

ARDUINO

사물인터넷  
Internet  
of  
Things



스마트인재개발원  
Smart Human Resources Development

스마트인재개발원  
나 예 호 선임연구원

18세기

19세기

20세기

2017년

## 1차 산업혁명

- ✓ 수력증기기관
- ✓ 철도, 운송수단 혁명
- ✓ 면사방직기



## 2차 산업혁명

- ✓ 조립설비
- ✓ 컨베이어벨트



## 3차 산업혁명

- ✓ 개인용 PC보급
- ✓ 정보기술발달
- ✓ 컴퓨터 자동 제어화



## 4차 산업혁명

- ✓ 초연결사회
- ✓ IoT
- ✓ 인공지능
- ✓ 빅데이터



2021년 130여개국, 2,000여 개 업체 전시



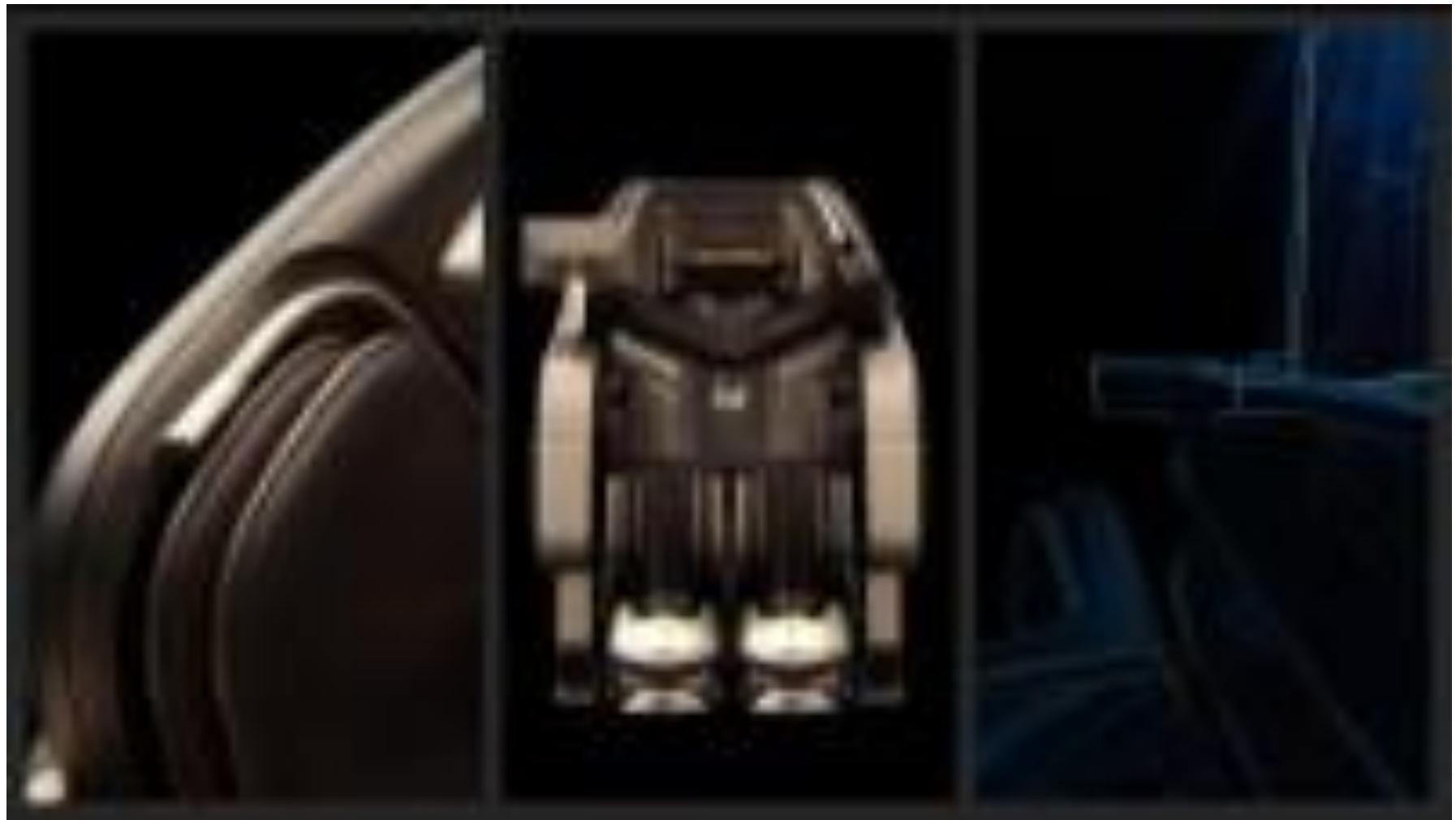
2019년 155개국, 4,400여 개 업체 전시

2020년 161개국, 4,500여 개 업체 전시

2021년 130여개국, 2,000여 개 업체 전시

2022년 160여개국, 2,300여 개 업체 전시

# Consumer Electronic Show





## TRACKS EVERYTHING

TOM BRADY  
QB, New England

ATHLETE RECOVERY SLEEPWEAR  
**REST. WIN. REPEAT.**

ENGINEERED WITH TB12 RECOVERY TECHNOLOGY

UA SPEEDFORM™  
VELOCITI RE

TRACK YOUR SLEEP  
WITH UA RECORD.



UA SPEEDFORM™  
EUROPA RE

UA SPEEDFORM™  
GEMINI 3 RE

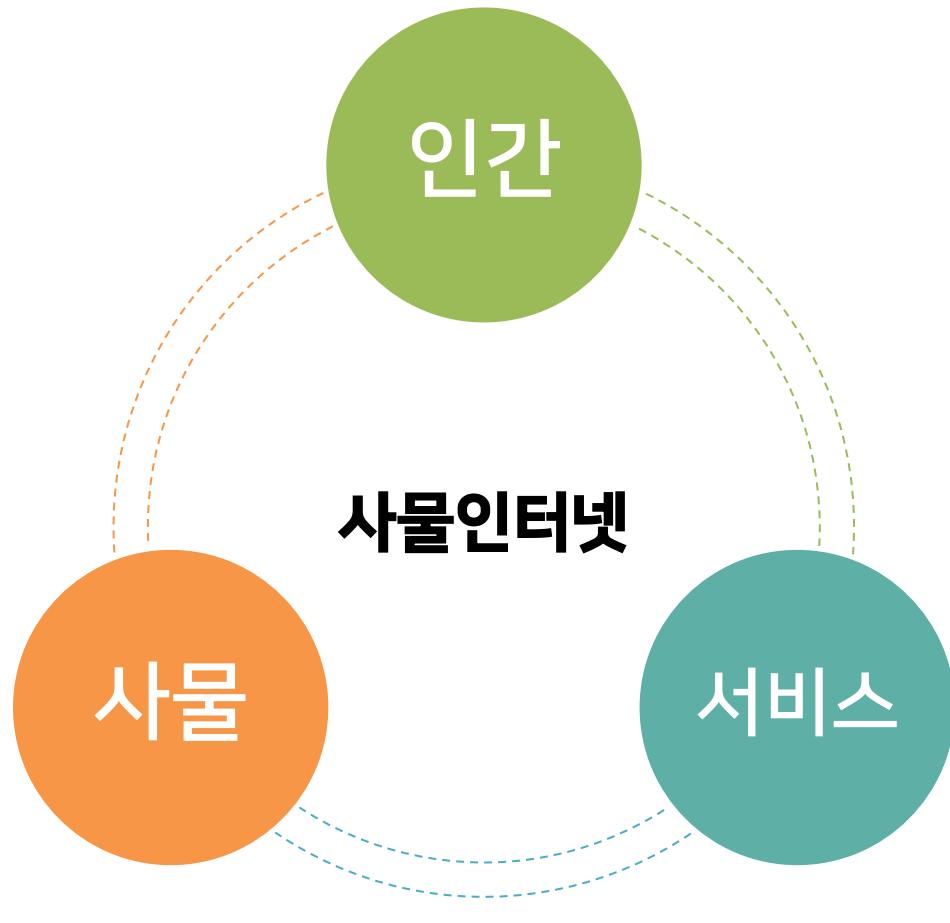


스마트인재개발원  
Smart Human Resources Development

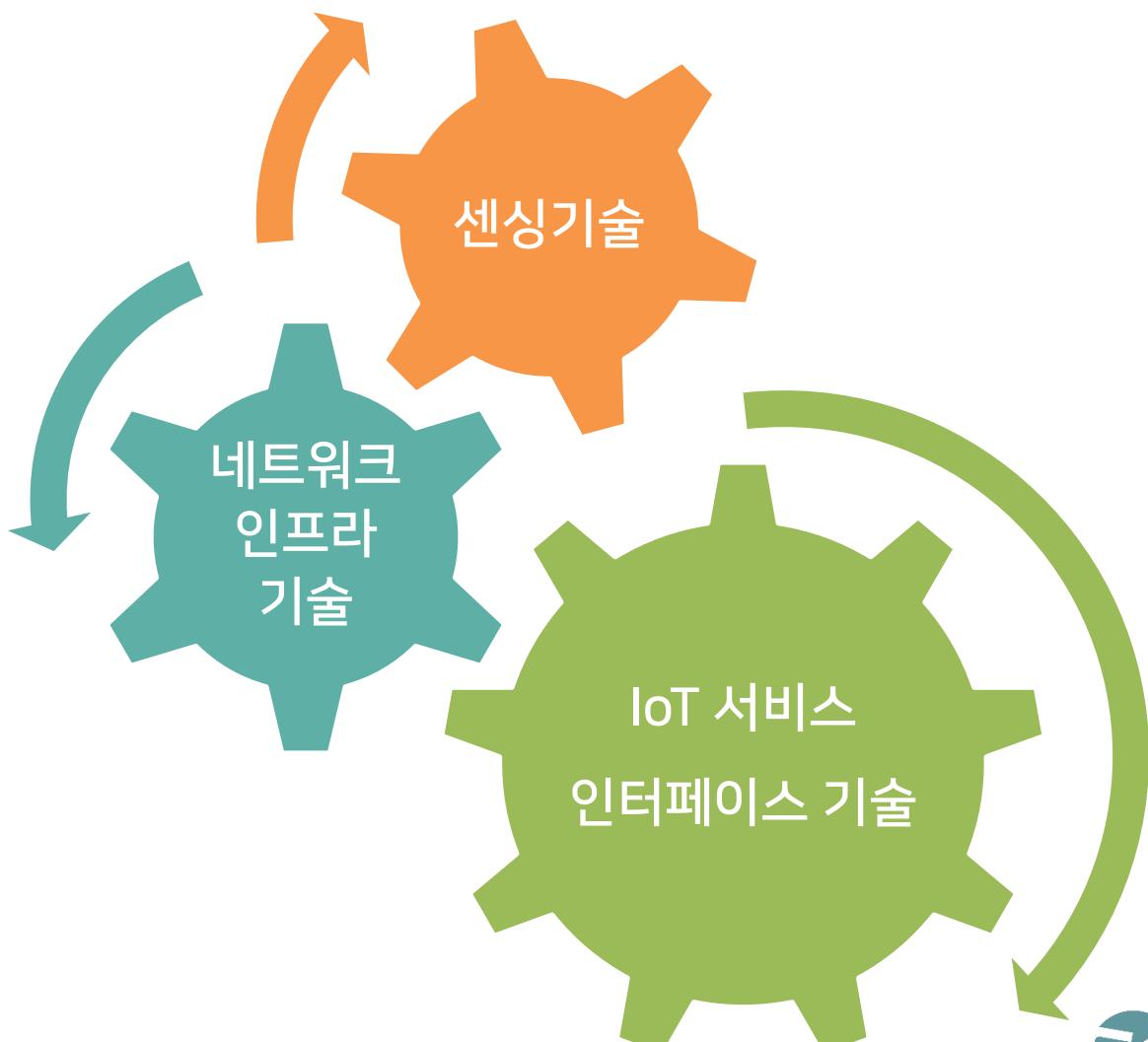
# 사물인터넷이란?



인간, 사물, 공간, 서비스 등 모든 사물을 하나로  
연결시켜 새로운 부가가치를 창출하는 것



# 사물인터넷 3대 주요 기술

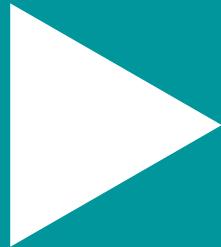


스마트인재개발원  
Smart Human Resources Development

# IoT 제품



스마트인재개발원  
Smart Human Resources Development



## 개인 IoT

### Car as a Service



차량을 인터넷으로 연결

→ 안전하고 편리한 운전

### Healthcare as a Service



심장박동, 운동량 등 IoT 정보 제공

→ 개인 건강 증진

### Home as a Service



주거환경 IoT 통합 제어

→ 생활 편의, 안전성 제고

## 공공 IoT

### Public Safety as a Service



CCTV, 노약자 GPS 등 IoT 정보제공  
→ 재난·재해 예방

### Environment as a Service



대기질, 쓰레기양 등 IoT 정보제공  
→ 환경오염 최소화

### Energy as a Service



에너지 관련 IoT 정보제공  
→ 에너지 관리 효율성 증대

## 산업 IoT

### Farm(&Food) as a Service



공정분석 및 시설물 모니터링  
→ 작업 효율 및 안전 제고

### Product as a Service



주변 생활제품의 IoT 접목  
→ 고부가 서비스 제품화

### Factory as a Service



생산·가공·유통 IoT 접목  
→ 생산성향상 및 안전유통체계

## ■ 단말 분실 및 도난

- ▶ 문제점 : 사용자 소유의 스마트기기 분실 및 도난 또는 파손으로 IoT서비스가 중단되고 **정보유출** 가능성
- ▶ 해결방안 : 사용자 주의요망, 위치추적

## ■ 데이터 위·변조

- ▶ 문제점 : 데이터를 중간에 가로채 **위·변조** 한 뒤 정상적인 기기가 이를 송신한 것으로 위장
- ▶ 해결방안 : 데이터 암호화를 하여 보안 강화,  
공용WiFi 사용자제, LTE등 개인망 이용

# IoT 디바이스 플랫폼



센서 디바이스 플랫폼	기업	특징
Arduino	The Arduino logo, featuring a teal infinity symbol with a minus sign on the left and a plus sign on the right, followed by the word "ARDUINO" in a teal sans-serif font.	Atmel <ul style="list-style-type: none"><li>- ATMega 계열 저전력 프로세서이용</li><li>- Arduino 통합개발환경 제공, C++언어기반 개발</li><li>- Window, Mac, Linux OS 플랫폼 지원</li></ul>
RaspberryPi	The Raspberry Pi logo, which is a stylized illustration of a red raspberry with green leaves.	Broadcom <ul style="list-style-type: none"><li>- Broadcom BCM2835 Soc, ARM Cortex-A7 0.9Hz 프로세서</li><li>- 이클립스와 같은 기존 존재 통합개발환경 이용</li><li>- Linux OS플랫폼 중심, Python 언어 기반 개발</li></ul>

- **Arduino – 단순한 기기를 만들고 싶은 입문자**
  - ▶ 장점 : 싸다 / 전원을 적게 먹는다 / 예제·튜토리얼 등을 쉽게 찾을 수 있다
  - ▶ 단점 : 복잡하거나 많은 계산량을 필요로 하는 기기에는 부적합하다.
  - ▶ 예제 : 세탁알림 기기 / 도어락 시스템 / 창문 블라인드 자동화기기
- **RaspberryPi – 복잡 / 멀티미디어 / 리눅스 기반**
  - ▶ 장점 : 모니터 연결 가능 / USB 활용 가능 / OS 교체 가능
  - ▶ 단점 : 아두이노에 비해 외부센서나 버튼의 선택권이 제한적이다

# 아두이노란 ?



# 나의 상상력이 듬뿍 담긴 물건, 환경을 간단하게 만들어낼 수 있는 장난감



# 아두이노란 ?



- AVR 기반의 마이크로컨트롤러하드웨어  
+ 쉬운 소프트웨어 개발환경(IDE)
- 오픈소스, 나만의 아두이노 제작가능



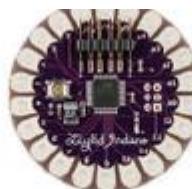
UNO R3



MEGA2560



NANO



LILYPAD



FIO



YUN



DUE



스마트인재개발원  
Smart Human Resources Development

# 아두이노 구성



## 아두이노 보드 (HW)



## 아두이노 소프트웨어

```
//LED blink example
int ledPin = 13;

void setup()
{
  pinMode(ledPin,OUTPUT);
}

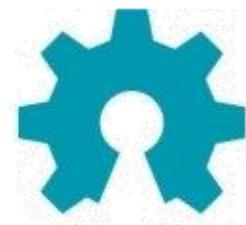
void loop ()
{
  digitalWrite(ledPin, HIGH);
  delay(1000);
  digitalWrite(ledPin, LOW);
  delay(1000);
}
```

Done uploading.  
Binary sketch size: 896 bytes (of a 14336 byte maximum)

