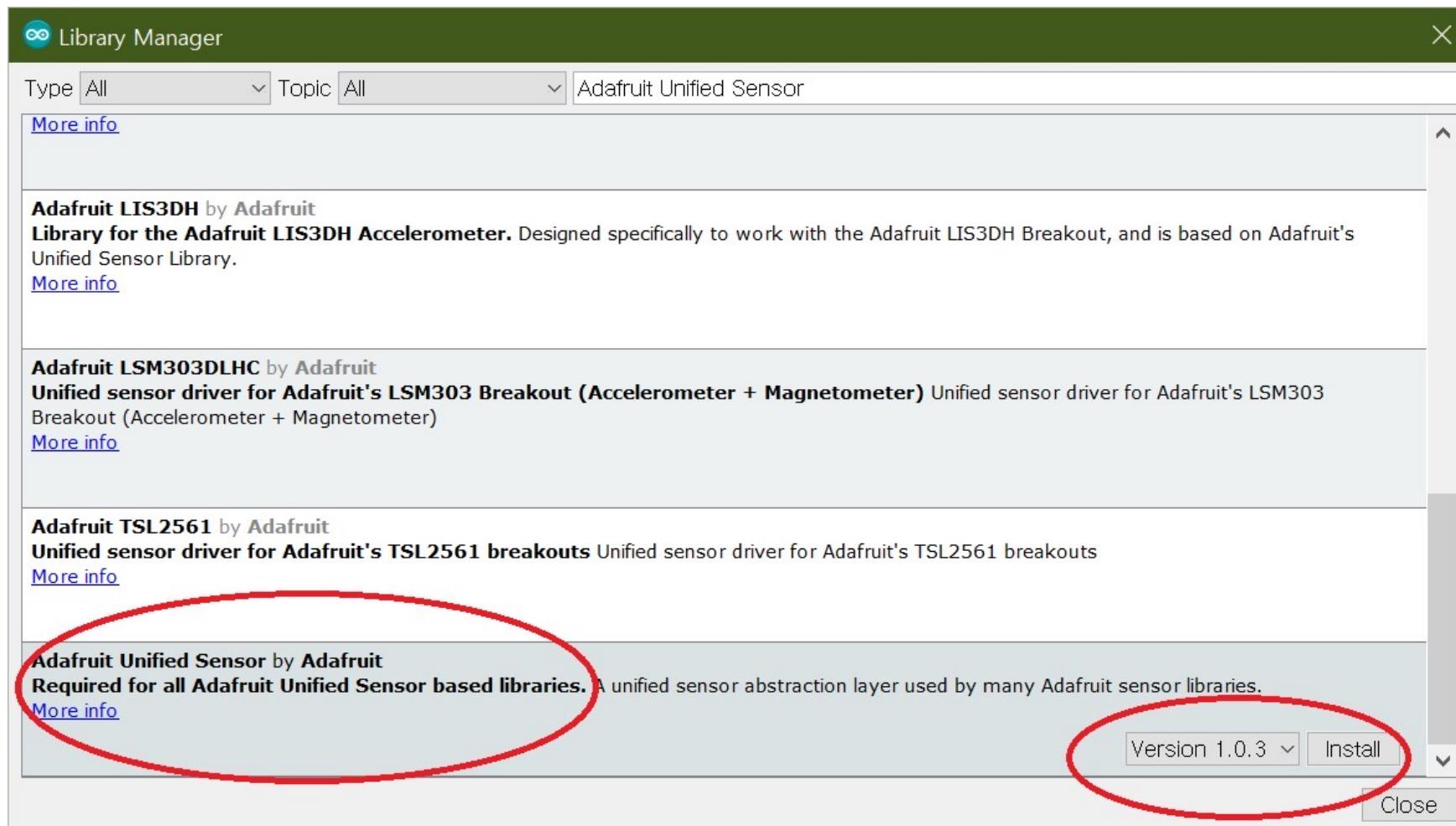
온습도센서 DHT Sensor Library 설치(1)

