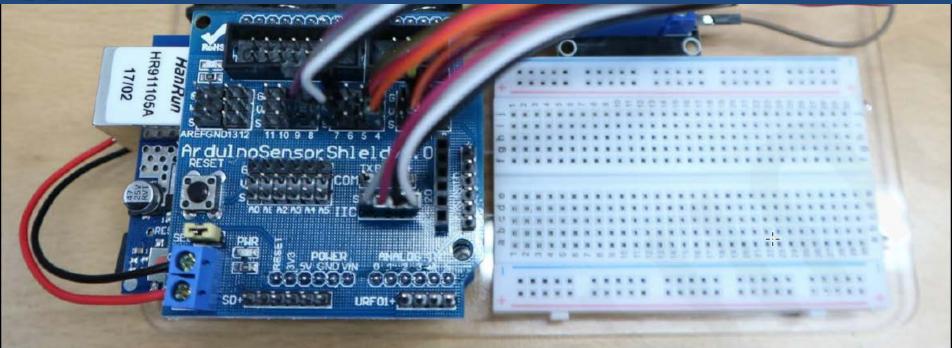


아크릴 판에 보드 고정하기



키트에서 아크릴판과 아두이노 보드, 브레드 보드, 나사 등을 꺼내어 조립합니다.



보드 알펴보기

http://www.eduino.kr/product/detail.html?product_no=59



정품 보드가 아닌 호환 보드입니다.

UNO R3 SMD BOARD

아두이노 우노 R3 SMD 호환보드







http://www.eduino.kr/product/detail.htm ?product_no=59



Microcontroller ATmega328

Operating Voltage 5V

Input Voltage (recommended) 7–12V

Input Voltage (limits) 6–20V

Digital I/O Pins 14 (of which 6 provide PWM output)

Analog Input Pins 6

DC Current per I/O Pin 40 mA

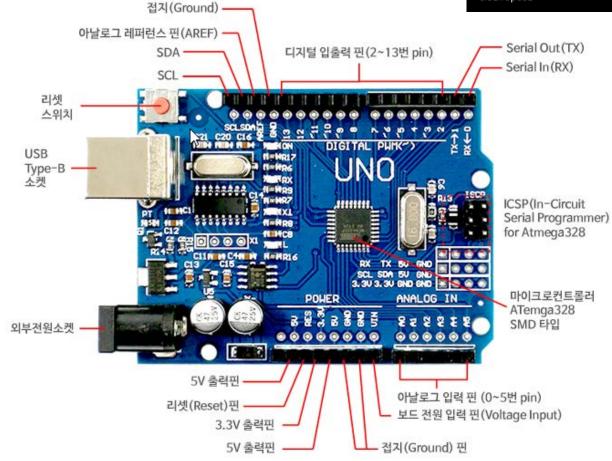
DC Current for 3.3V Pin 50 mA

Flash Memory 32KB (ATmega328) of which 0.5KB used by bootloader

SRAM 2KB (ATmega328)

EEPROM 1KB (ATmega328)

Clock Speed 168







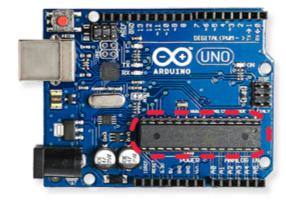
보드 알펴보기

http://www.eduino.kr/product/detail.html ?product_no=59

DIP타입과 SMD타입의 차이는?

반도체, 소자의 경우에는 그 형태에 따라 다양하게 정의됩니다. 아누이노 우노의 경우에는 DIP타입과 SMD타입 2가지가 존재하는데 그 차이는 다음 그림과 같습니다. 즉, 마이크로 프로세서인 ATmega328 보드의 형태에 따라 DIP, SMD타입이라 부르며 DIP타입의 경우에는 탈부착이 가능하여 메인보드 프로세서인 ATmega328 보드가 고장이 났을 때 교체할 수 있는 장점이 있습니다.

DIP타입 소자의 간격은 2.54mm로 브레드보드(Breadboard)나 만능기판에 탈부착 할 수 있습니다.