## 오픈 소스



open source™



open source  
hardware



스마트인재개발원  
Smart Human Resources Development

# 아두이노 활용범위



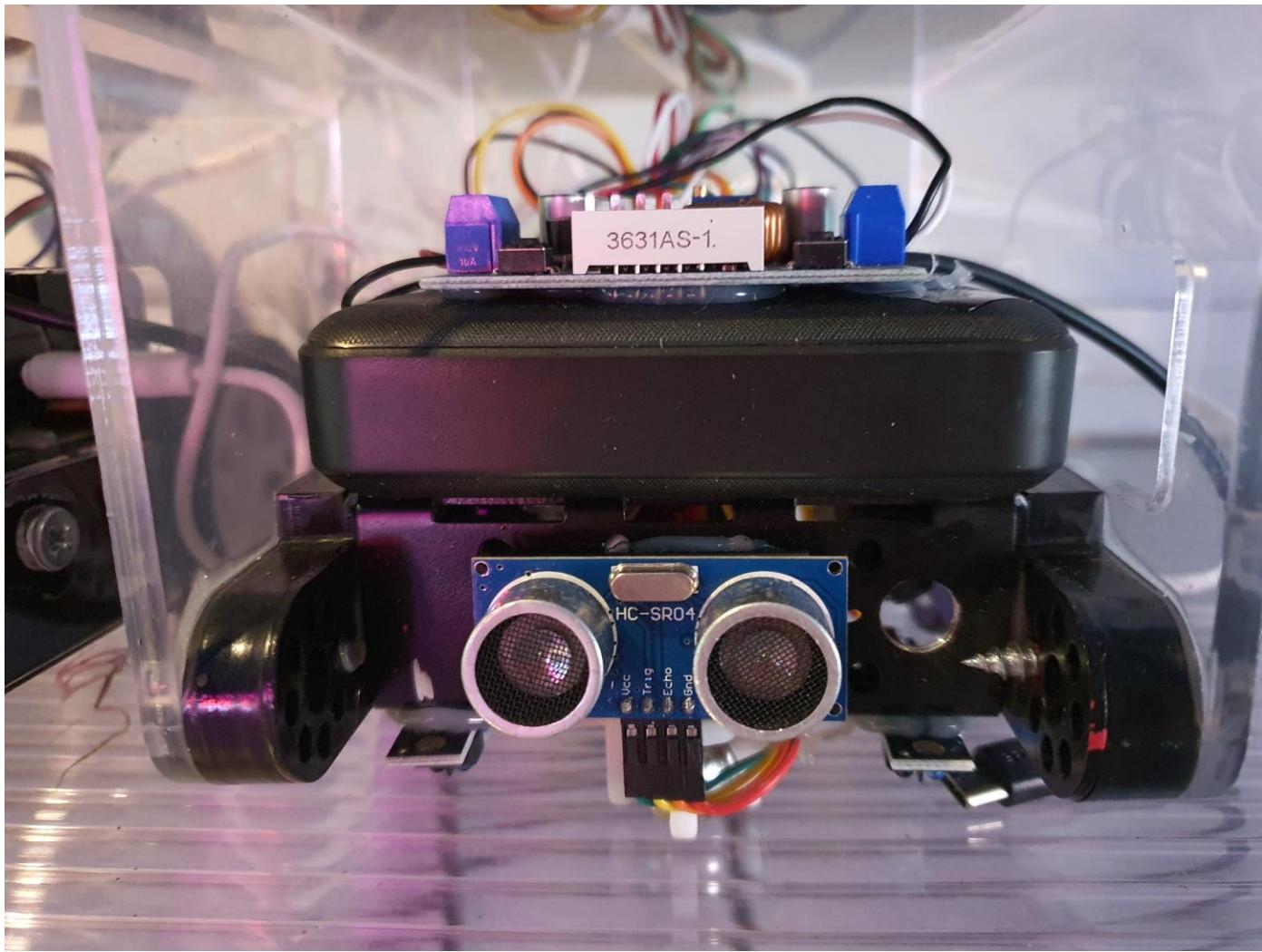
- **로봇공학(모바일/ 휴머노이드/ 물고기로봇등)**
- **음악및사운드장치**
- **게임분야와의연동**
- **다양한센서와의연동(온도, 습도, 기울기등)**
- **모바일애플리케이션(스마트폰과의통신)**
- **네트워킹시스템(유선혹은무선통신망구축)**
- **스마트홈구현(조명제어및실시간모니터링)**

# 아두이노 활용범위

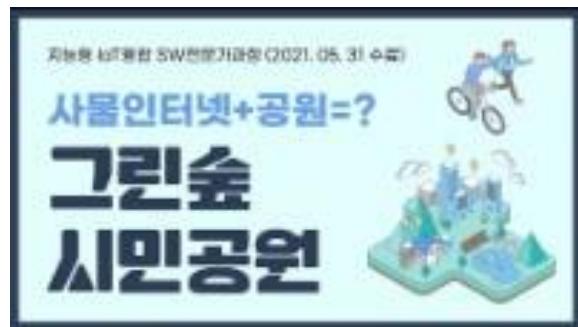


스마트인재개발원  
Smart Human Resources Development

# 아두이노 활용범위



# 교육생 프로젝트

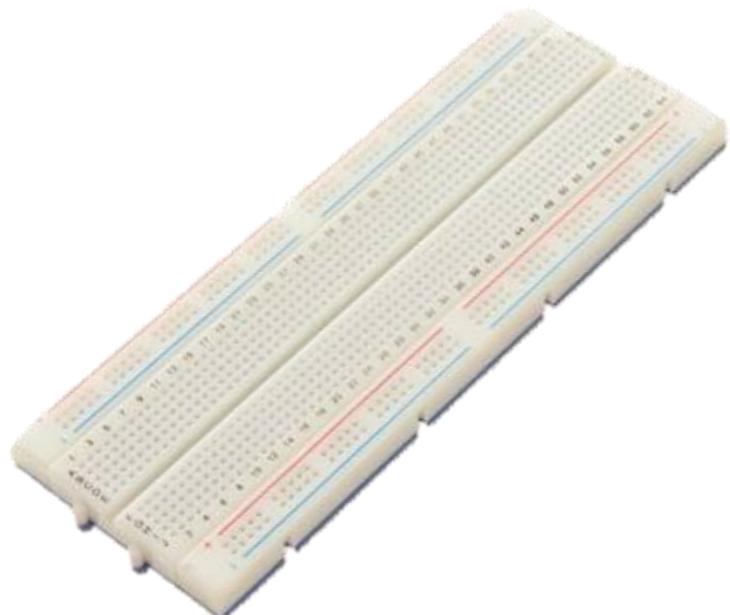


스마트인재개발원  
Smart Human Resources Development

1) 우노 보드



2) 브레드보드(빵판)



## 3) USB 케이블



4) DC 모터



5) 서버 모터 /  
6) 서버 모터 날개



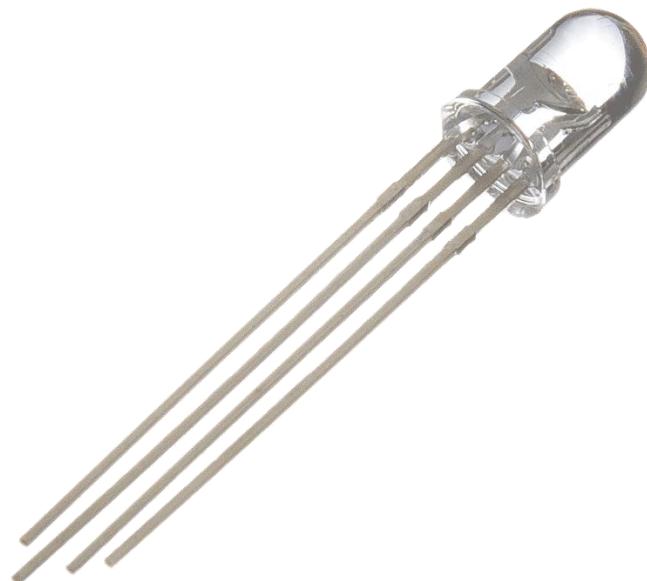
7) 저항



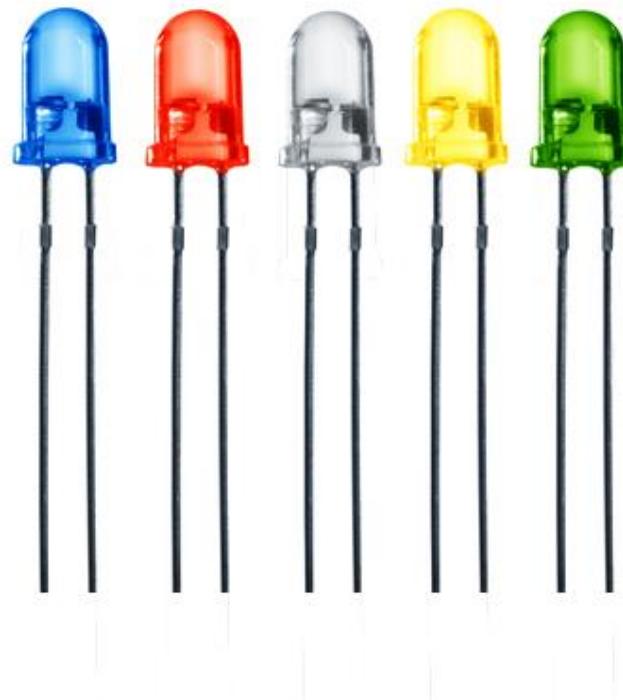
8) 가변저항



9) RGB LED



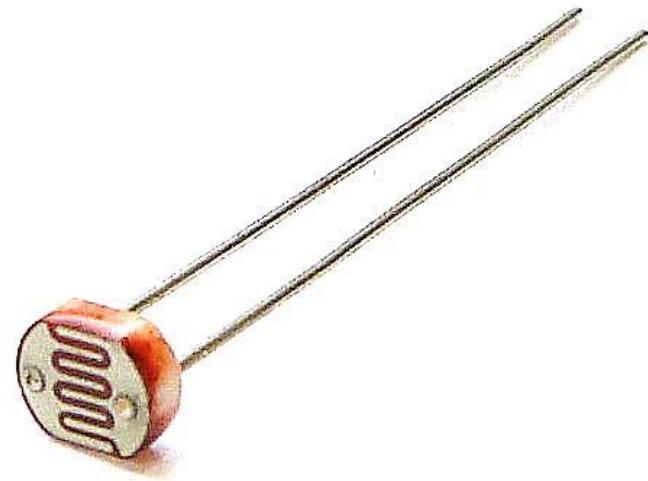
10) LED



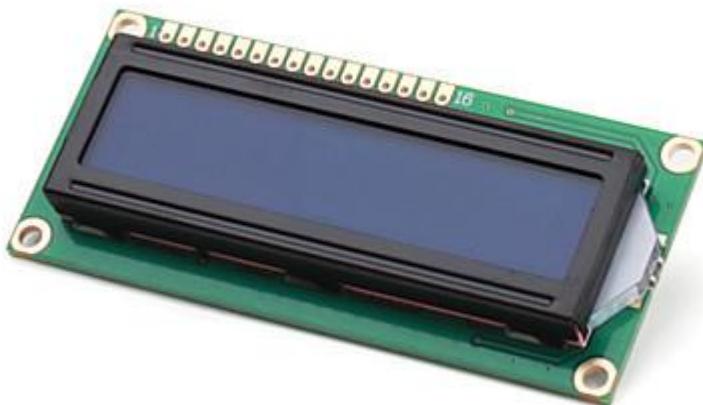
11) 버튼



12) 조도센서



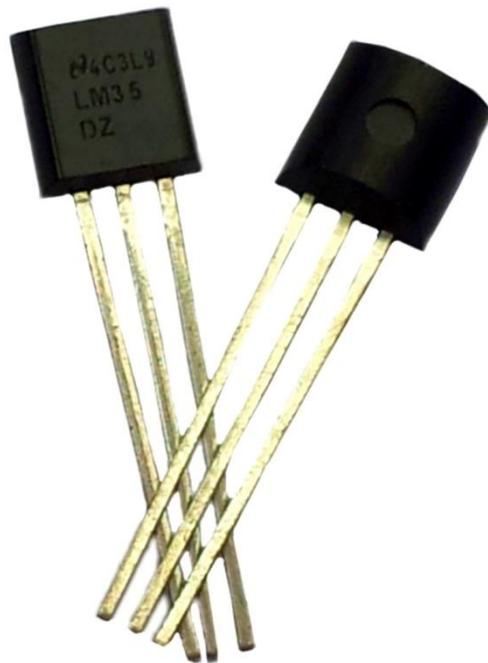
13) LCD



14) 틸트센서



15) 온도센서



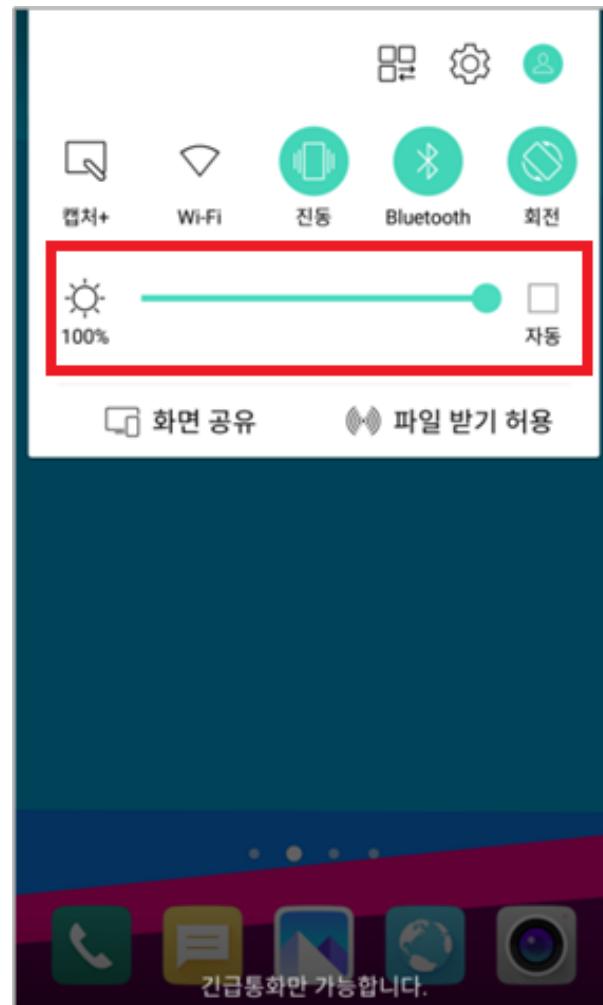
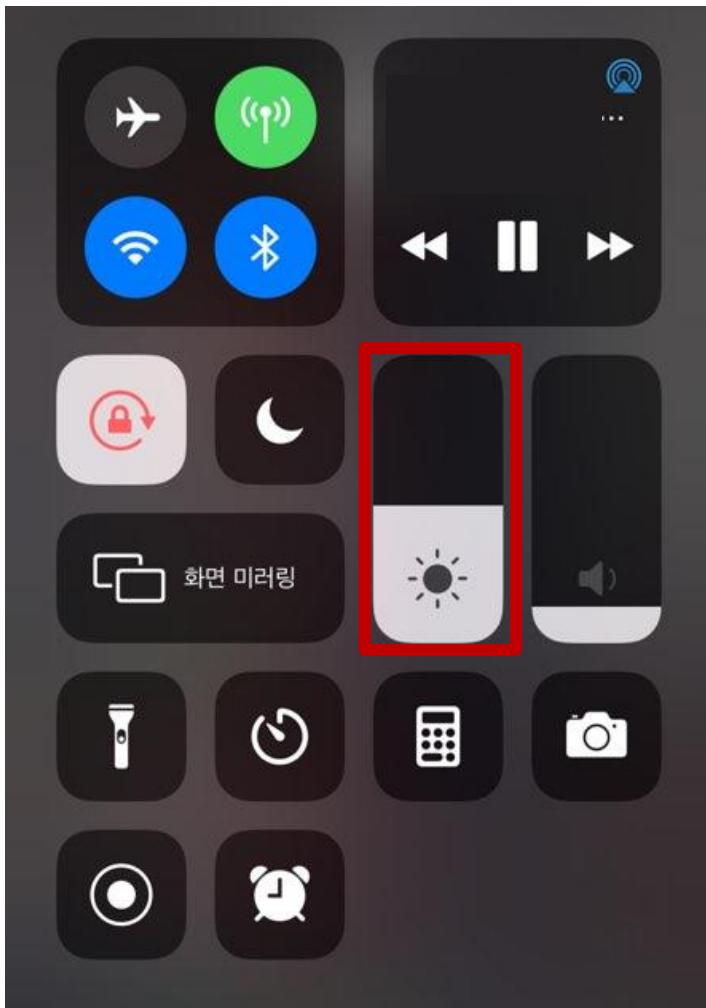
16) 피에조 센서



# 디지털 신호



# 아날로그 신호

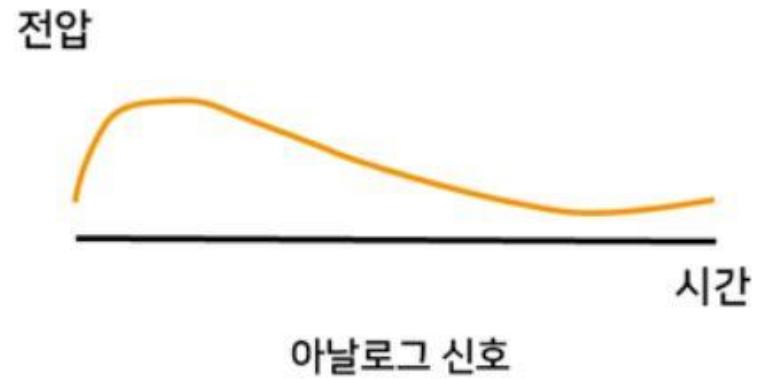


스마트인재개발원  
Smart Human Resources Development

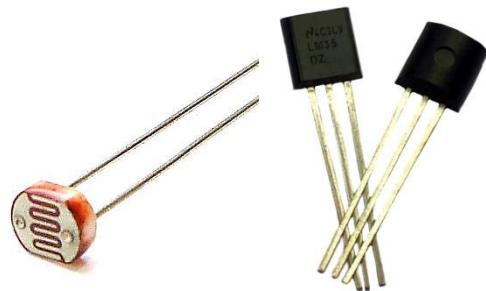
# 전기 신호



2개의 신호로 불연속적으로 변함

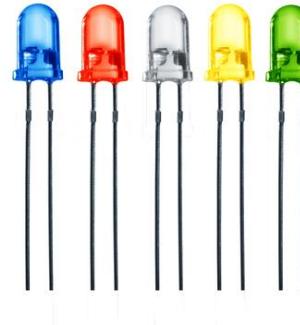


여러개의 신호로 연속적으로 변함



센서  
(Sensor)

감각하여  
알아내는 장치

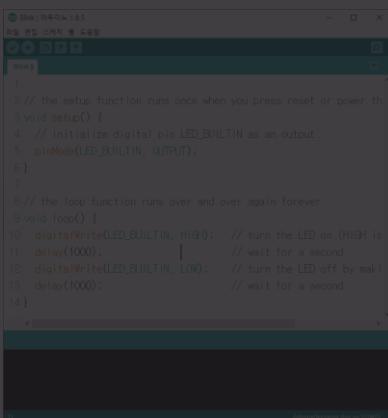


액추에이터  
(Actuator)