- 상단 메뉴 -> 스케치 -> 라이브러리 포함하기 -> 라이브러리 관리

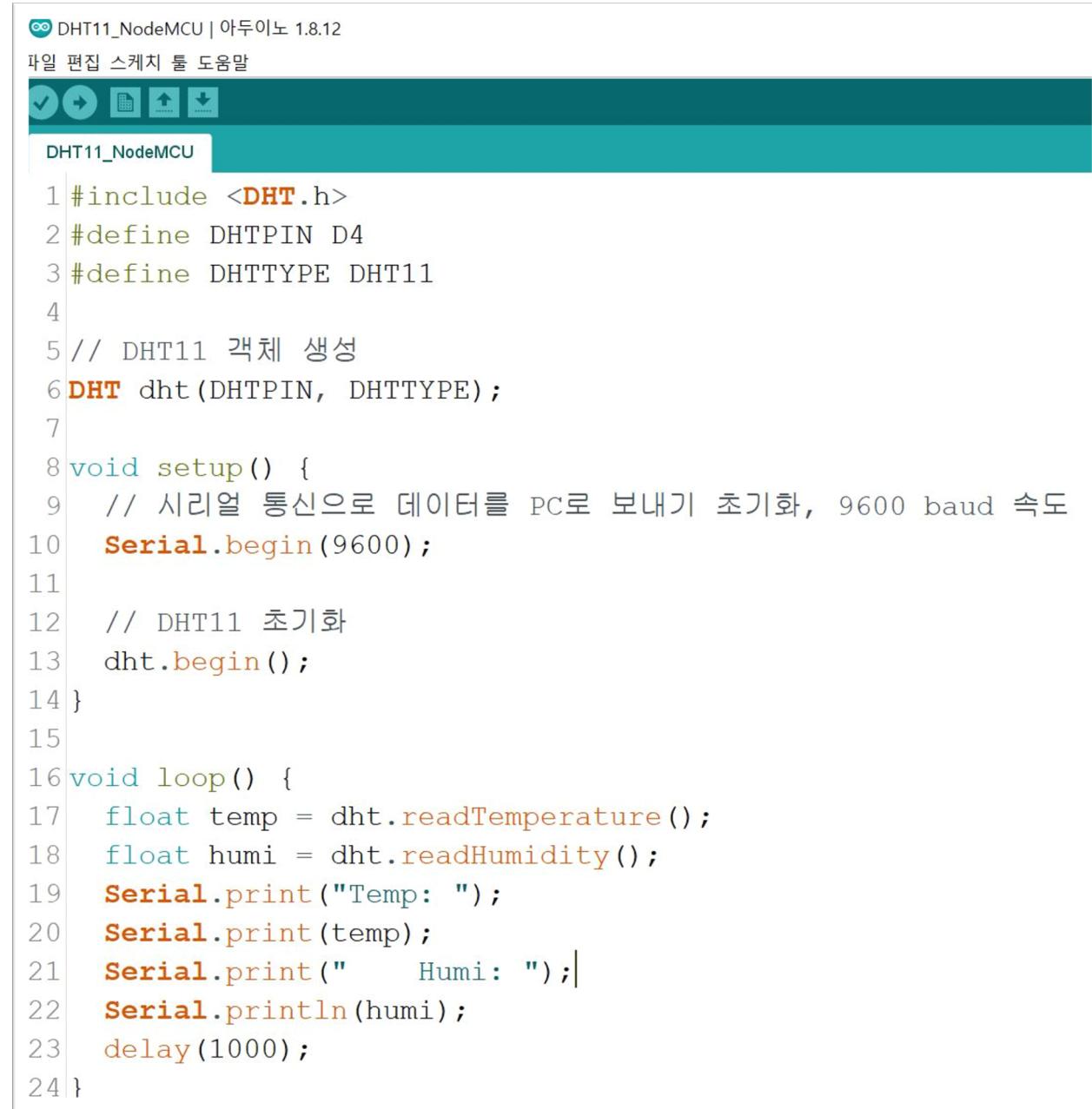


Adafruit Unified Sensor Library 설치(2)

- 상단 메뉴 -> 스케치 -> 라이브러리 포함하기 -> 라이브러리 관리



스케치 코드 작성



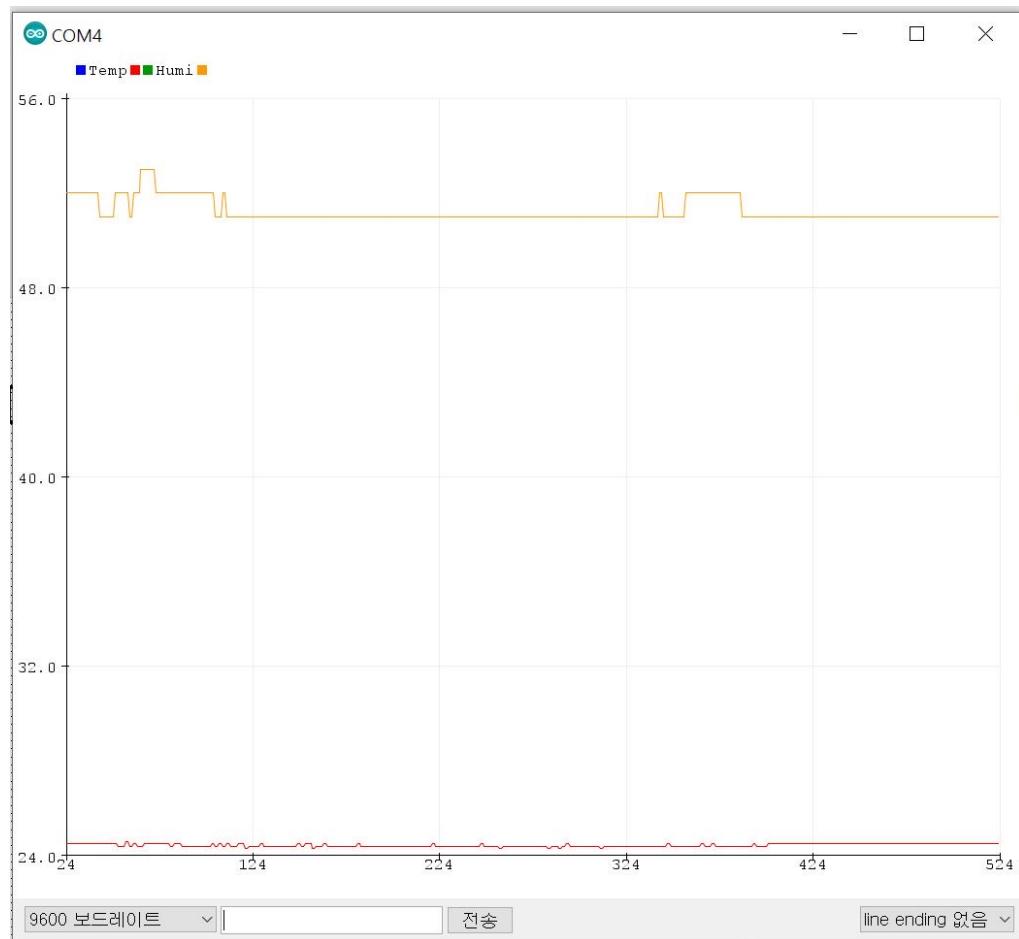
The screenshot shows the Arduino IDE interface with the following details:

- Top bar: DHT11_NodeMCU | 아두이노 1.8.12
- Menu bar: 파일 편집 스케치 툴 도움말
- Toolbar: Save, Load, Open, Upload, Download
- Sketch name: DHT11_NodeMCU
- Code area:

```
1 #include <DHT.h>
2 #define DHTPIN D4
3 #define DHTTYPE DHT11
4
5 // DHT11 객체 생성
6 DHT dht(DHTPIN, DHTTYPE);
7
8 void setup() {
9     //シリ얼 통신으로 데이터를 PC로 보내기 초기화, 9600 baud 속도
10    Serial.begin(9600);
11
12    // DHT11 초기화
13    dht.begin();
14 }
15
16 void loop() {
17     float temp = dht.readTemperature();
18     float humi = dht.readHumidity();
19     Serial.print("Temp: ");
20     Serial.print(temp);
21     Serial.print("    Humi: ");
22     Serial.println(humi);
23     delay(1000);
24 }
```

센서 데이터 출력

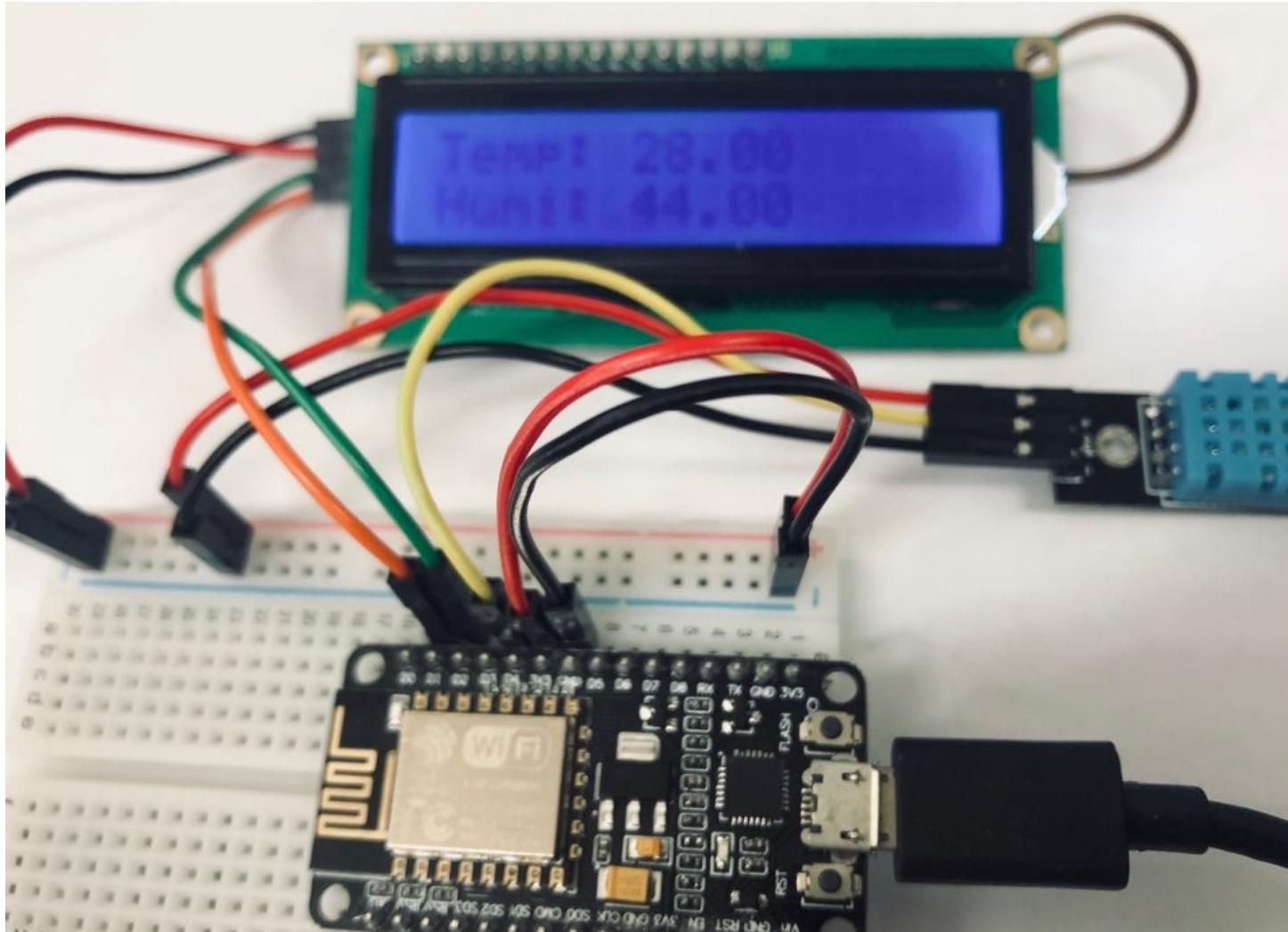
- 상단 메인 메뉴 -> 툴 > 시리얼 모니터/ 시리얼 플로터



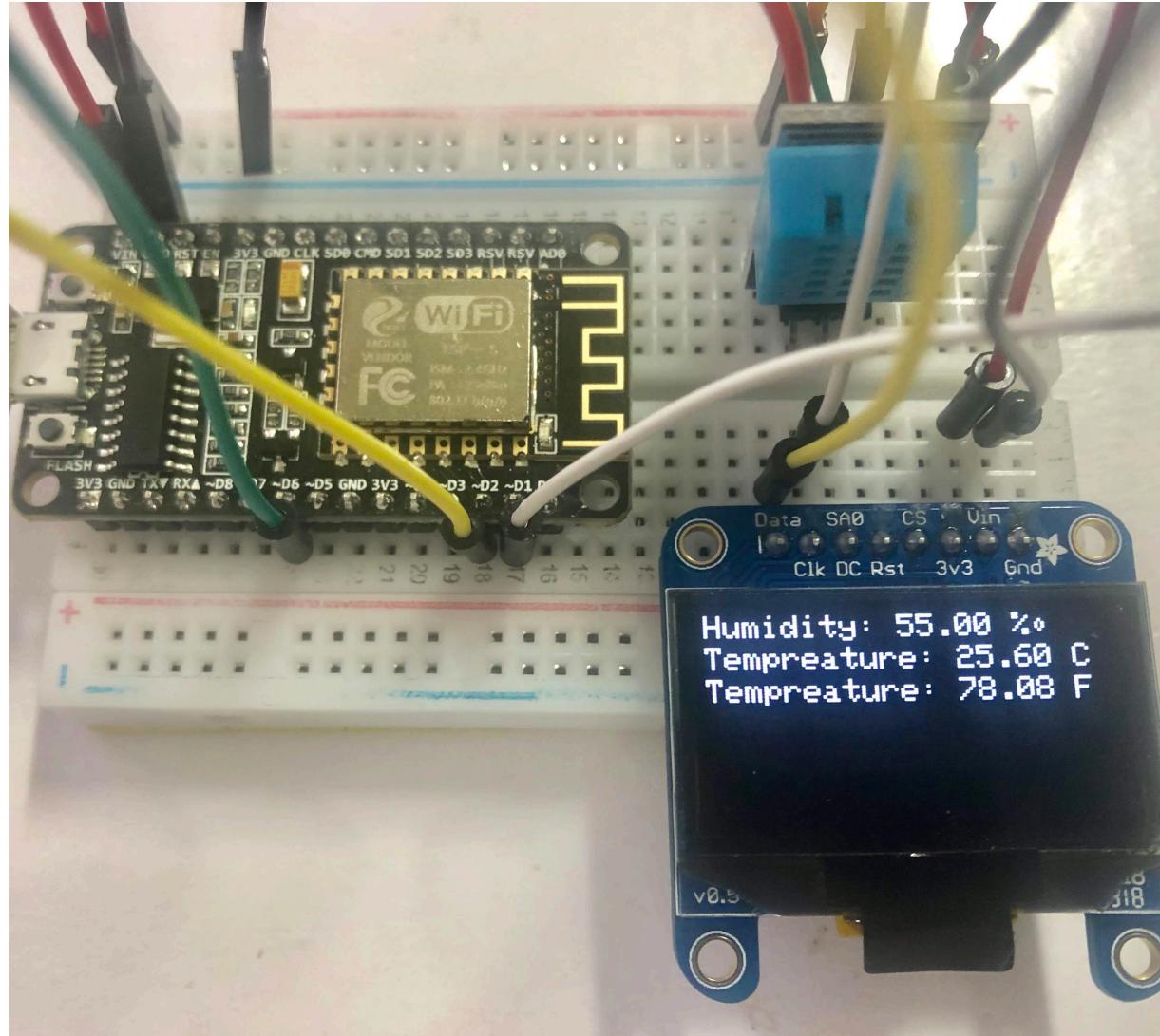
< 시리얼 플로터 >

< 시리얼 모니터 >