아두이노 호환보드(CH430&CH341) 드라이버 설치하기

http://www.eduino.kr/board/gallery/read.html?no=3975&board_no=8



Code formatted for HTML has been copied to the clipboard.

스케치는 프로그램 저장 공간 928 바이트(2%)를 사용. 최대 32,256 바이트.
전역 변수는 동적 메모리 9바이트(0%)를 사용, 2,039바이트의 지역변수가 남음. 최대는 2,048 바이트.
avrdude: ser_open(): can't open device "喘,#COM6": 지정된 파일을 찾을 수 없습니다.

보드에 업로당중에 문제 발생. 다음을 참고하세요. http://www.arduino.cc/en/Buide/Troubleshooting#upload

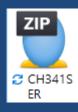
Arduino/Genuino Uno on COM8



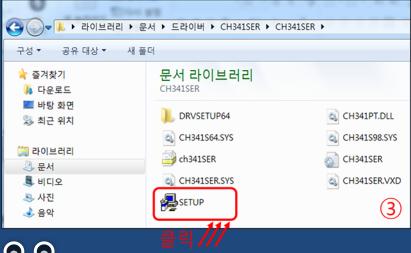


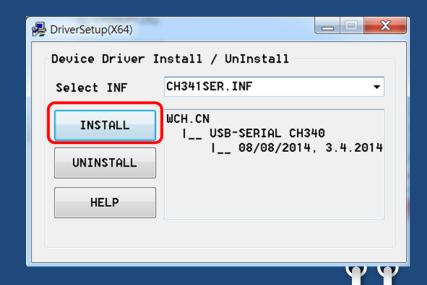
아두이노 호환보드(CH430&CH341) 드라이버 설치하기

http://www.eduino.kr/board/gallery/read.html?no=3975&board_no=8





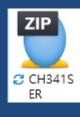


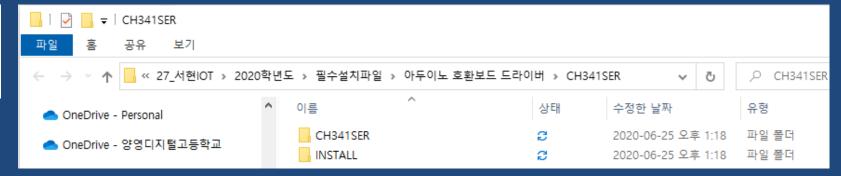


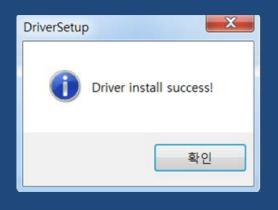


아두이노 호환보드(CH430&CH341) 드라이버 설치하기

http://www.eduino.kr/board/gallery/read.html?no=3975&board_no=8

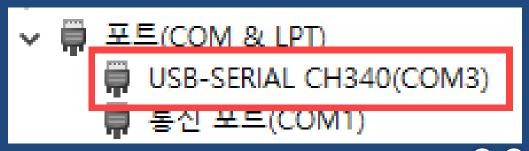








이후 아두이노 보드를 컴퓨터에 연결하면, CH430 포트가 생성 포트 번호는 컴퓨터마다 다를 수 있음.

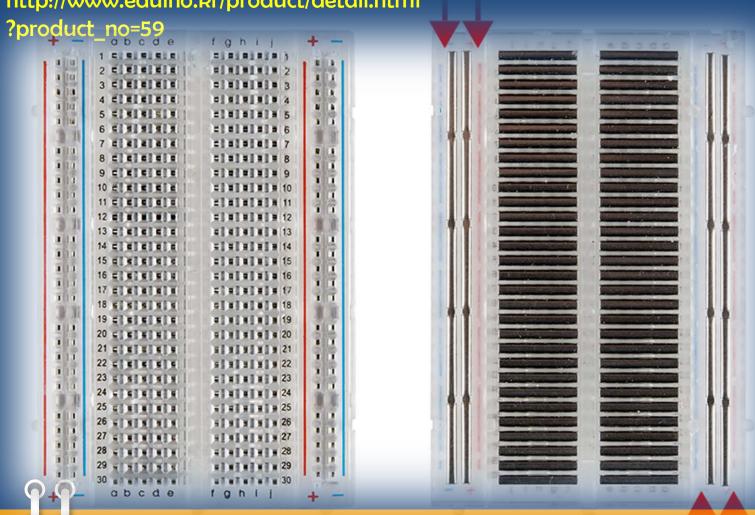






보드 살펴보기

http://www.eduino.kr/product/detail.html



https://www.arduino.cc/en/Main/Software







https://www.arduino.cc/en/Main/Software



ARDUINO 1.8.13

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.