시스템을 움직이거나  
제어하는 기계 장치

## LED





```
1 // the setup function runs once when you press reset or power the board
2 void setup() {
3     // initialize digital pin LED_BUILTIN as an output:
4     pinMode(LED_BUILTIN, OUTPUT);
5 }
6
7 // the loop function runs over and over again forever
8 void loop() {
9     digitalWrite(LED_BUILTIN, HIGH);    // turn the LED on (HIGH is the voltage level)
10    delay(1000);                    // wait for a second
11    digitalWrite(LED_BUILTIN, LOW);   // turn the LED off by making it a low voltage level
12    delay(1000);                    // wait for a second
13}
```

액츄에이터  
원하는 명령을 통해  
제어되는 장치

업로드

우노보드



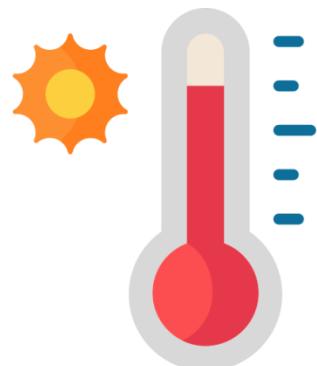
LED



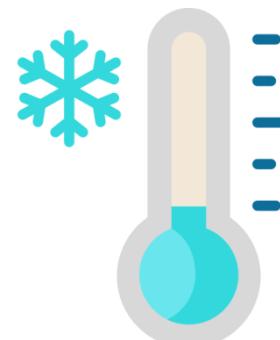
전기신호



스마트인재개발원  
Smart Human Resources Development



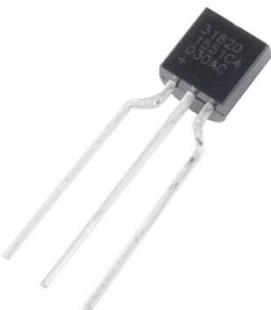
## 온도센서



# 센서 / 액츄에이터

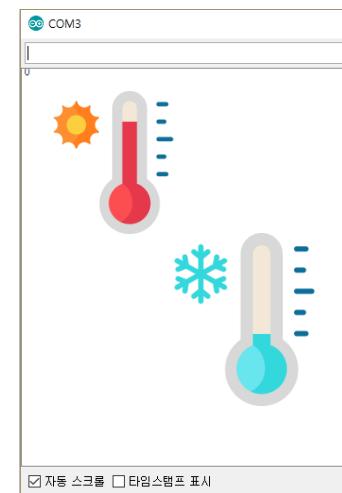
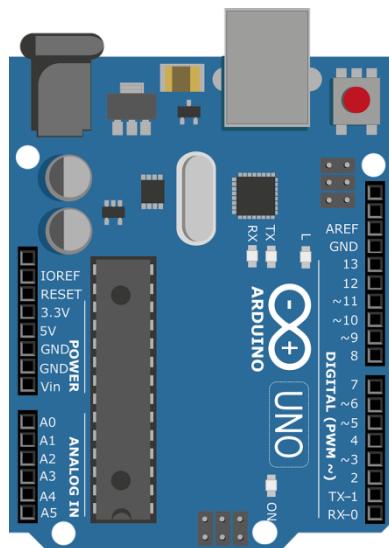


온도센서



전기신호

우노보드



스마트인재개발원  
Smart Human Resources Development

우노보드

버튼



전기신호

양, 변화를 감지하거나 구분하여  
일정한 신호로 알려주는 부품이나 기구



0 1

자동 스크롤  타임스탬프 표시

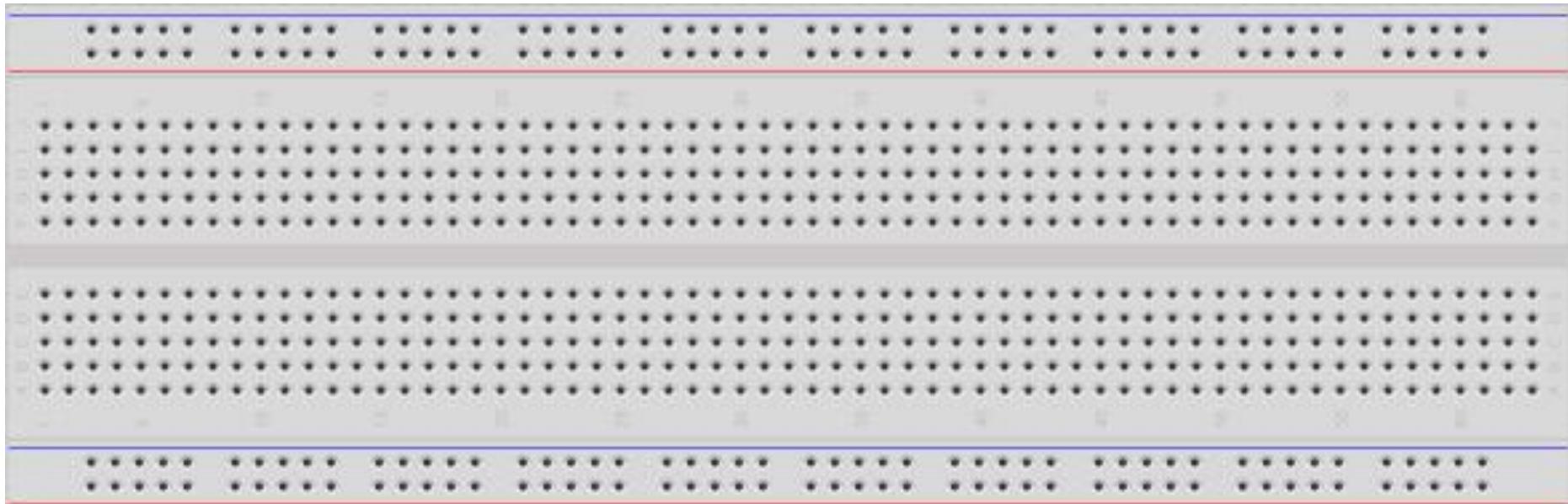


스마트인재개발원  
Smart Human Resources Development



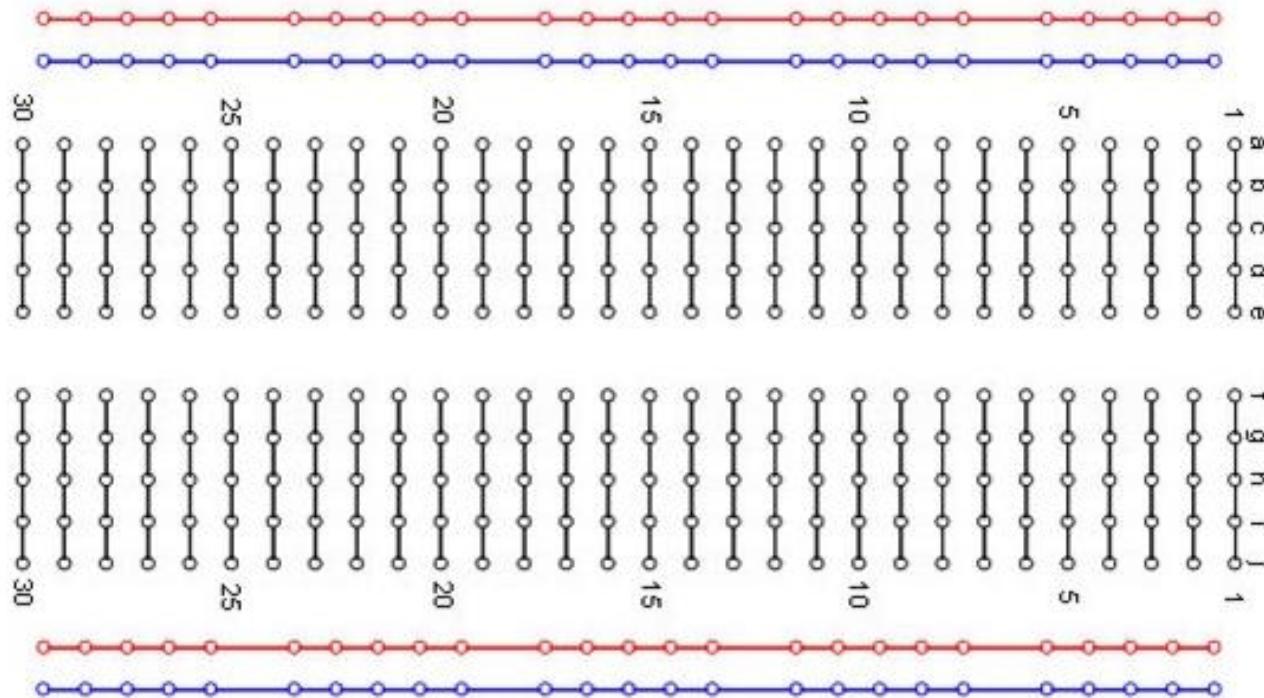
# LED

# Breadboard를 이용한 LED제어

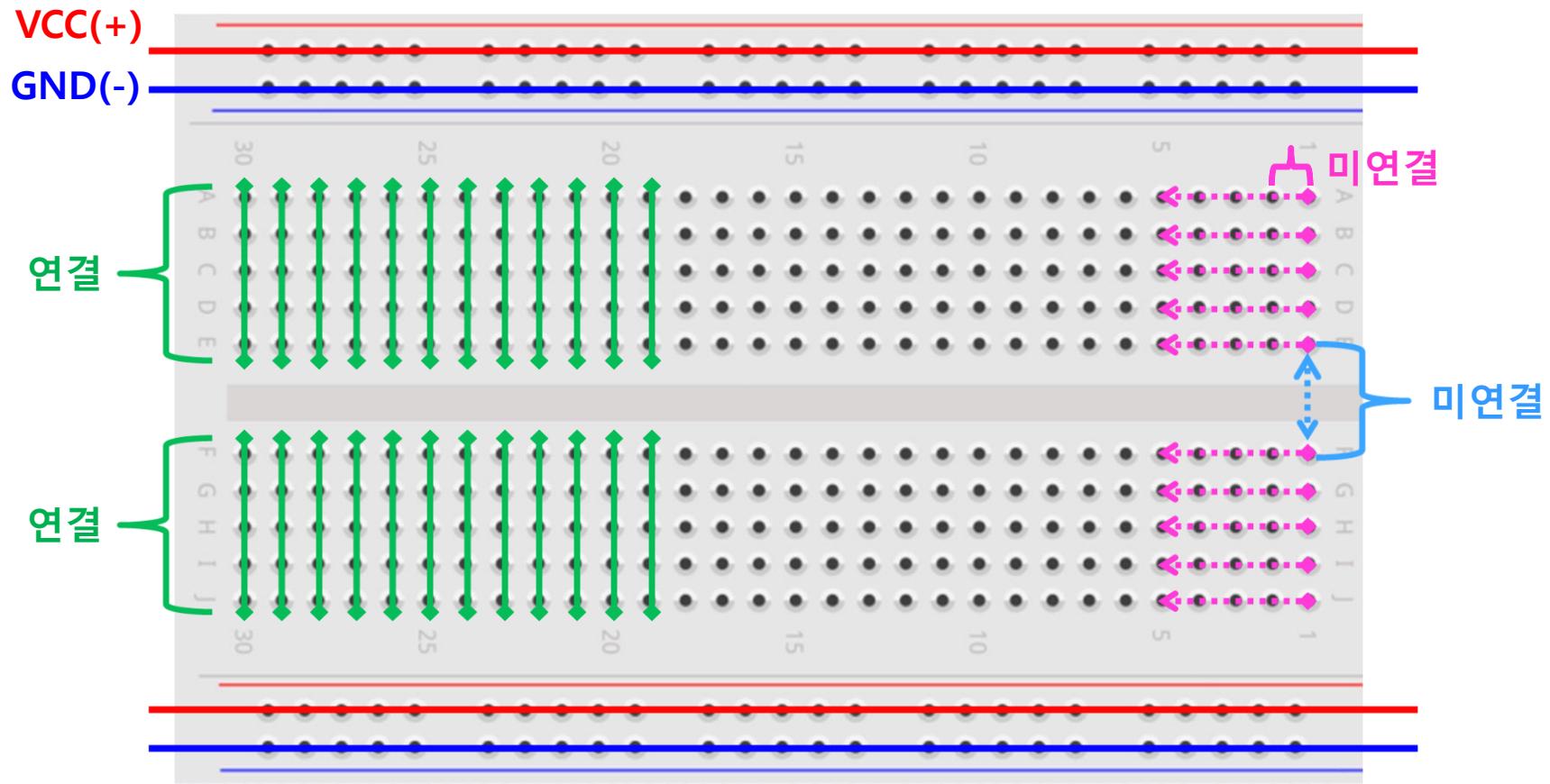


## Breadboard(빵판)

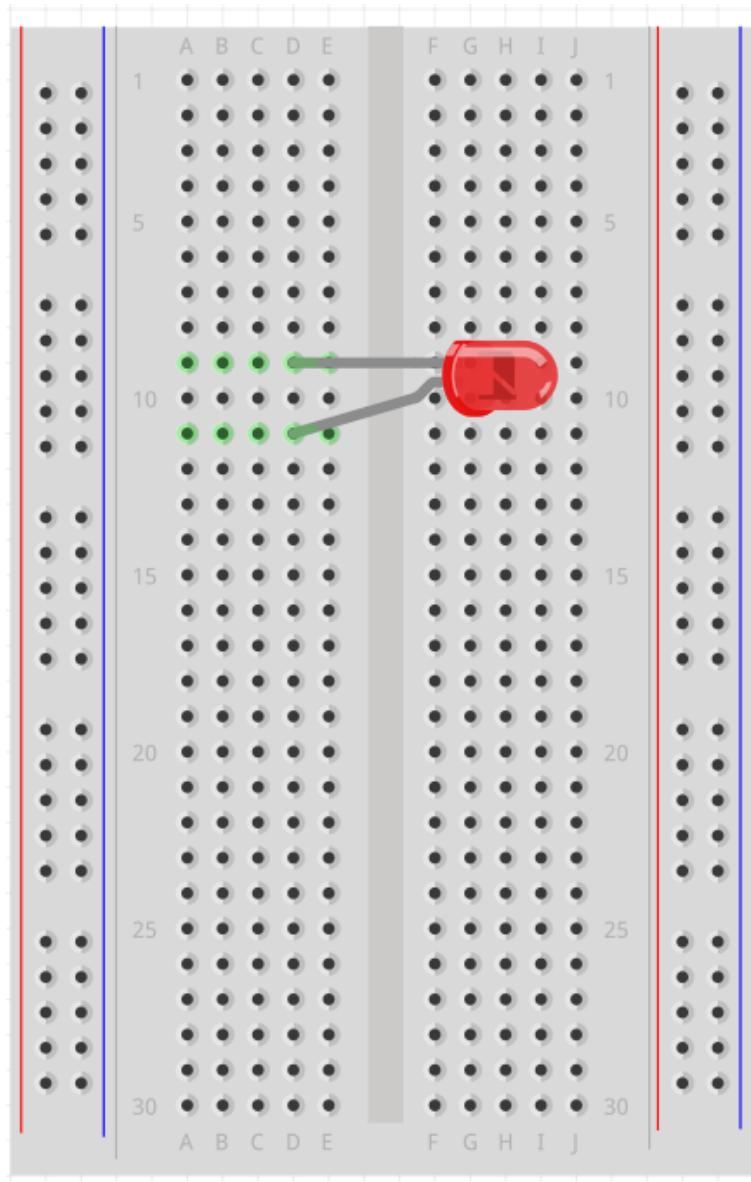
# Breadboard 구조

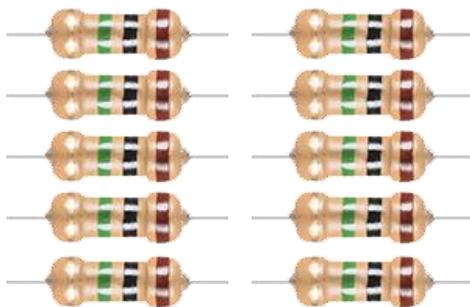


# Breadboard 구조



# Breadboard를 이용한 LED제어

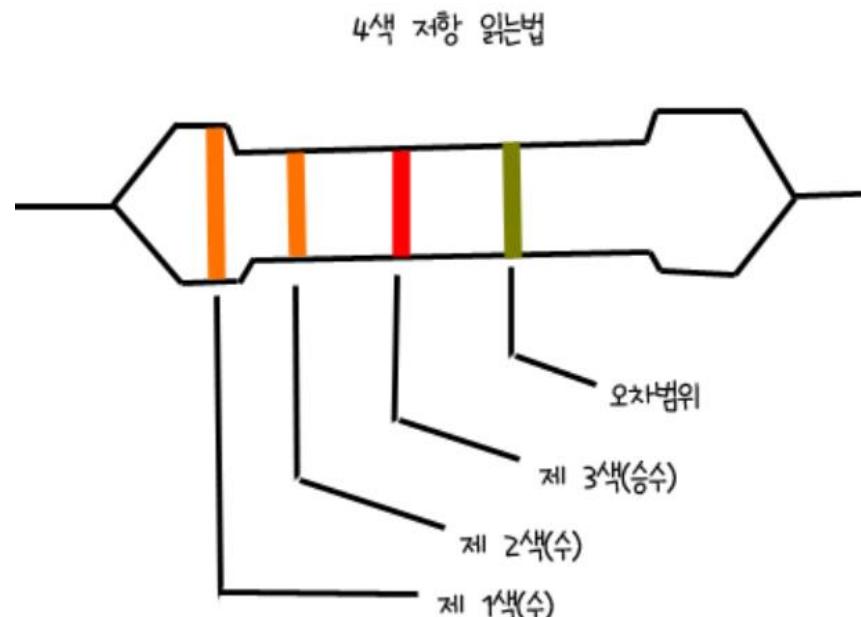




저항

## 저항(resistance)색띠읽는법

수	승수	오차
0	1	
1	10	1%
2	100	2%
3	1000	
4	10000	
5	100000	
6	1000000	
7	10000000	
8	100000000	
9	1000000000	