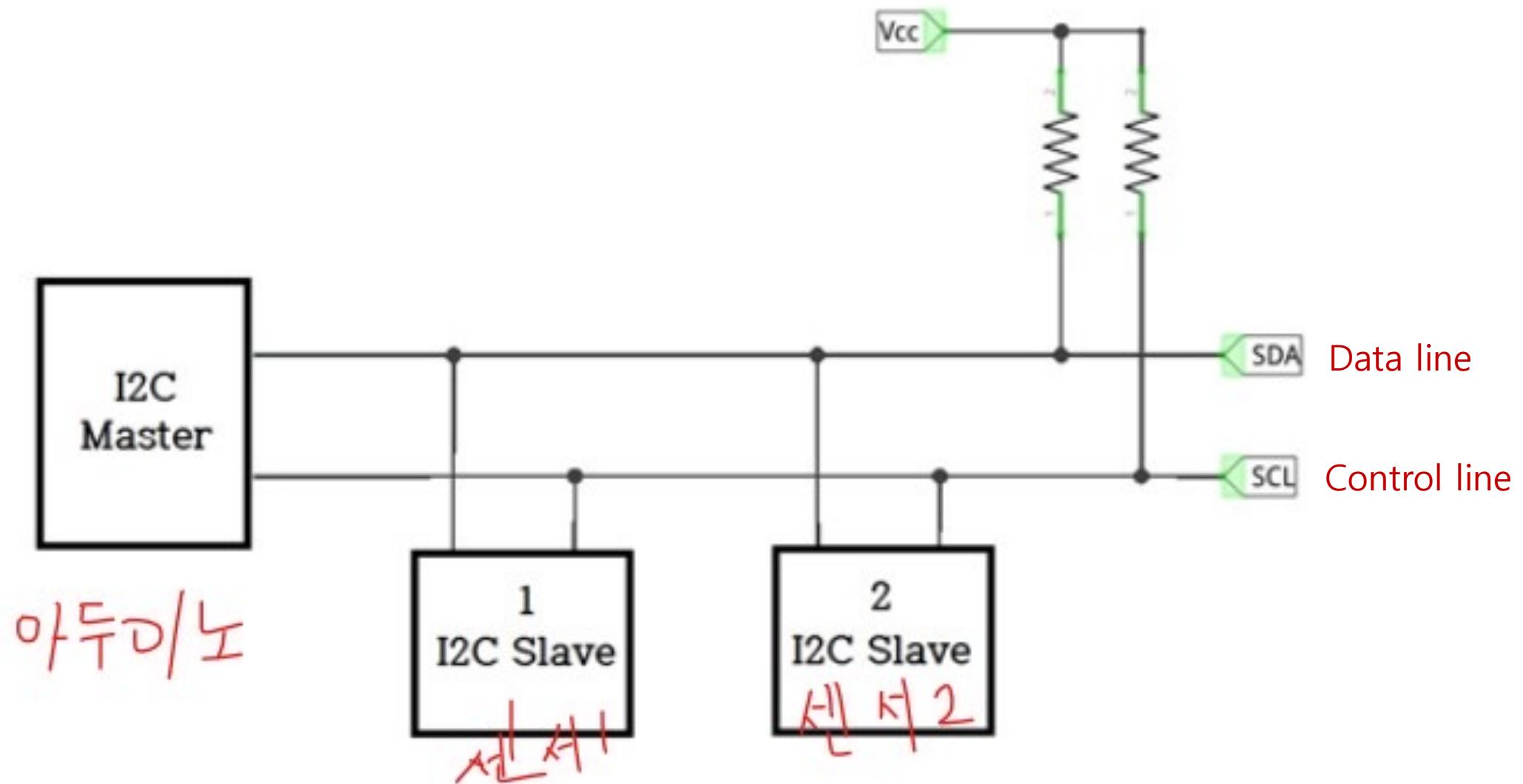
16x2 I2C LCD 디스플레이 출력



OLED 디스플레이

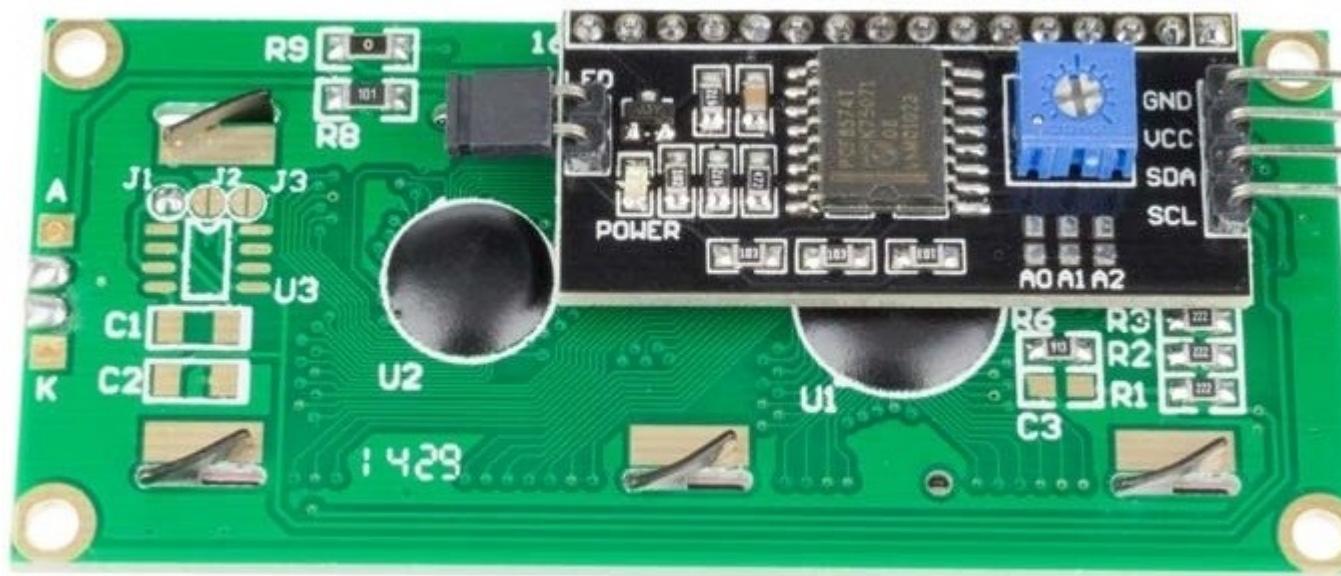


아두이노 I2C 통신

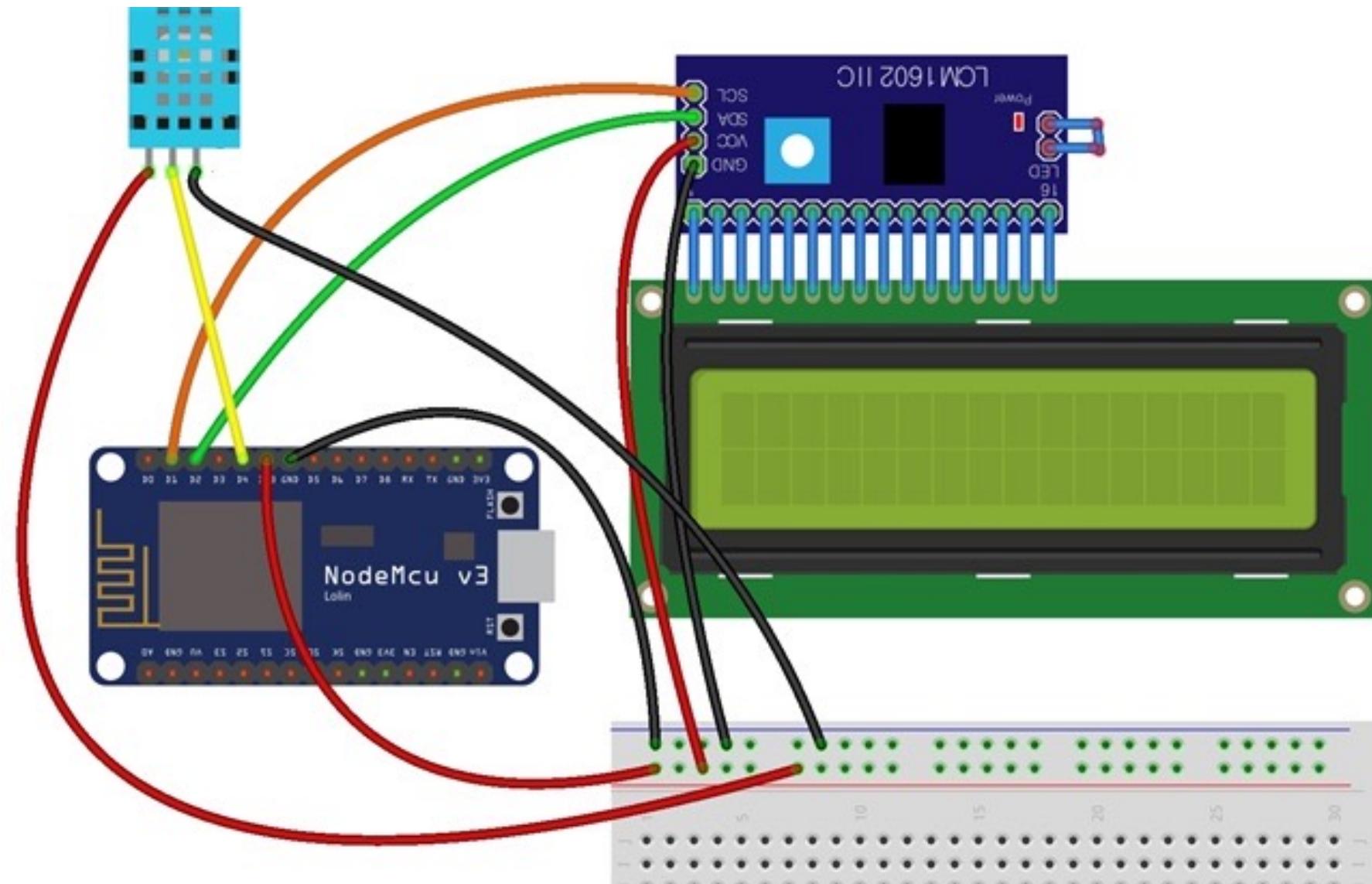


LCD I2C Pin 연결

LCD	NodeMCU
GND	GND
VCC	3.3V
SDA	D2
SCL	D1

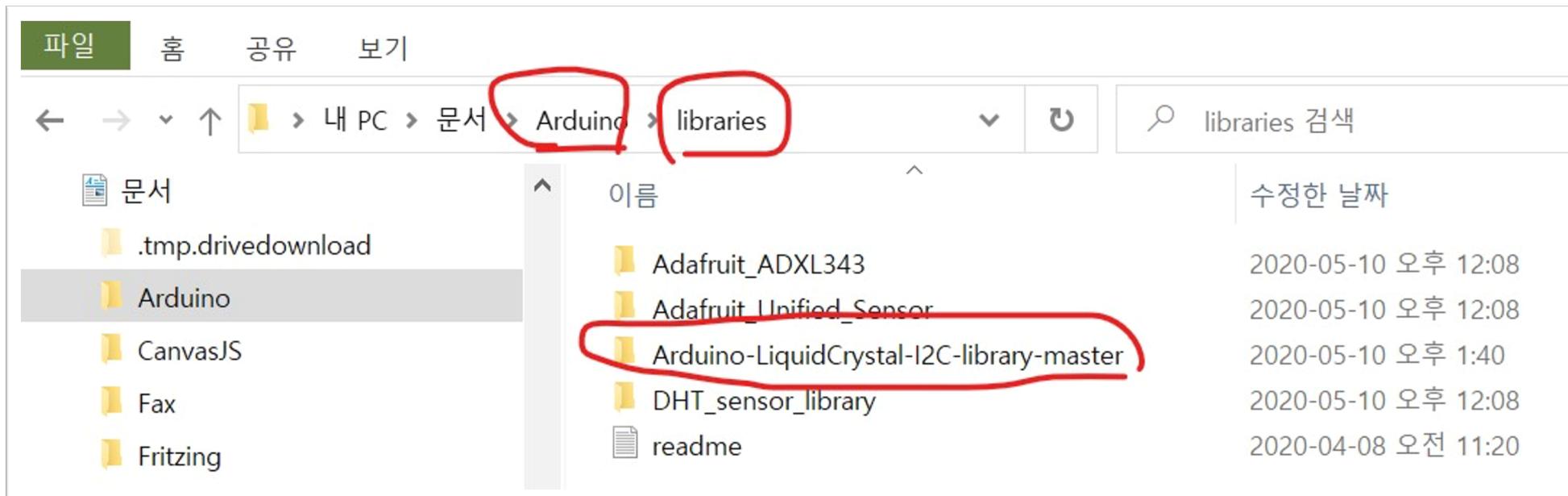


Fritzing 회로 구성



LiquidCrystal-I2C-library 설치

- Github에서 라이브러리 다운로드
 - <https://github.com/fdebrabander/Arduino-LiquidCrystal-I2C-library>
- 다운로드 받은 zip 파일을 “스케치->라이브러리 포함하기>zip 라이브리 추가”



스케치 작성

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <DHT.h>
#define DHTPIN D4
#define DHTTYPE DHT11

DHT dht(DHTPIN, DHTTYPE);
LiquidCrystal_I2C lcd(0x27,16,2);

void setup() {
    Serial.begin(115200);
    dht.begin();
    // lcd init
    lcd.begin();
    // lcd back light on
    lcd.backlight();
    //lcd.print("Test");
}
```

```
void loop() {
    float temp = dht.readTemperature();
    float humi = dht.readHumidity();
    Serial.print("Temp: ");
    Serial.print(temp);
    Serial.print(" Humi: ");
    Serial.println(humi);

    // LCD Display
    // LCD Cursor change to (0,0)
    lcd.setCursor(0,0);
    lcd.print("Temp: ");
    lcd.print(temp);

    // LCD Cursor change to (0,1)
    lcd.setCursor(0,1);
    lcd.print("Humi: ");
    lcd.print(humi);