This software can be used with any Arduino board. Refer to the Getting Started page for Installation instructions.

Windows Installer, for Windows 7 and up

Windows ZIP file for non admin install

Windows app Requires Win 8.1 or 10



Mac OS X 10.10 or newer

Linux 32 bits

Linux 64 bits

Linux ARM 32 bits

Linux ARM 64 bits

Release Notes Source Code Checksums (sha512)





https://www.arduino.cc/en/Main/Software

STORE

SOFTWARE

EDUCATION

PRO

RESOURCES

COMMUNITY

HELP

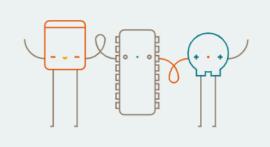
Contribute to the Arduino Software



arduino-1.8.13-windows.zip

38.2MB / 192MB | 20초 남음

Consider supporting the Arduino Software by contributing to its development. (US takes payers, prease note and contribution will be used.



SINCE MARCH 2015, THE ARDUINO IDE HAS BEEN DOWNLOADED

42,983,611 TIMES. (IMPRESSIVE!) NO LONGER JUST FOR ARDUINO AND GENUINO BOARDS, HUNDREDS OF COMPANIES AROUND THE WORLD ARE USING THE IDE TO PROGRAM THEIR DEVICES, INCLUDING COMPATIBLES, CLONES, AND EVEN COUNTERFEITS. HELP ACCELERATE ITS DEVELOPMENT WITH A SMALL CONTRIBUTION! REMEMBER: OPEN SOURCE IS LOVE!