# LED ON - LED 사용저항

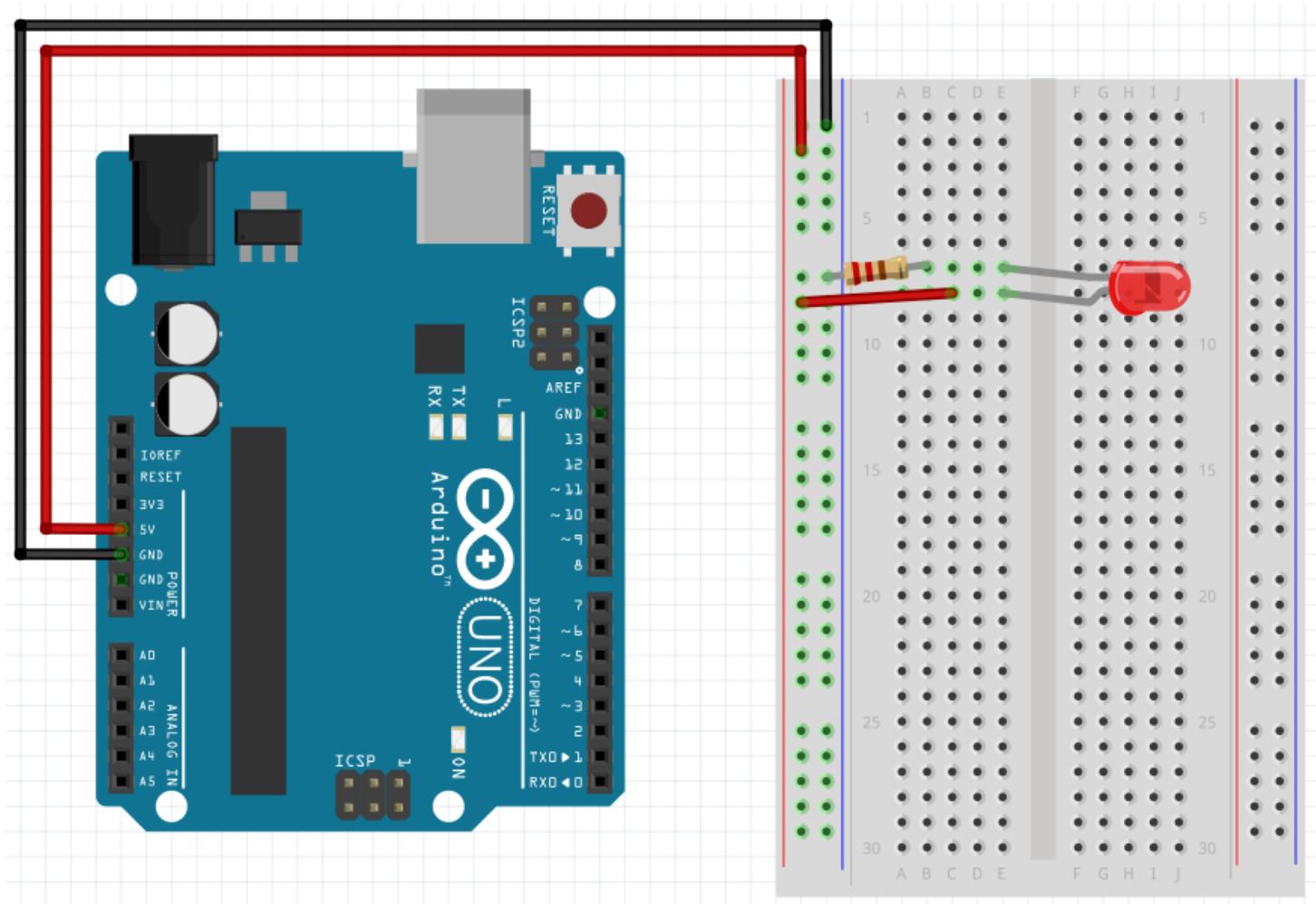


**220 ohm**



스마트인재개발원  
Smart Human Resources Development

# LED ON - LED 사용저항



스마트인재개발원  
Smart Human Resources Development

LED를 ON/OFF 제어 하려면?....

# UNO Board



reset 스위치

핀 13 LED

DIGITAL 핀

USB  
커넥터

전원 LED

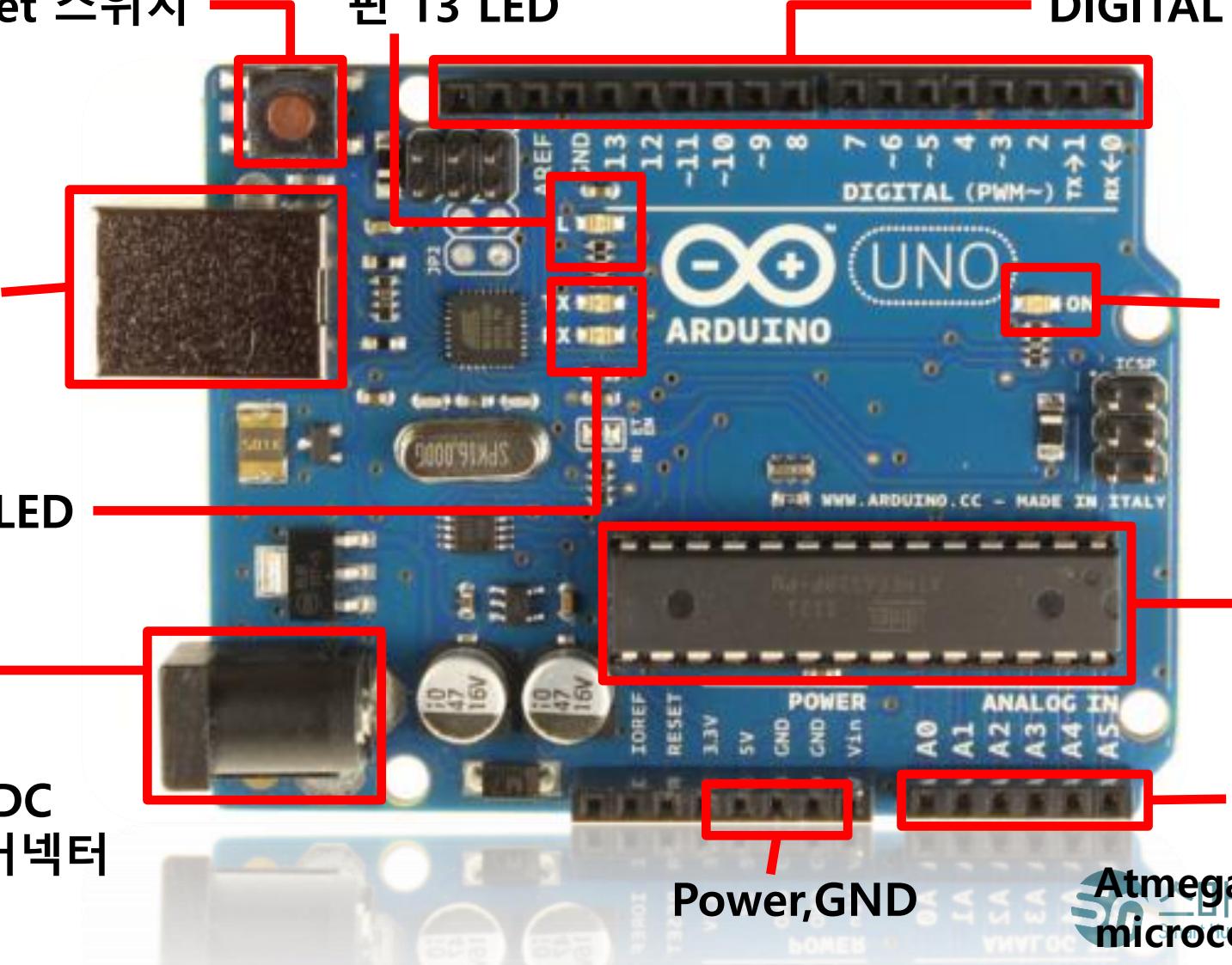
TX,RX LED

외부DC  
전원 커넥터

Power,GND

ANALOG  
입력 핀

Atmega  
microcontroller  
스마트인재개발원  
Smart Talent Development



# 아두이노 설치



# www.arduino.cc

A screenshot of the Arduino website homepage. The header features the Arduino logo, a search bar, and a sign-in button. A red box highlights the "SOFTWARE" menu item in the navigation bar. The main content area includes sections for "WHAT IS ARDUINO?", "BUY AN ARDUINO", "LEARN ARDUINO", "DONATE", "ARDUINO CREATE COMES TO CHROME OS DEVICES!", "ARDUINO UNO, THE CLASSIC ARDUINO TO GET STARTED.", and "ARDUINO EDUCATION". The "ARDUINO EDUCATION" section features the text "REDEFINING THE LEARNING EXPERIENCE ONE CLASSROOM AT A TIME".

HOME BUY SOFTWARE PRODUCTS LEARNING FORUM SUPPORT BLOG

WHAT IS ARDUINO?

ARDUINO UNO

BUY AN ARDUINO

LEARN ARDUINO

DONATE

ARDUINO CREATE COMES TO CHROME OS DEVICES!

ARDUINO UNO, THE CLASSIC ARDUINO TO GET STARTED.

ARDUINO EDUCATION

REDEFINING THE LEARNING EXPERIENCE  
ONE CLASSROOM AT A TIME

ARDUINO UNO

BLOG



스마트인재개발원  
Smart Human Resources Development

# 아두이노 설치



[HOME](#) [PRODUCTS](#) [BUY](#) [DOWNLOAD](#) [LEARNING](#) [FORUM](#) [MAKERS](#) [BLOG](#) [FOUNDATION](#)



스마트인재개발원  
Smart Human Resources Development

# 아두이노 설치



## Download



Program your hardware  
with the Arduino IDE!

Arduino provides an open-source and easy-to-use programming tool, for writing code and uploading it to your board. It is often referred to as the Arduino IDE (Integrated Development Environment).

If you are new to Arduino and don't know where to start we have prepared a getting started page with all the various steps you have to follow to install the IDE and connect your board to it.



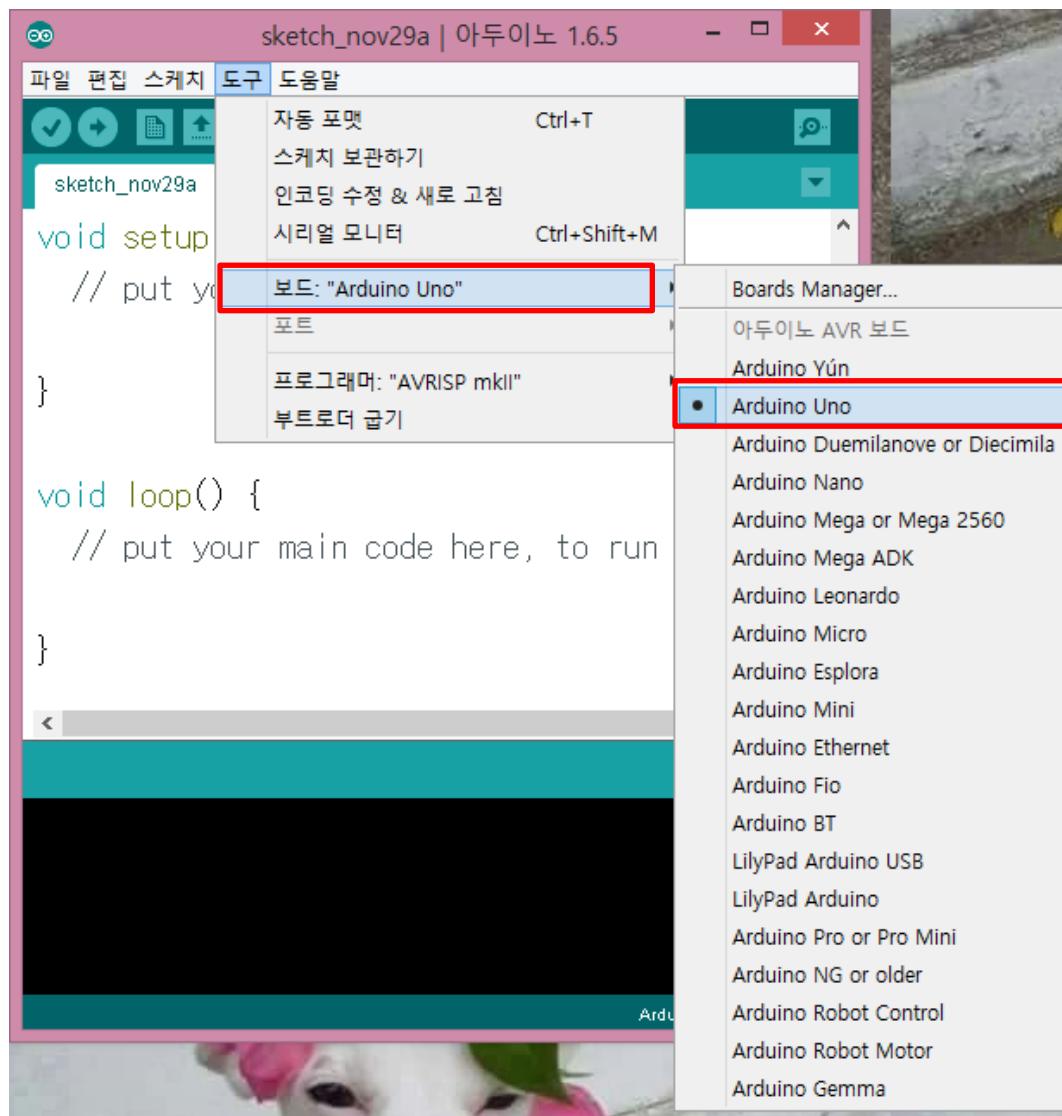
DOWNLOAD  
ARDUINO IDE

- [WINDOWS INSTALLER](#)
- [WINDOWS ZIP FILE](#)
- [MAC OS X](#)
- [LINUX 32 BITS](#)
- [LINUX 64 BITS](#)

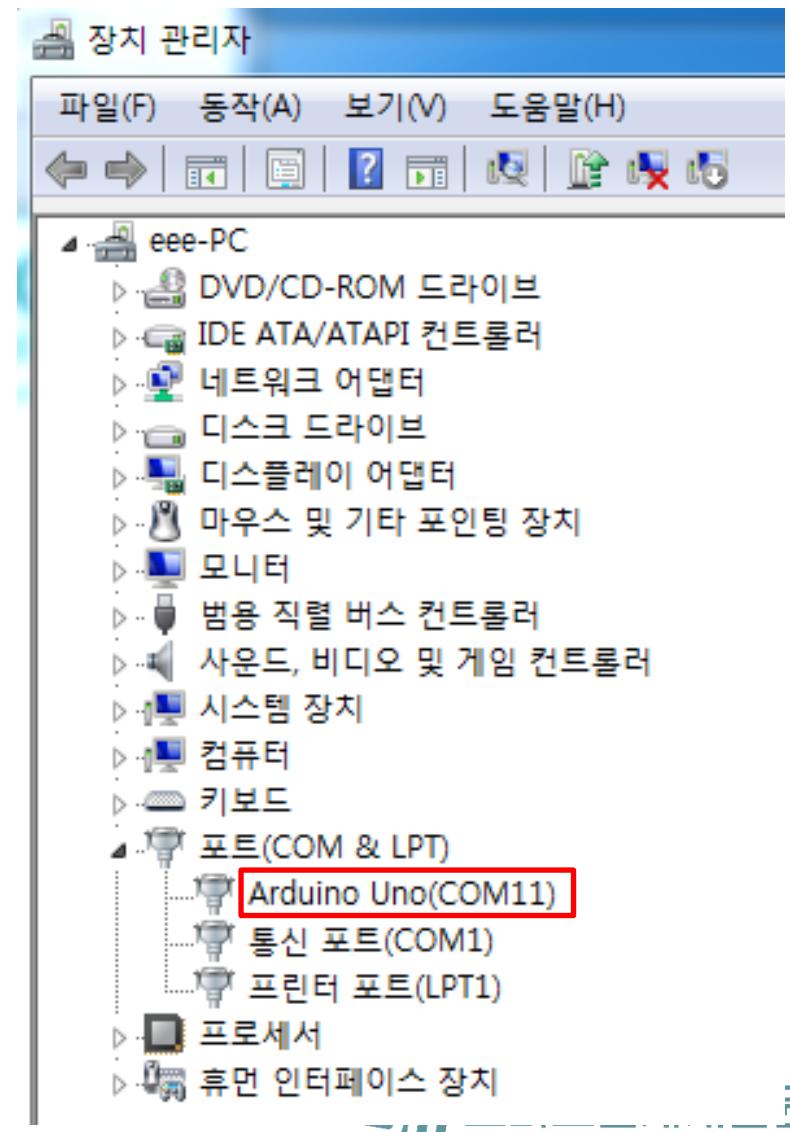
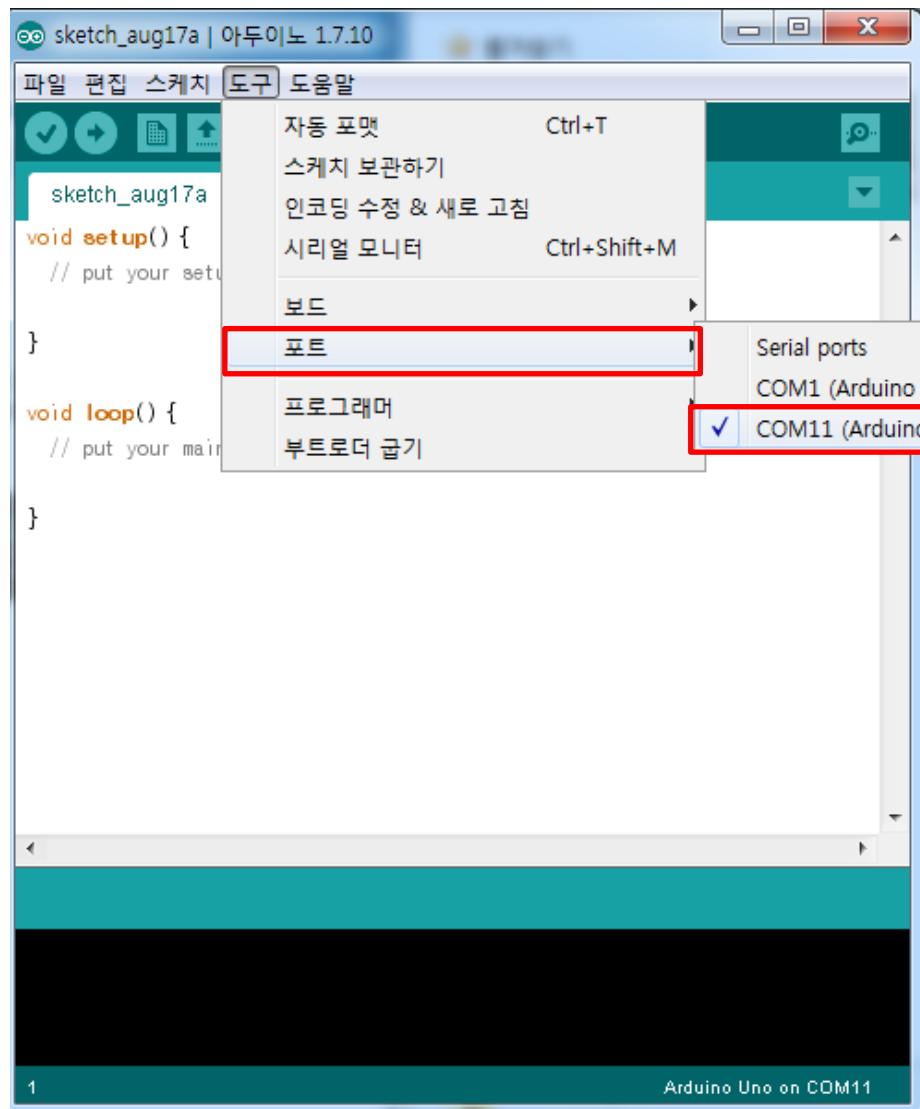