    delay(1000);
}
```

I2C Scanner

- 아두이노 보드에서 I2C 통신을 하는 장치의 주소를 찾는 코드
- `LiquidCrystal_I2C lcd(0x27, 16, 2);`
 - I2C 버스에서 I2C 지원 장치(**0x27**) 검색
 - 다운로드 : https://github.com/IoT-Lab-02/Week05-Lab/blob/master/i2c_scanner/i2c_scanner.ino

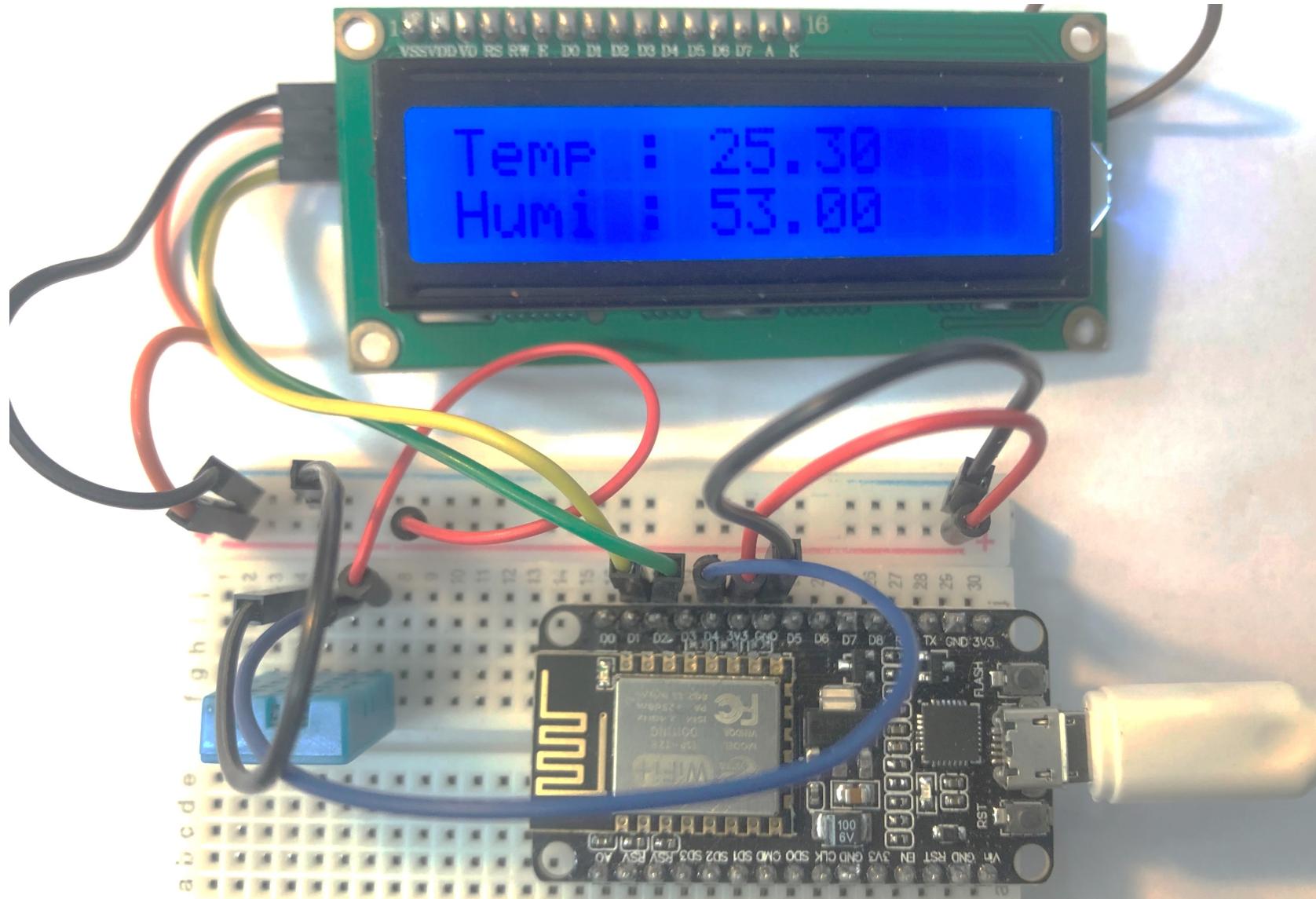
I2C Scanner 실행 결과

The image shows the Arduino IDE interface. On the left, the code for the I2C Scanner is displayed in the editor:

```
49  for(address = 1; address < 127; address++ )  
50  {  
51      // The i2c_scanner uses the return value of  
52      // the Write.endTransmission to see if  
53      // a device did acknowledge to the address.  
54      Wire.beginTransmission(address);  
55      error = Wire.endTransmission();  
56  
57      if (error == 0)  
58      {  
59          Serial.print("I2C device found at address 0x");  
60          if (address<16)  
61              Serial.print("0");  
62          Serial.print(address,HEX);  
63          Serial.println(" !");  
64  
65          nDevices++;  
66      }  
67      else if (error==4)  
68      {  
69          Serial.print("Unknown error at address 0x");  
70          if (address<16)  
71              Serial.print("0");  
72          Serial.println(address,HEX);  
73      }  
74  }  
75  if (nDevices == 0)  
76      Serial.println("No I2C devices found\n");  
77  else  
78      Serial.println("done\n");  
79  
80  delay(5000);           // wait 5 seconds for next scan  
81}
```

On the right, the Serial Monitor window titled "COM4" shows the output of the I2C scan. It displays four instances of "Scanning..." followed by "I2C device found at address 0x27 !", each highlighted with a red oval. Below the serial output, the baud rate is set to 9600.

실행 결과



실습 과제

1. DHT11, NodeMCU를 사용하여 온습도를 Serial Monitor와 Serial Ploter로 출력하시오.
2. DHT11, LCD, NodeMCU를 사용하여 온습도를 LCD에 출력하시오. 온습도를 Serial Monitor에도 동시에 출력하다.
3. I2C Scanner 코드를 실행하여 자신의 NodeMCU에 있는 I2C 장치(LCD 모듈)의 주소를 찾으시오.