\$3

\$5

\$10

\$25

\$50

OTHER



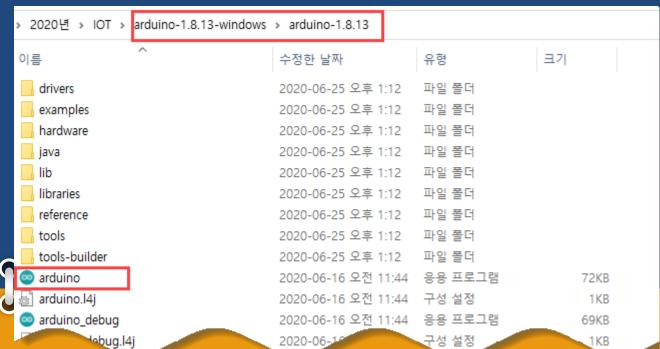


https://www.arduino.cc/en/Main/Software



C 드라이브 또는 D 드라이브에 'IOT' 폴더를 만드시고,

압축 파일을 붙여 넣고, 압축을 풀어 주고, 폴더 안에 있는 'arduino' 실행







아두이노 보드



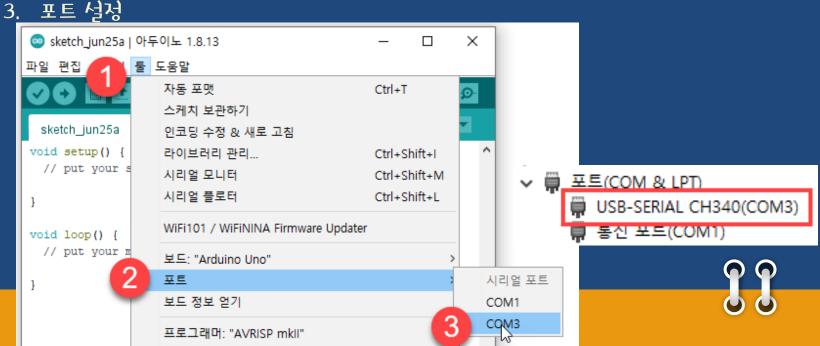
아두이노 보드를 연결하고,



프로그램 실행

부트로더 굽기



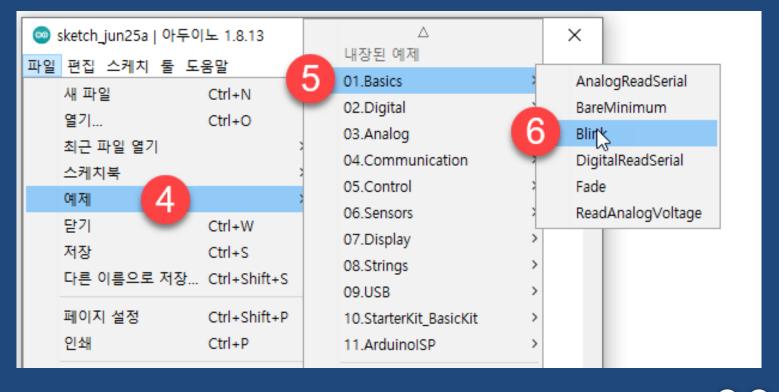




아두이노 보드 동작 확인하기



4. 예제 파일 불러오기 (01,Basics – Blink)







예제 파일 살펴 보기

```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED BUILTIN as an output.
  pinMode(LED BUILTIN, OUTPUT);
                                     ◎ sketch_aug31b | 아두이노 1.6.10
                                                                          \times
                                    파일 편집 스케치 툴 도움말
                                         // the loop function runs over and
                                     sketch_aug31b §
void loop() {
                                     /* 범위
                                                              /* 내용 */
  digitalWrite(LED BUILTIN, HIGH);
                                                                             ge level)
                                       주석
  delay(1000);
                                       입니다.
                                                              범위 주석
  digitalWrite(LED BUILTIN, LOW);
                                                                             tage LOW
  delay(1000);
                                                              // 내용
                                     // 한줄 주석 입니다.
                                                               한줄 주석
                                    void setup() {
                                      명령문;
                                                            문장의 끝에는
                                      명령문;
                                                            항상 ;(세미콜론)
                                                            을 붙인다.
                                    void loop() {
                                      명령문;
                                      명령문;
```

예제 파일 살펴 보기

- Setup() 함수 : 하드웨어 초기화 함수 한 번 실행
- loop() 함수 : 하드웨어 반복 실행 함수 무한 반복 실행



예제 파일 살펴 보기



Ctrl 키와 훧 을 사용하면, 스케치 프로그램 내의 폰트의 크기를 조절 할 수 있어요.



- pinMode(핀번호, INPUT/OUTPUT) 함수 : 연결 핀의 입력 또는 출력 설정
- digitalWrite(핀번호, HIGH/Low) 함수: 5V 또는 3.3V 신호의 High 신호를 출력
 하거나, oV의 Low 신호를 출력하는 함수
- delay(숫자) 함수 : 0,001 초 단위의 실행 대기 함수



프로그램 컴파일 및 보드에 업로드





- '컴파일'을 활용하여 프로그램의 오류를 찾을 수 있습니다.
- '컴파일 및 보드 업로드 ' 를 활용하여 컴파일이 끝나면, 자동으로 보드에 업로 드할 수 있습니다.



프로그램 컴파일 및 보드에 업로드





- '컴파일 및 보드 업로드'를 클릭하여, 예제 프로그램을 업로드합니다.
- 정상적인 업로드 후 화면 하단의 메시지는 다음과 같습니다.

업로드 완료.

스케치는 프로그램 저장 공간 924 바이트(2%)를 사용. 최대 32256 바이트. 전역 변수는 동적 메모리 9바이트(0%)를 사용, 2039바이트의 지역변수가 남음. 최대는 2048 바이트.



내장 LED 깜박임 속도 조절하기



• 예제 프로그램을 활용하여 내장 LED의 깜박임 속도를 조절해 봅시다.





시리얼 포트를 활용하여 시리얼 모니터에 출력해 보자.



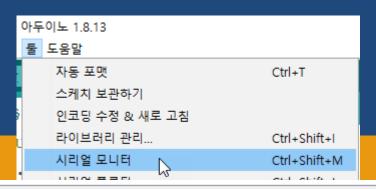
• 시리얼 모니터는 실습 과정에서 실행 결과 또는 논리적 오류 추출을 위한 디버깅 도구로 활용됩니다.