스마트인재개발원  
Smart Human Resources Development

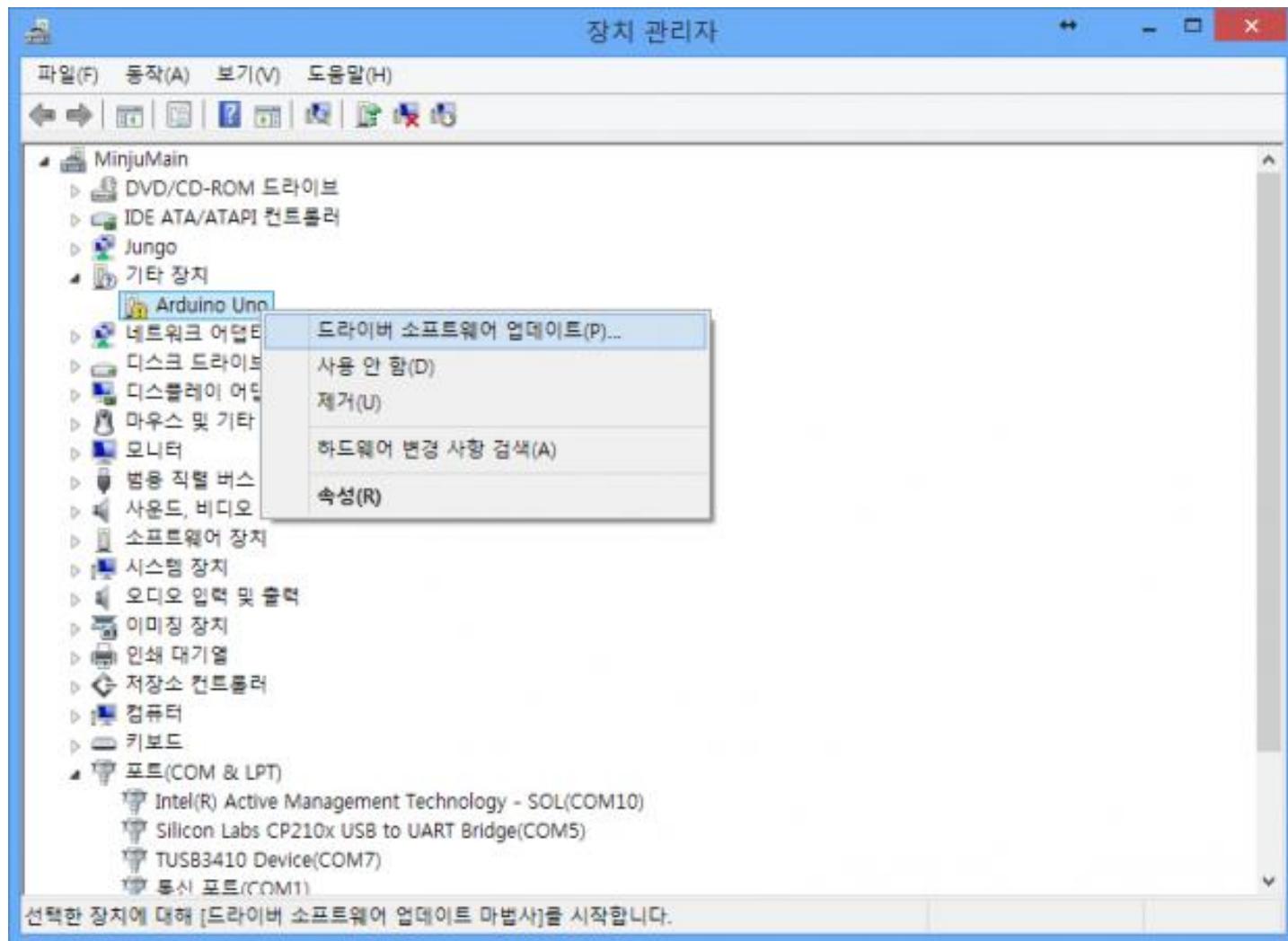
# 아두이노 설치 - 드라이버 잡기



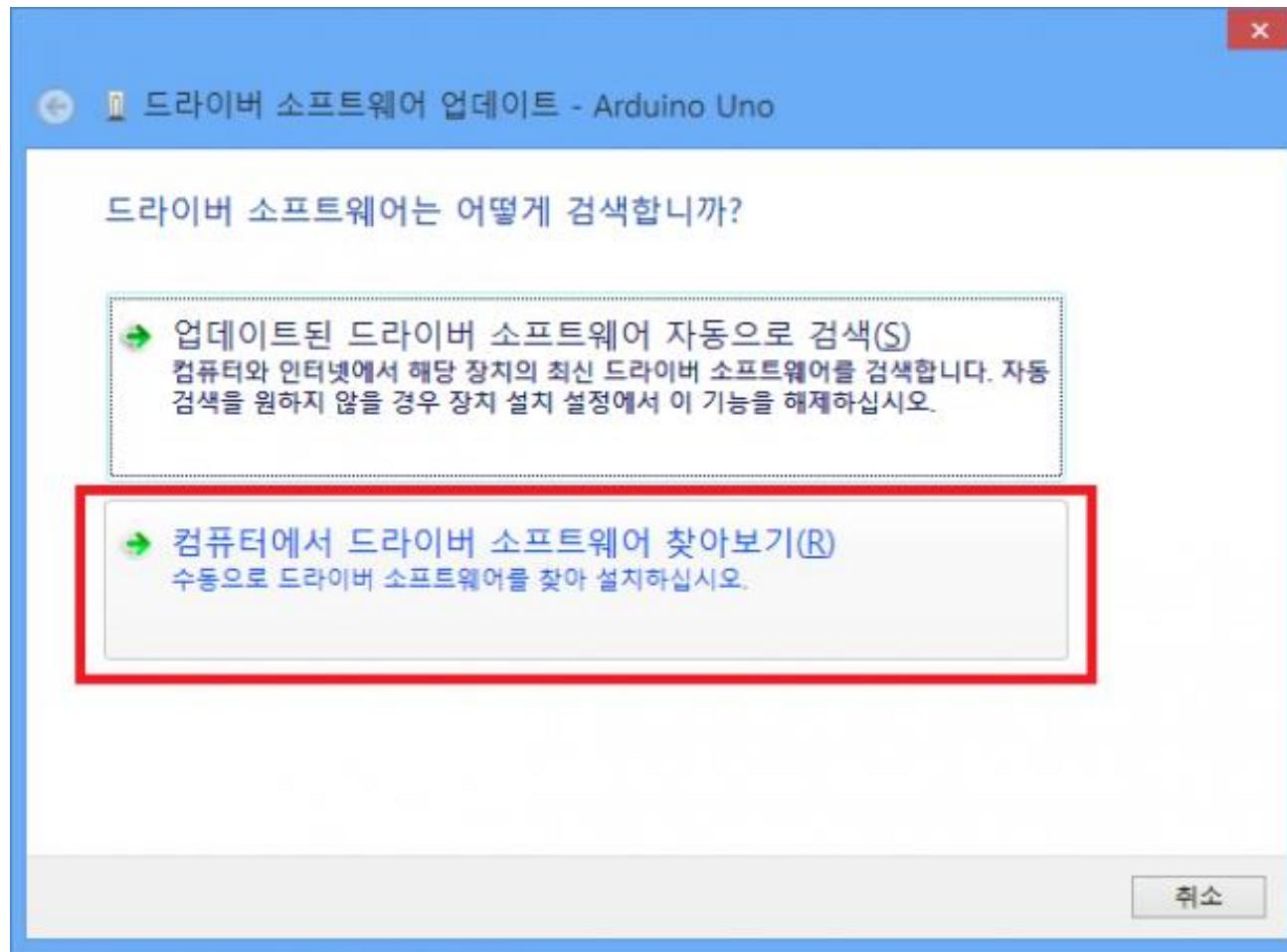
# 아두이노 설치 - 드라이버 잡기



# 아두이노 설치 - 드라이버 잡기

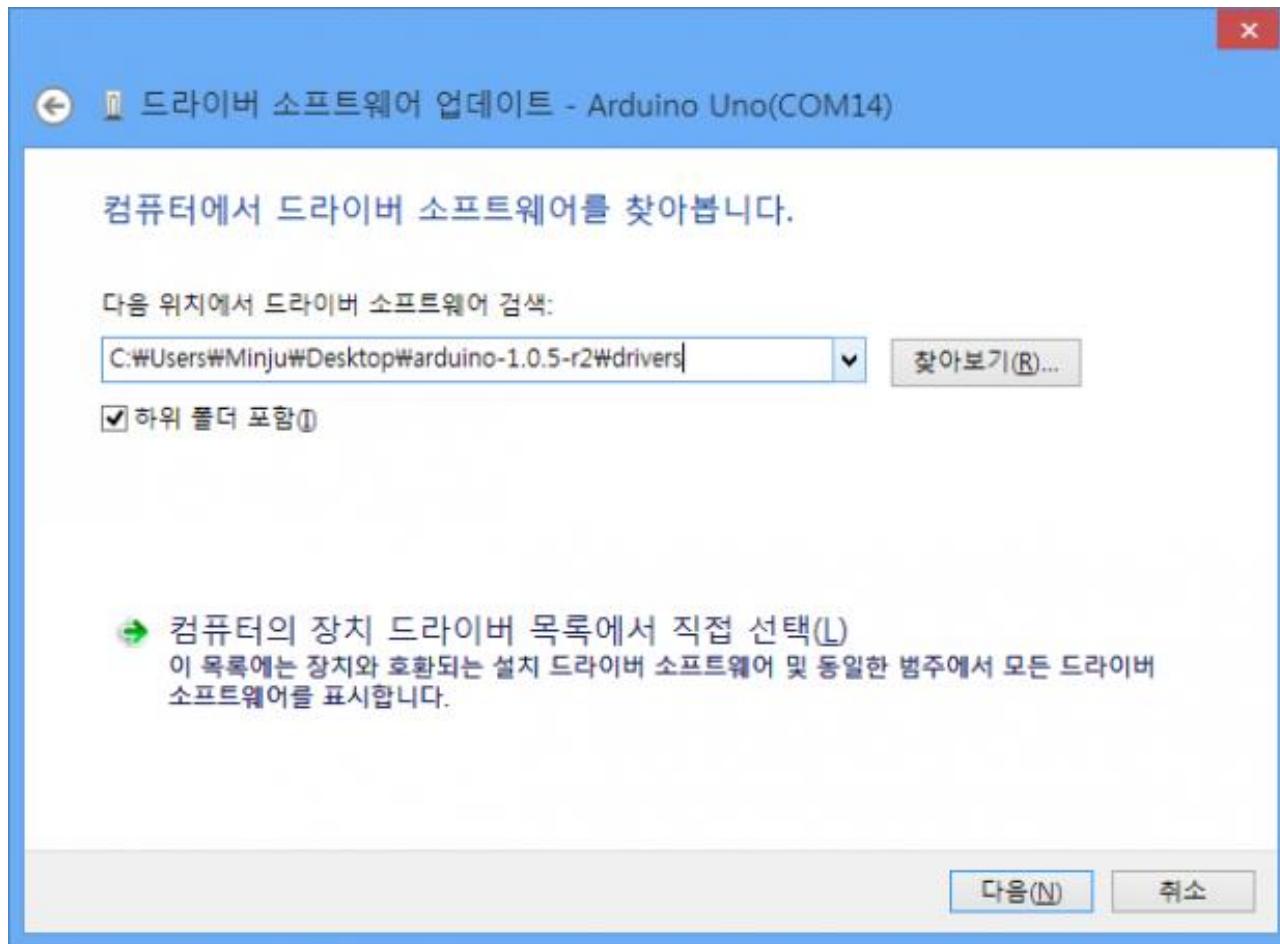


# 아두이노 설치 - 드라이버 잡기

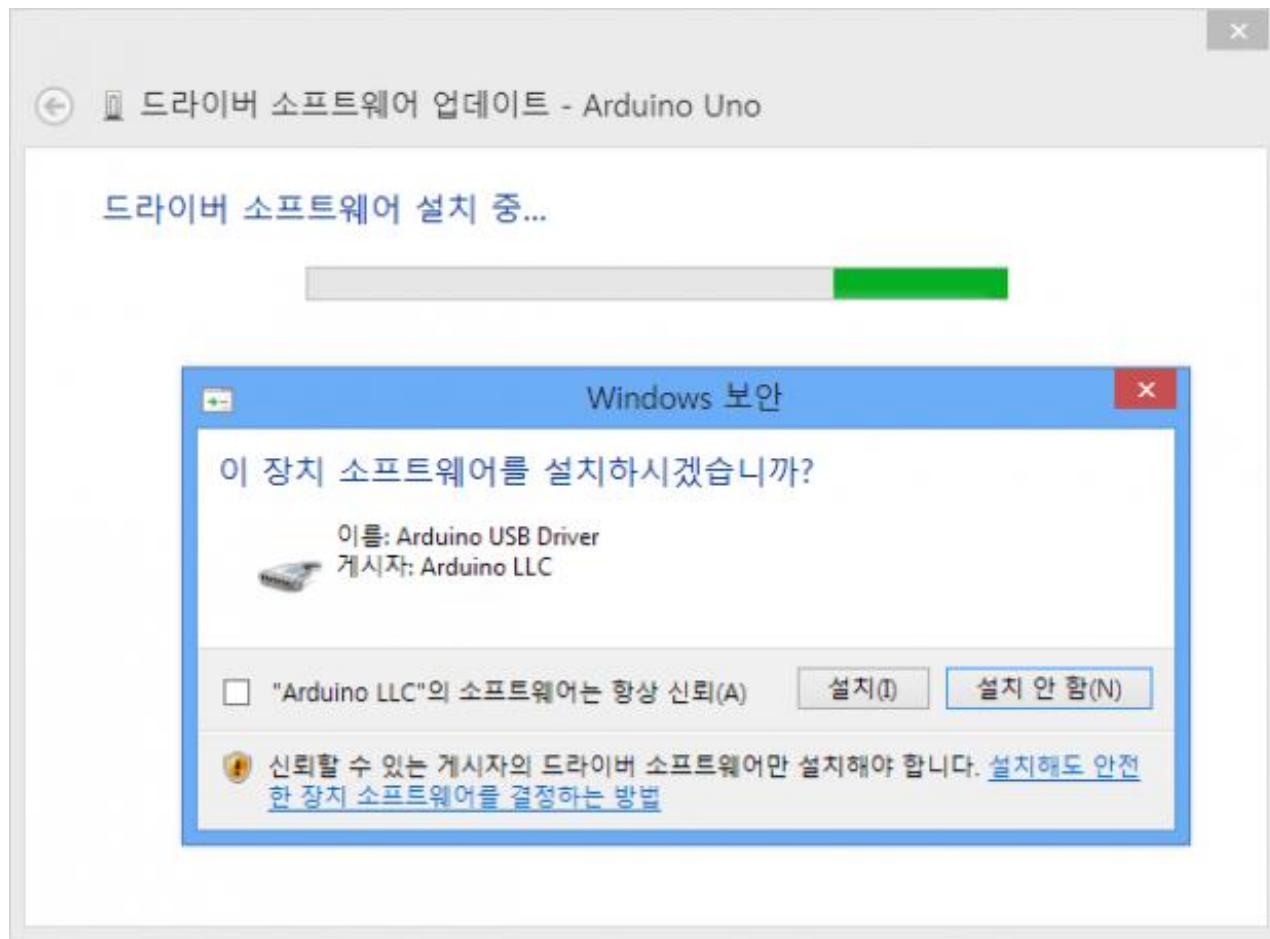


스마트인재개발원  
Smart Human Resources Development

# 아두이노 설치 - 드라이버 잡기

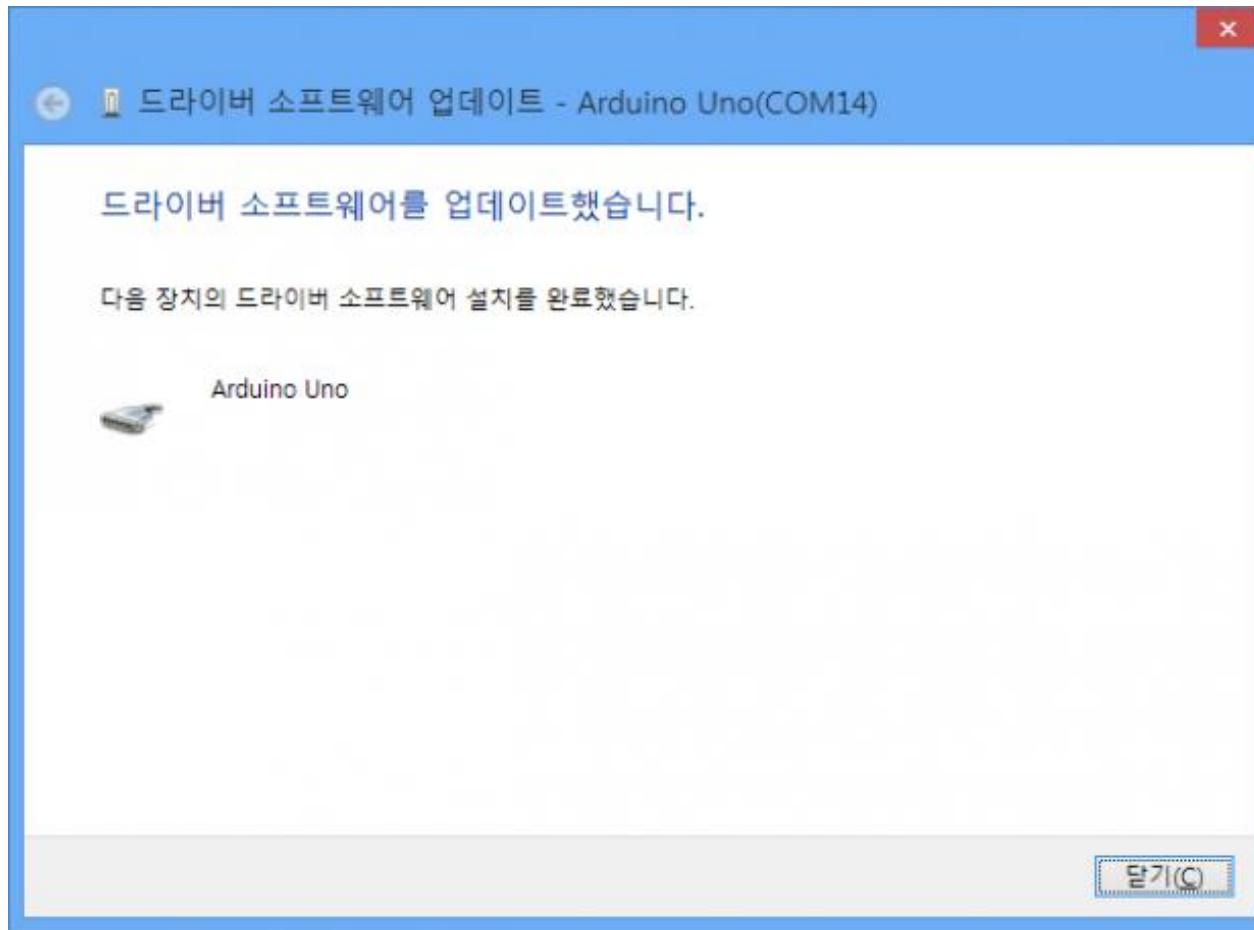


# 아두이노 설치 - 드라이버 잡기



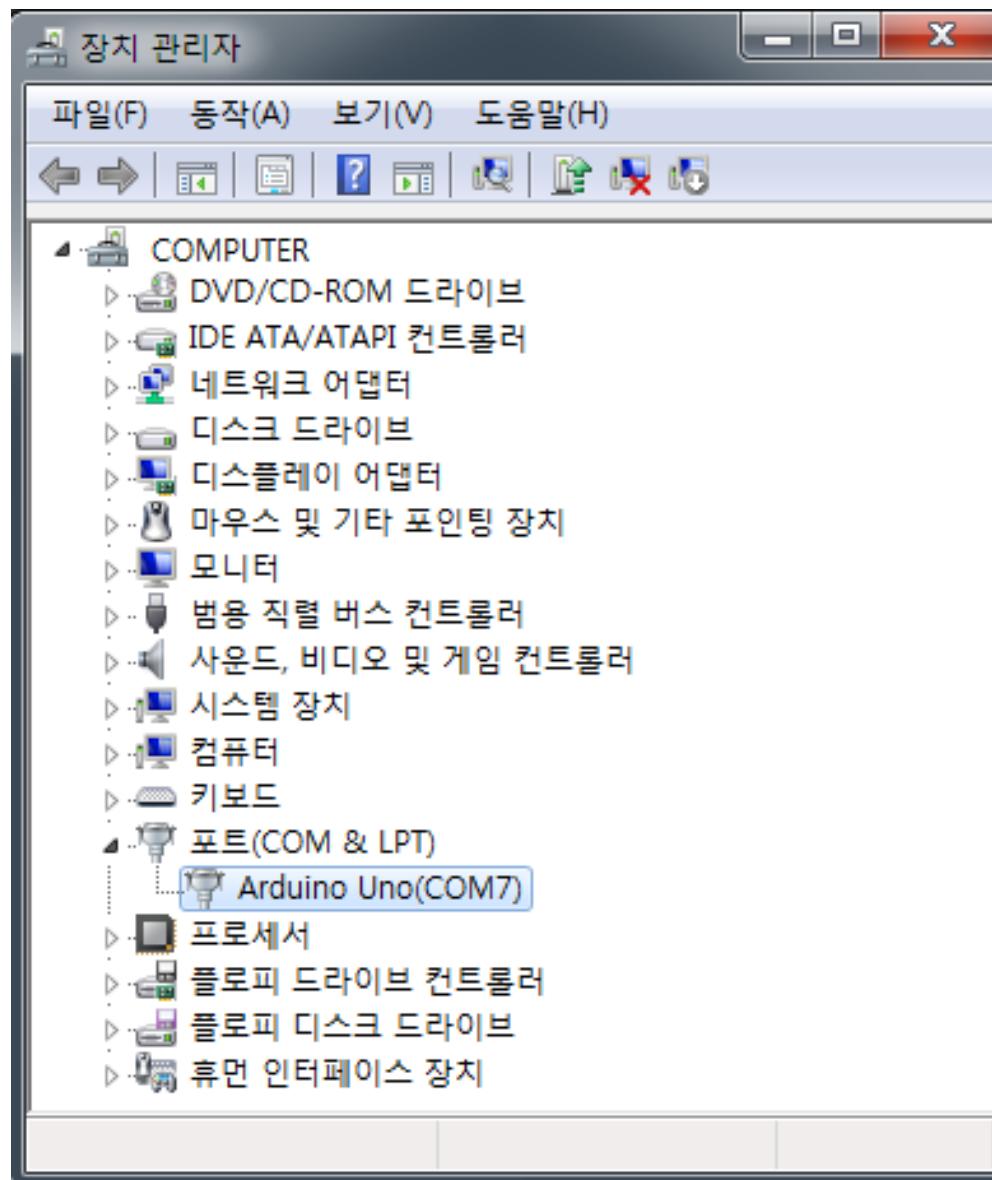
스마트인재개발원  
Smart Human Resources Development

# 아두이노 설치 - 드라이버 잡기

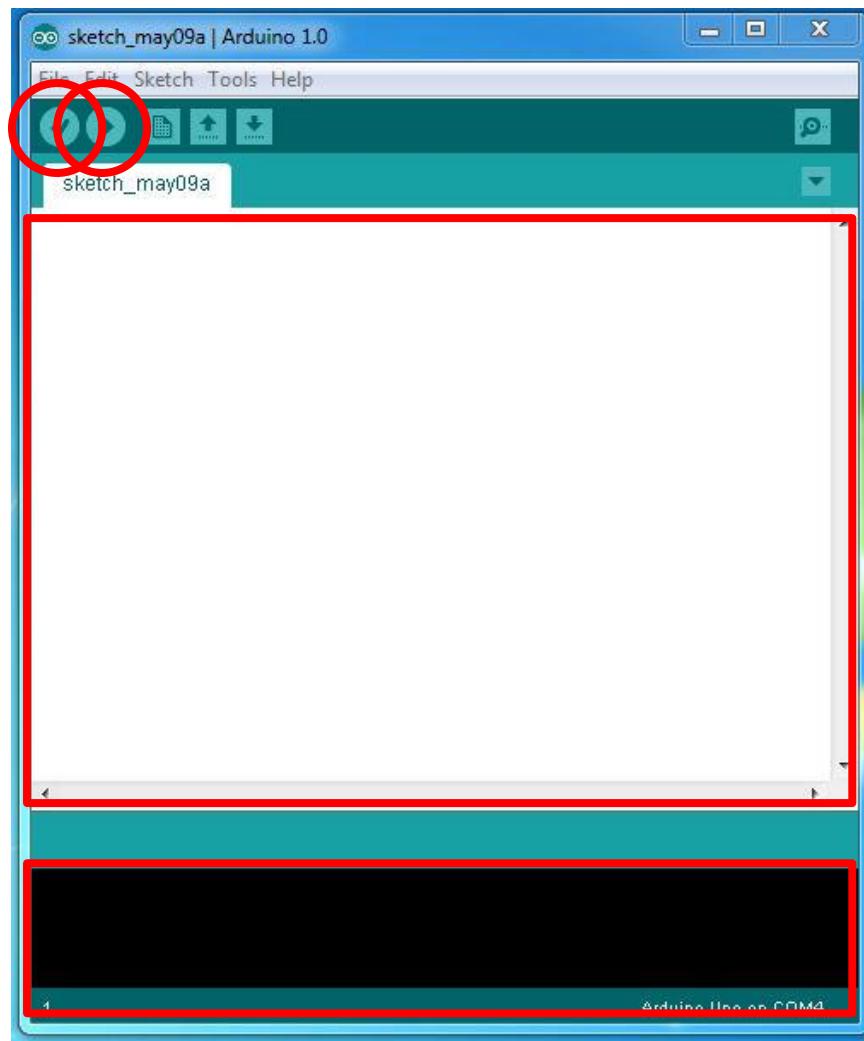


스마트인재개발원  
Smart Human Resources Development

# 아두이노 설치 - 드라이버 잡기



# 아두이노 설치



Sketch



스마트인재개발원  
Smart Human Resources Development

# 아두이노 설치



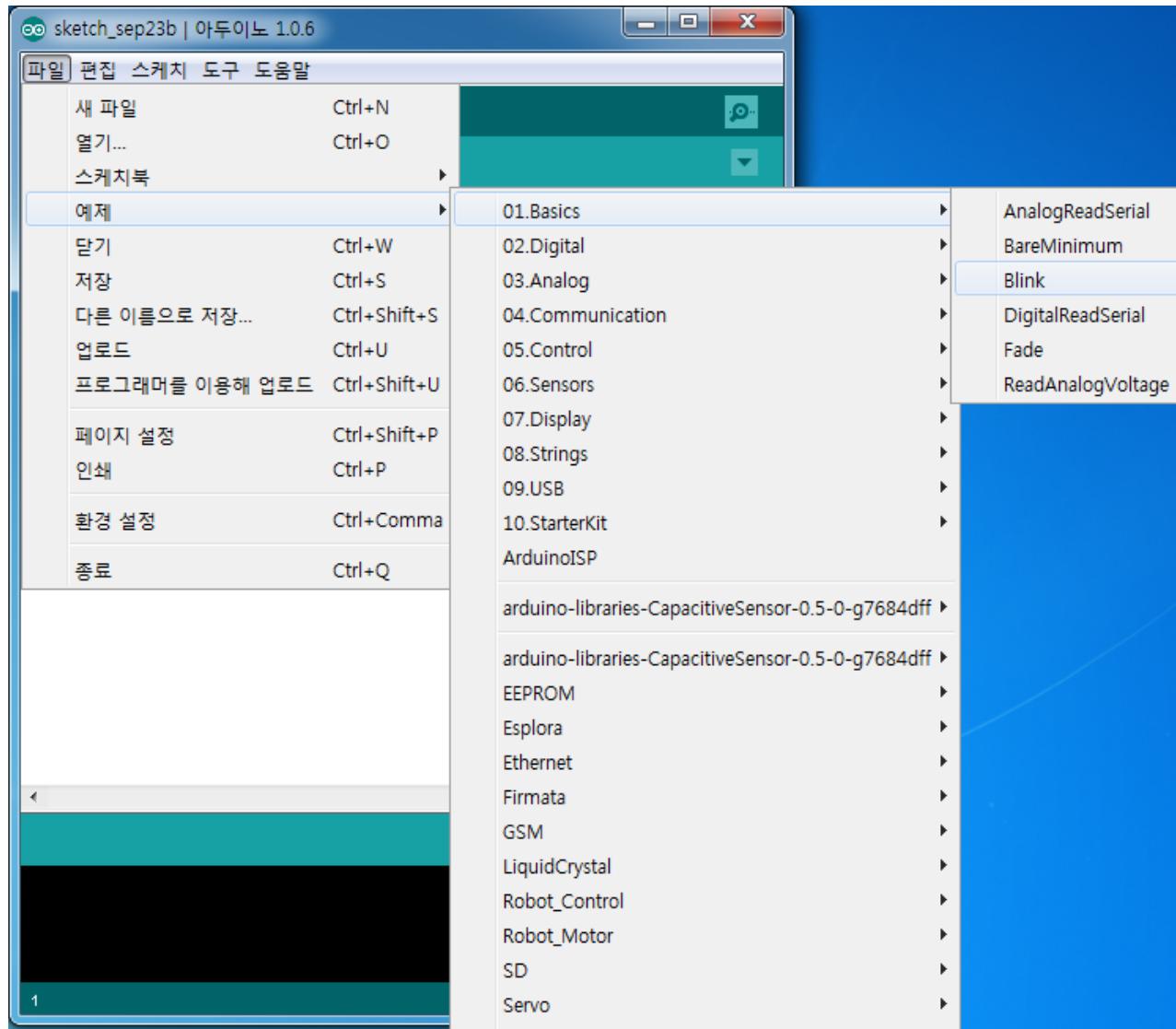
The screenshot shows the Arduino IDE interface with a sketch titled "sketch\_sep23b". The code editor displays two functions: `void setup()` and `void loop()`. The `setup()` function is enclosed in a red rectangular box, and the `loop()` function is also enclosed in a red rectangular box. A red arrow points from the `setup()` box to the text "처음 한번만 실행" (Run once). Another red arrow points from the `loop()` box to the text "반복" (Loop).

```
sketch_sep23b | 아두이노 1.0.6
파일 편집 스케치 도구 도움말
sketch_sep23b §
void setup() {
}
void loop() {
```



스마트인재개발원  
Smart Human Resources Development

# 1. LED ON/OFF - Blink



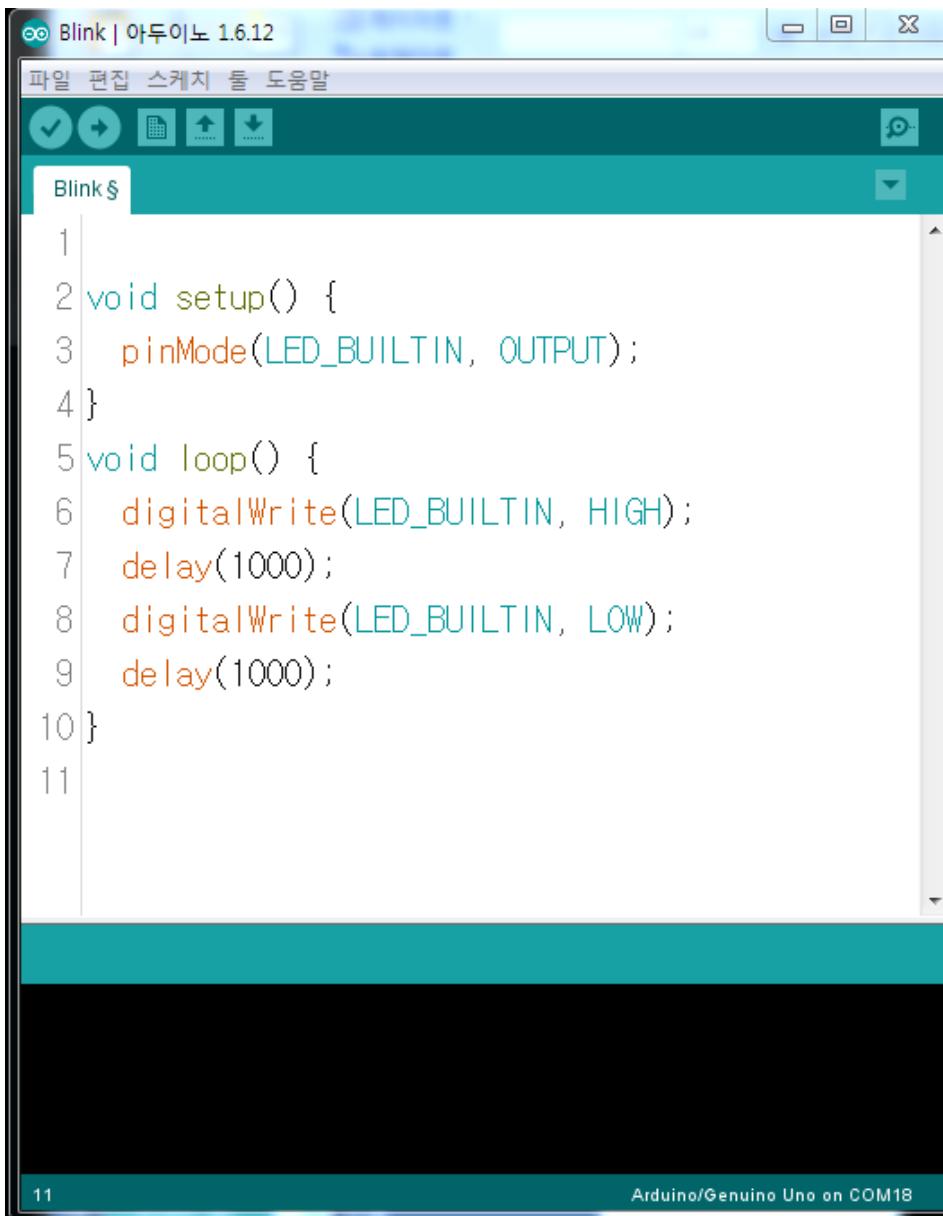
## 1. 파일

## 2. 예제

## 3. Basics

## 4. Blink

# 1. LED ON/OFF - Blink

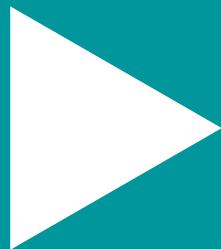


The screenshot shows the Arduino IDE interface with the title bar "Blink | 아두이노 1.6.12". The menu bar includes "파일", "편집", "스케치", "툴", and "도움말". Below the menu is a toolbar with icons for save, upload, and other functions. The main code editor window displays the following C++ code:

```
1
2 void setup() {
3   pinMode(LED_BUILTIN, OUTPUT);
4 }
5 void loop() {
6   digitalWrite(LED_BUILTIN, HIGH);
7   delay(1000);
8   digitalWrite(LED_BUILTIN, LOW);
9   delay(1000);
10}
11
```