- 활용함수
 - 1. 시리얼 포트의 신호 전송 속도 설정 : Serial,begin(통신속도);
 - 2. 시리얼 모니터에 출력 명령 :

Serail.print(출력내용/변수) : 내용 출력 후 줄바꿈하지 않음.

Serial.println(출력내용/변수) : 내용 출력 후 줄바꿈

- 3. 프로그램 컴파일 후, 업로드
- 4. 시리얼모니터 실행 후 출력 내용 확인





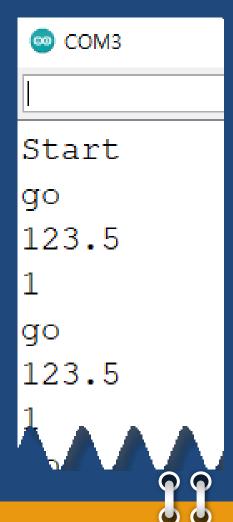


시리얼 모니터 활용 실습



• 아래의 프로그램을 직접 입력해 보고, 실행 결과를 예측해 보고, 확인해 보자.

```
void setup() {
  Serial.begin (9600);
  pinMode(LED BUILTIN, OUTPUT);
  Serial.print("Start");
  Serial.print("\n");
void loop() {
  Serial.println("go");
  Serial.println(123.546,1);
  digitalWrite(LED BUILTIN, HIGH);
  Serial.println(digitalRead(13));
  delay(2000);
  digitalWrite(LED BUILTIN, LOW);
  Serial.println(digitalRead(13));
  delay(2000);
```





시리얼 모니터 활용 실습_혹시요?



· 동일한 결과가 나오는 프로그램을 좀 더 짧게 작성할 수는 없을까요? (힌트 : C언어의 연산자를 활용해 보자.)

```
void setup() {
  Serial.begin (9600);
  pinMode(LED BUILTIN, OUTPUT);
  Serial.print("Start");
  Serial.print("\n");
void loop() {
  Serial.println("go");
  Serial.println(123.546,1);
  digitalWrite(
  Serial.println(digitalRead(13));
  delay(2000);
```

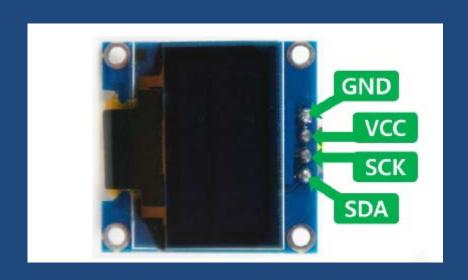
```
COM3
Start
qo
123.5
go
123.5
go
123.5
```







OLED 디스플레이 모듈



- 박막 발광다이오드
- 0.9인치 초소형 디스클 레이 모듈
- 저전력으로 웨어러블이나 IOT기기에 적합.
- 128*64 해상도 지원
- 글자와 이미지 표현가능