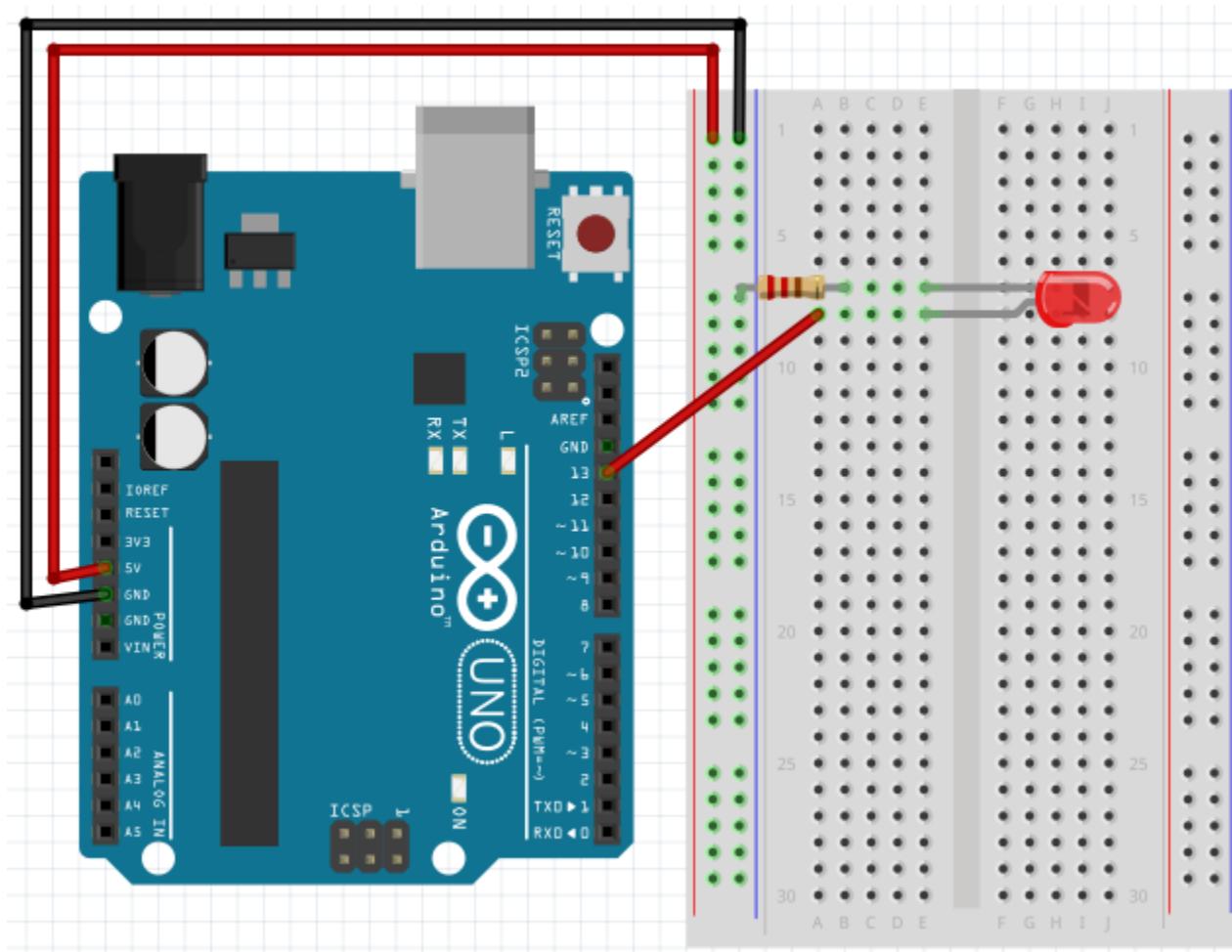
The code implements the classic Arduino "Blink" sketch, which alternates the state of the built-in LED every second.

# 1. LED ON/OFF - Blink



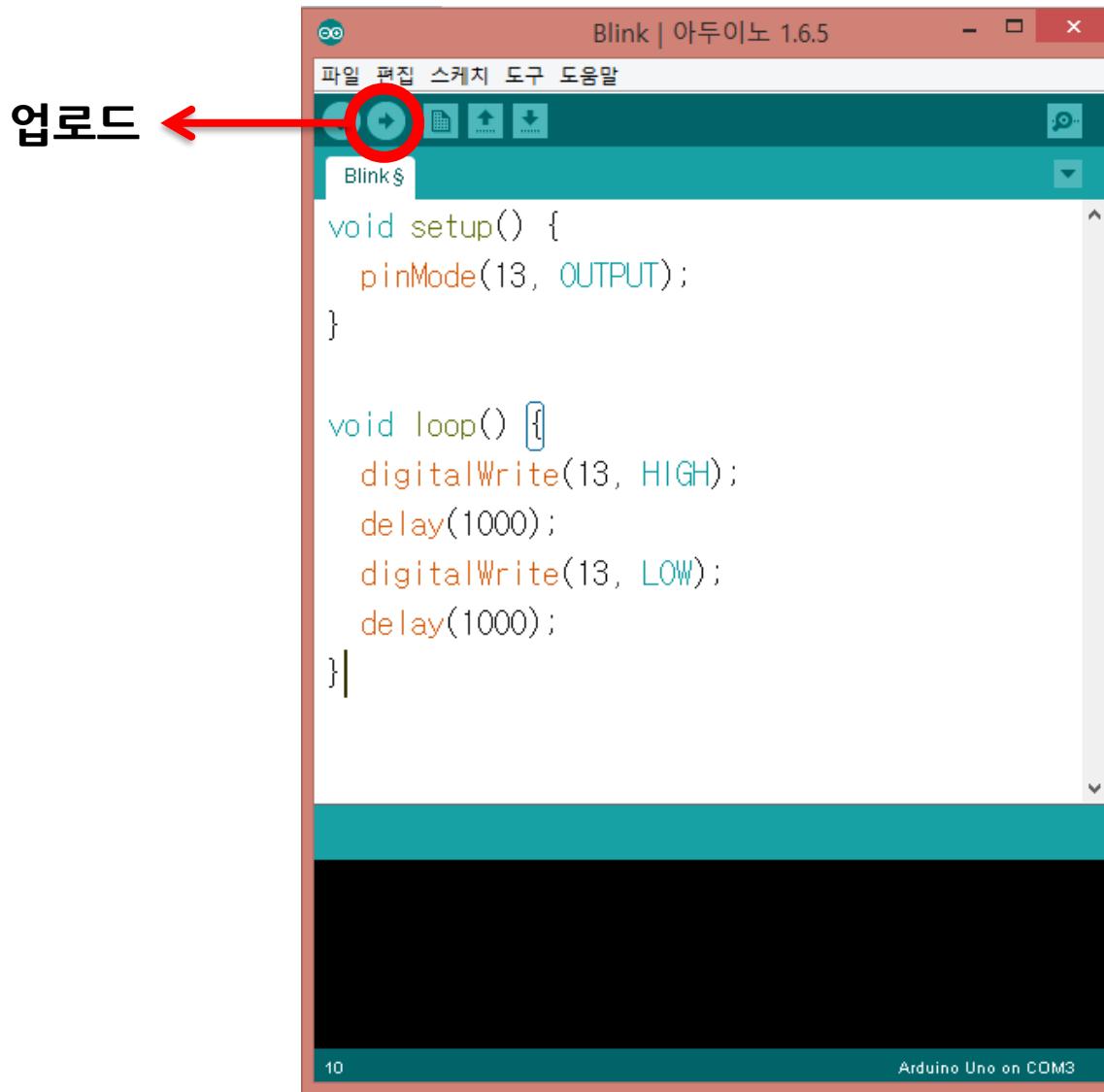
스마트인재개발원  
Smart Human Resources Development

# 1. LED ON/OFF - Blink 배선도



스마트인재개발원  
Smart Human Resources Development

# 1. LED ON/OFF - Blink



업로드 ←

```
void setup() {  
    pinMode(13, OUTPUT);  
}  
  
void loop() {  
    digitalWrite(13, HIGH);  
    delay(1000);  
    digitalWrite(13, LOW);  
    delay(1000);  
}
```

10 Arduino Uno on COM3

# 1. LED ON/OFF - Blink



- **pinMode(13, OUTPUT);**

연결된 pin번호

입·출력 설정

- **digitalWrite(13, HIGH);**

연결된 pin번호

LED ON / 1값

- **digitalWrite(13, LOW);**

연결된 pin번호

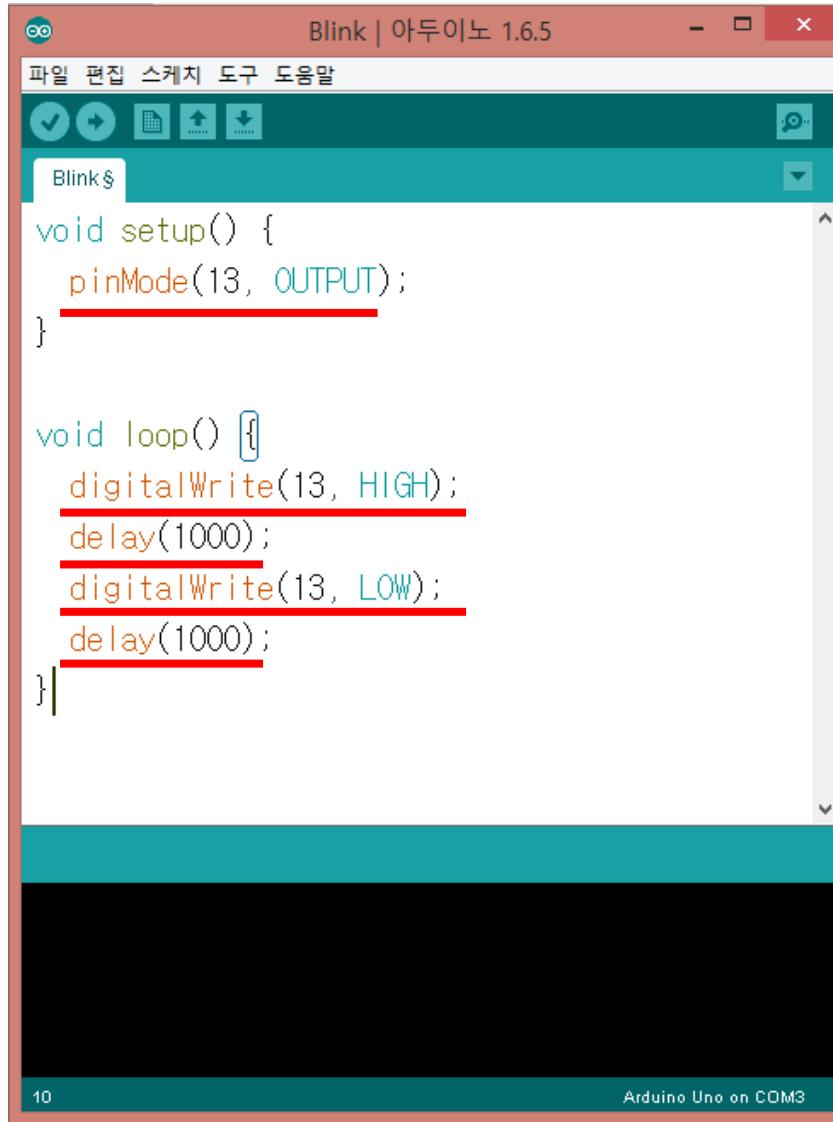
LED OFF / 0값

- **delay(1000);** 1초 지연 시키다



스마트인재개발원  
Smart Human Resources Development

# 1. LED ON/OFF - Blink 코드



The screenshot shows the Arduino IDE interface with the title bar "Blink | 아두이노 1.6.5". The menu bar includes "파일", "편집", "스케치", "도구", and "도움말". Below the menu is a toolbar with icons for file operations. The code editor contains the following sketch:

```
void setup() {
    pinMode(13, OUTPUT);
}

void loop() {
    digitalWrite(13, HIGH);
    delay(1000);
    digitalWrite(13, LOW);
    delay(1000);
}
```

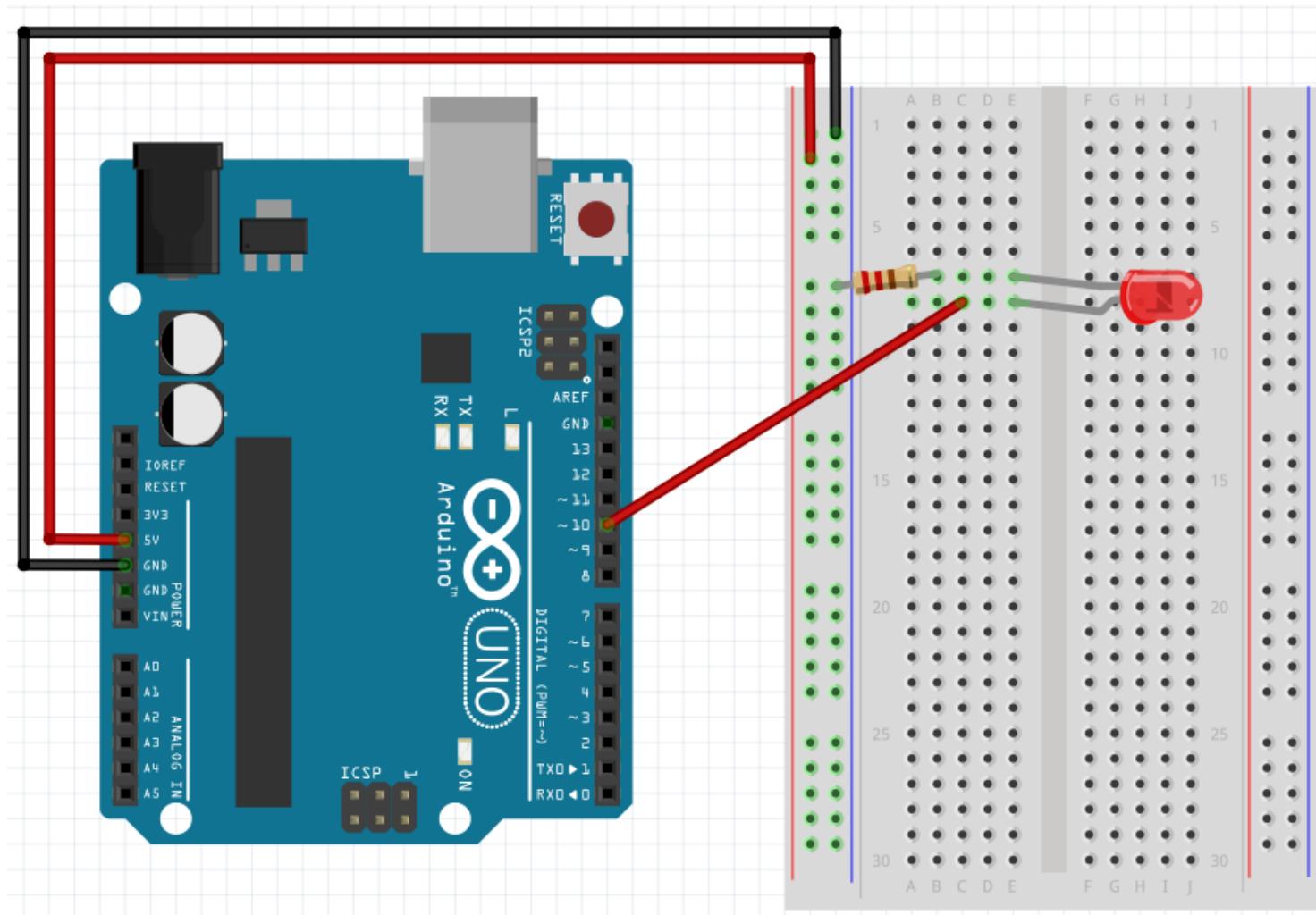
The code uses the `pinMode`, `digitalWrite`, and `delay` functions to control pin 13. The `loop` function alternates between HIGH and LOW states every second. The code is highlighted in blue and orange, and specific lines are underlined in red.