• 스마트폰, 텔레비전, 모니터, 조명, 자동차 후미등 등에 사용되고 있음





라이브러리 추가하기

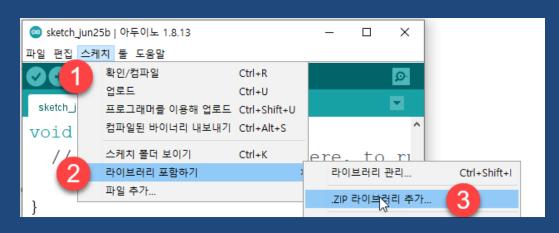


OzOLED.zip 파일을 라이브러리에 추가하기

아두이노 스케치 프로그램 실행 후

스케치 - 라이브러리 포함하기 - ZIP 라이브러리 추가

에서 다운받은 파일 선택



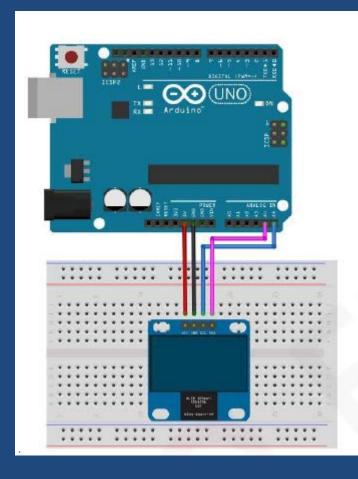


• 스케치 — 라이브러리 포함하기 에서 목록에 OzOLED라이브러리 있는지 확인

OLED 연결 회로구성







아두이노 우노 보드	OLED 디스플레이
GND	GND
5V	VCC
A5	SCA
A4	SDA



OzOLED.h 는 라이브러리 포함하기를 통해 가져옴.

```
확인/컴파일
                      Ctrl+U
프로그래머를 이용해 업로드 Ctrl+Shift+U
컴파일된 바이너리 내보내기 Ctrl+Alt+S
스케치 폴더 보이기
                                      라이브러리 관리...
                                                           Ctrl+Shift+L
파일 추가
                                      .ZIP 라이브러리 추가...
                                      아두이노 라이브러리
                                      Bridge
                                      Ethernet
                                      GSM
                                      Keyboard
                                      LiquidCrystal
                                      Robot Control
                                      Robot IR Remote
                                      Robot Motor
                                      SD
                                      Servo
                                      SpacebrewYun
                                      Stepper
                                      TFT
                                      Temboo
                                      WiFi
                                     Contributed 라이브러리
                                      EEPROM
                                      OzOLED
```

```
#include <OzOLED.h>
void setup() {
  Serial.begin (9600);
 OzOled.init();
 pinMode(LED BUILTIN, OUTPUT);
 Serial.print("Start");
  Serial.print("\n");
void loop() {
  Serial.println("go");
  Serial.println(123.546,1);
  digitalWrite(LED BUILTIN, HIGH);
  Serial.println(digitalRead(13));
 OzOled.printString("13th pin : ", 0, 1);
 OzOled.printNumber((long)digitalRead(13), 11, 1);
 delay(2000);
  digitalWrite(LED BUILTIN, LOW);
  Serial.println(digitalRead(13));
 OzOled.printString("13th pin : ", 0, 1);
 OzOled.printNumber((long)digitalRead(13), 11, 1);
 delay(2000);
```

온 - 습도 센서(DHT-11)

- 대기의 온도와 습도를 신호로 출력하는 센서
- 저항값이 변하는 것을 이용하여 값을 측정
- 온도: 0~50도, 습도 20~90% 측정가능

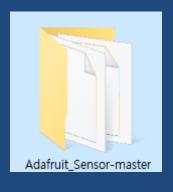




라이브러리 복사

• DHT 폴더와 Adafruit_Sensor-master폴더를 Arduino\libraries 하위폴더에 복사



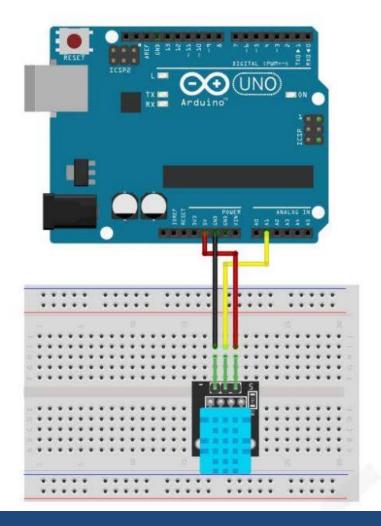




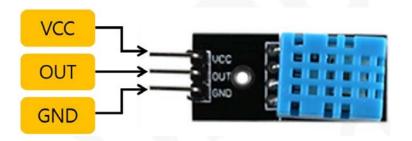
온습도센서 회로 구성







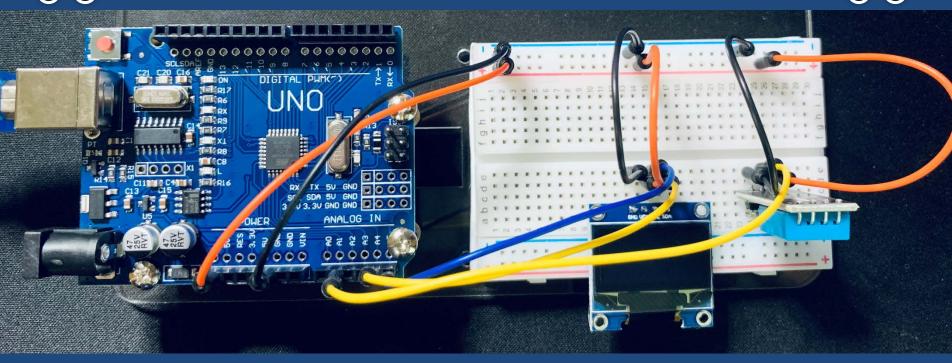
아두이노 우노 보드	온・습도 센서
5V	VCC
GND	GND
A1	DATA



연결사진





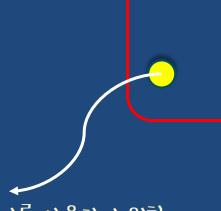




GND 와 VCC 를 보드로 연결하고, 각 부품의 GND와 VCC를 보드에서 연결하는 것이 깔끔!

코 드







이 부분은 온습도 센서를 사용하기 위한 입력입니다.

'A1' 만 상황에 따라 변경되며, 나머지는 동일 하게 입력합니다.

단, 미리 라이브러리를 복사 해 놓아야 합니다.

```
#include <DHT.h>
#define DHTPIN A1
#define DHTTYPE DHT11
DHT dht (DHTPIN, DHTTYPE);
void setup() {
  Serial.begin (9600);
void loop() {
  int h = dht.readHumidity();
  int t = dht.readTemperature();
  Serial.print("humidity : ");
  Serial.println(h);
  Serial.print("temperature : ");
 Serial.println(t);
  delay(5000);
```

시리얼 모니터 확인





```
humidity: 0
temperature: 0
humidity: 62
temperature: 27
humidity: 62
temperature: 27
```



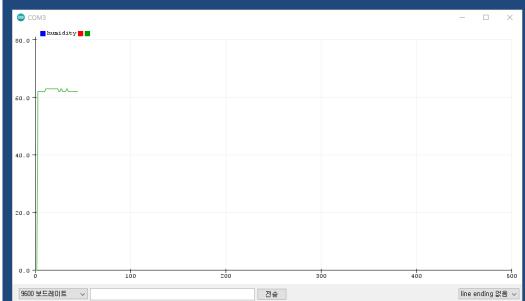
온습도 표시 속도가 느리면, 알아서 조절하시길…

시리얼 플로터로 습도값 변화 확인하기

```
#include <DHT.h>
#define DHTPIN A1
#define DHTTYPE DHT11
DHT dht (DHTPIN, DHTTYPE);
void setup() {
  Serial.begin (9600);
void loop() {
  int h = dht.readHumidity();
  int t = dht.readTemperature();
  Serial.print("humidity : ");
  Serial.println(h);
  //Serial.print("temperature : ");
  //Serial.println(t);
  delay(1000);
```

```
물 도움말

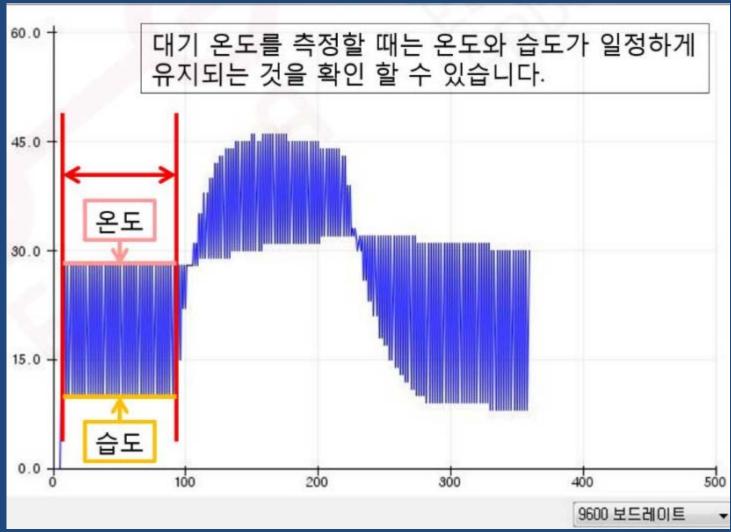
자동 포맷 Ctrl+T
스케치 보관하기
인코딩 수정 & 새로 고침
라이브러리 관리... Ctrl+Shift+I
시리얼 모니터 Ctrl+Shift+M
시리얼 플로터 Ctrl+Shift+L
```



시리얼 플로터 확인







실습



온습도 센서의 측정값을 OzOLED 에 표시해 볼까요?

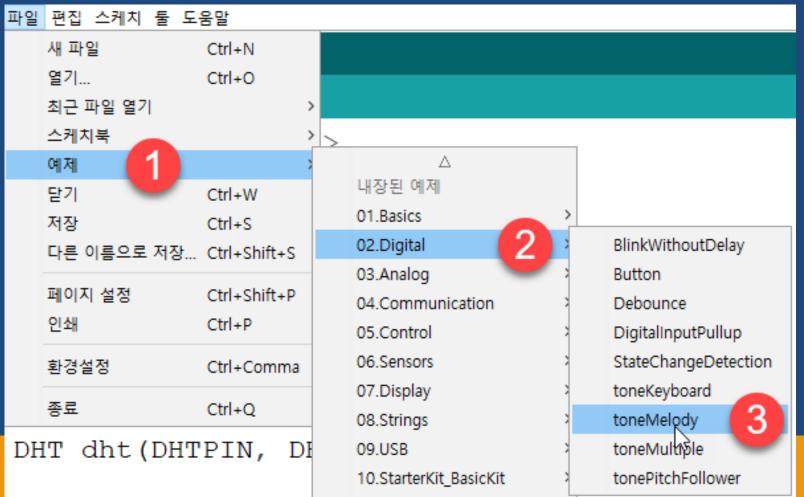
```
#include <OzOLED.h>
#include <DHT.h>
#define DHTPIN A1
#define DHTTYPE DHT11
DHT dht (DHTPIN, DHTTYPE);
void setup() {
  OzOled.init();
void loop() {
  int h = dht.readHumidity();
  int t = dht.readTemperature();
  OzOled.printString(
  OzOled.printNumber(
  OzOled.printString(
  OzOled.printNumber(
  delay(2000);
```



실습 : 부저를 연결하고, 멜로디를 재생시켜 보자.



▶ 예제 프로그램을 활용해 볼까요?





실습 : 부저를 연결하고, 멜로디를 재생시켜 보자.



구석 프로그램을 지우고, 필요한 프로그램만 남겨서, 이해해 보자.



• + : 8번 pin • - : GND



```
#include "pitches.h"
int melody[] = {
  NOTE C4, NOTE G3, NOTE G3, NOTE A3, NOTE G3, 0, NOTE B3, NOTE C4
};
int noteDurations[] = {
 4, 8, 8, 4, 4, 4, 4, 4
};
void setup() {
  for (int thisNote = 0; thisNote < 8; thisNote++) {</pre>
    int noteDuration = 1000 / noteDurations[thisNote];
    tone (8, melody[thisNote], noteDuration);
    int pauseBetweenNotes = noteDuration * 1.30;
    delay (pauseBetweenNotes);
    noTone (8);
void loop() {
```

- 온습도센서와 LED, 부저를 이용하여 LED 에 온습도를 표시하고, 일정한 온도 이상 이 되면 멜로디가 재생되게 하세요.
- Advanced : 그런데, 멜로디가 계속 재생되니, 시끄럽네요…어떻게 일정 횟수만 재생되고, 더 이상은 재생이 안되게 할 수 있을까요?



```
void loop() {
  int h = dht.readHumidity();
  int t = dht.readTemperature();
  if(t > 25) {
    melody call();
    cnt --;
    if (cnt == 0) while (1);
  Serial.println(cnt);
  delay(1000);
void melody call() {
  for (int thisNote = 0; thisNote < 8; thisNote++) {</pre>
    int noteDuration = 1000 / noteDurations[thisNote];
    tone(BUZERPIN, melody[thisNote], noteDuration);
    int pauseBetweenNotes = noteDuration * 1.30;
    delay (pauseBetweenNotes);
    noTone (BUZERPIN);
```