# 1. LED ON/OFF - Blink

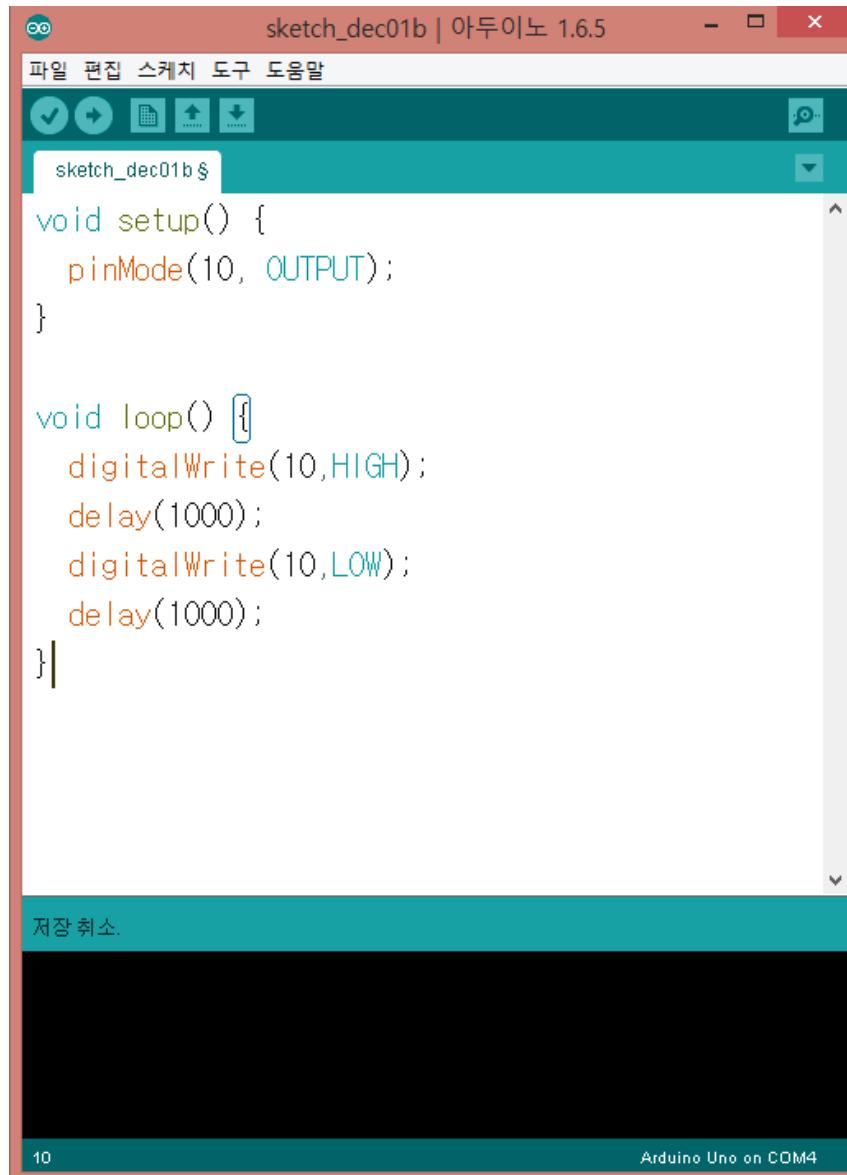


10번 핀을 이용해서 LED 제어

# 1. LED ON/OFF - Blink 배선도



# 1. LED ON/OFF - Blink 코드



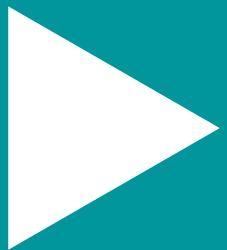
The screenshot shows the Arduino IDE interface with the title bar "sketch\_dec01b | 아두이노 1.6.5". The menu bar includes "파일", "편집", "스케치", "도구", and "도움말". The code editor contains the following sketch:

```
void setup() {
  pinMode(10, OUTPUT);
}

void loop() {
  digitalWrite(10,HIGH);
  delay(1000);
  digitalWrite(10,LOW);
  delay(1000);
}
```

The status bar at the bottom indicates "저장 취소." (Save canceled) and "Arduino Uno on COM4".

## 2. LED ON/OFF X 3



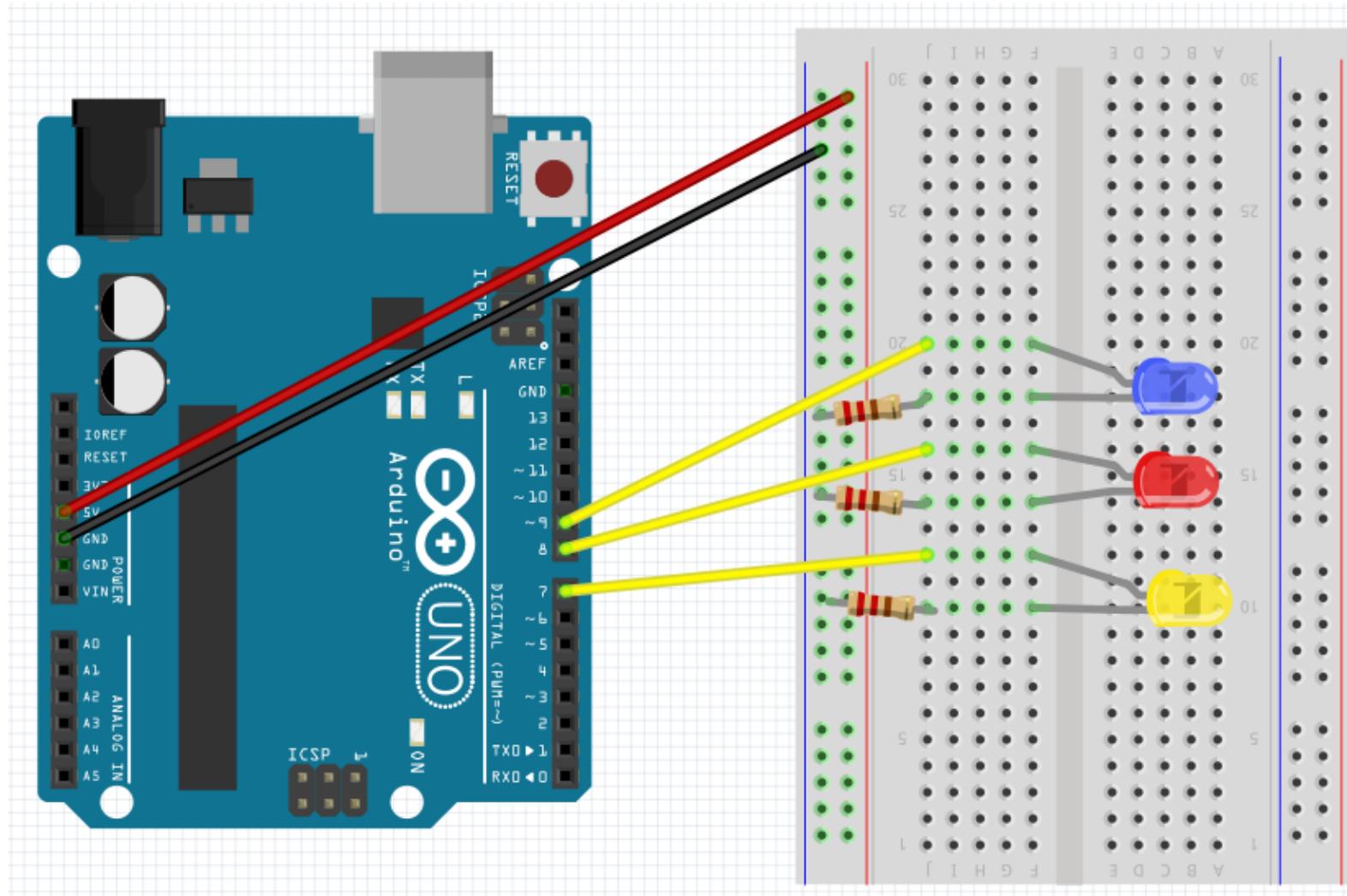
스마트인재개발원  
Smart Human Resources Development

## 2. LED ON/OFF X 3



- Arduino Board
- LED X 3
- 220-ohm resistor X 3

## 2. LED ON/OFF X 3 - 배선도



## 2. LED ON/OFF X 3 - 코드



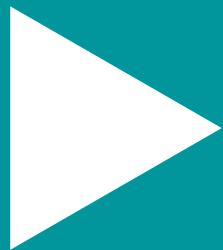
```
void setup() {  
    pinMode(7, OUTPUT);  
    pinMode(8, OUTPUT);  
    pinMode(9, OUTPUT);  
}
```

```
void loop() {  
    digitalWrite(7, HIGH);  
    delay(1000);  
    digitalWrite(7, LOW);  
    delay(1000);  
    digitalWrite(8, HIGH);  
    delay(1000);  
    digitalWrite(8, LOW);  
    delay(1000);  
    digitalWrite(9, HIGH);  
    delay(1000);  
    digitalWrite(9, LOW);  
    delay(1000);  
}
```



스마트인재개발원  
Smart Human Resources Development

# 3. LED ON/OFF2



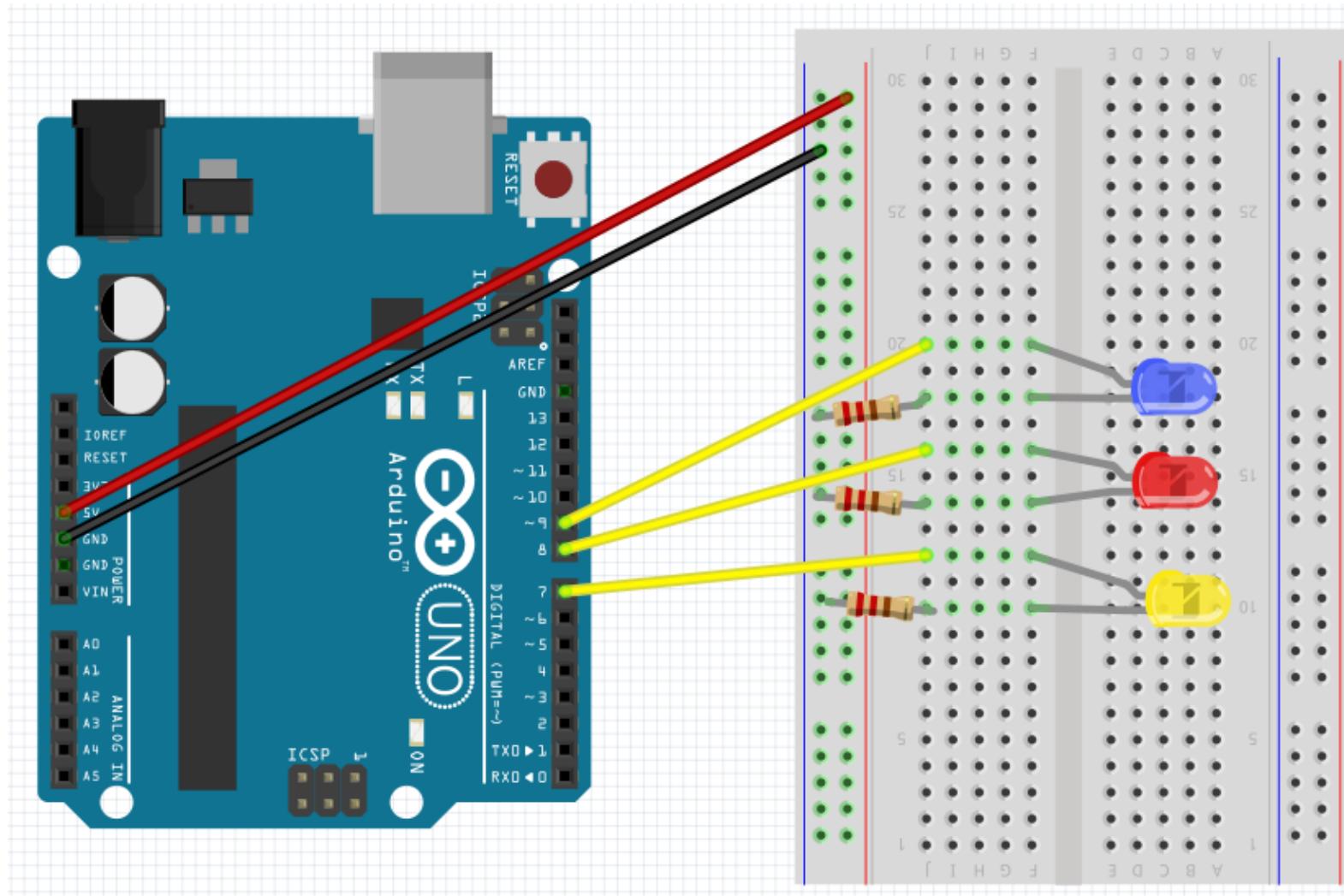
스마트인재개발원  
Smart Human Resources Development

### 3. LED ON/OFF2



- Arduino Board
- LED X 3
- 220-ohm resistor X 3

### 3. LED ON/OFF2 - 배선도



### 3. LED ON/OFF2 - 코드



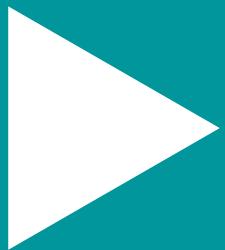
```
void setup() {  
    pinMode(7, OUTPUT);  
    pinMode(8, OUTPUT);  
    pinMode(9, OUTPUT);  
}  
}
```

```
void loop() {  
    digitalWrite(7, HIGH);  
    delay(1000);  
    digitalWrite(8, HIGH);  
    delay(1000);  
    digitalWrite(9, HIGH);  
    delay(1000);  
    digitalWrite(7, LOW);  
    digitalWrite(8, LOW);  
    digitalWrite(9, LOW);  
    delay(1000);  
}  
}
```



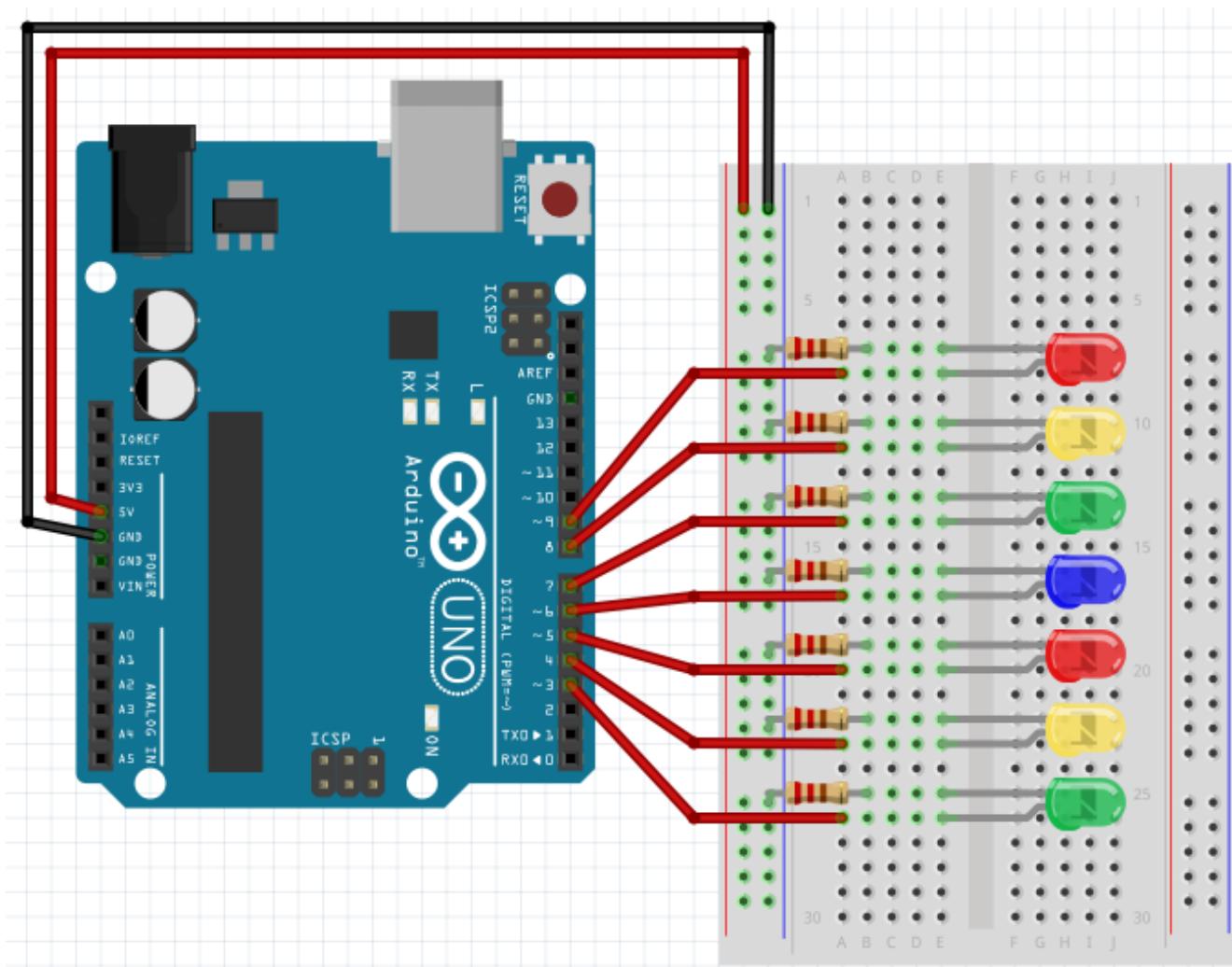
스마트인재개발원  
Smart Human Resources Development

# 4. LED제어 Wave



스마트인재개발원  
Smart Human Resources Development

# 4. LED제어 Wave - 배선도



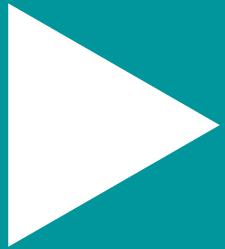
# 4. LED제어 Wave - 코드



```
void setup() {  
    pinMode(3, OUTPUT);  
    pinMode(4, OUTPUT);  
    pinMode(5, OUTPUT);  
    pinMode(6, OUTPUT);  
    pinMode(7, OUTPUT);  
    pinMode(8, OUTPUT);  
    pinMode(9, OUTPUT);  
}  
  
void loop() {  
    digitalWrite(3, HIGH);  
    delay(300);  
    digitalWrite(3, LOW);  
    digitalWrite(4, HIGH);  
    delay(300);  
    digitalWrite(4, LOW);  
    digitalWrite(5, HIGH);  
    delay(300);  
    digitalWrite(5, LOW);  
    digitalWrite(6, HIGH);  
    delay(300);  
    digitalWrite(6, LOW);  
    digitalWrite(7, HIGH);  
    delay(300);  
    digitalWrite(7, LOW);  
    digitalWrite(8, HIGH);  
    delay(300);  
    digitalWrite(8, LOW);  
    digitalWrite(9, HIGH);  
    delay(300);  
    digitalWrite(9, LOW);  
}
```

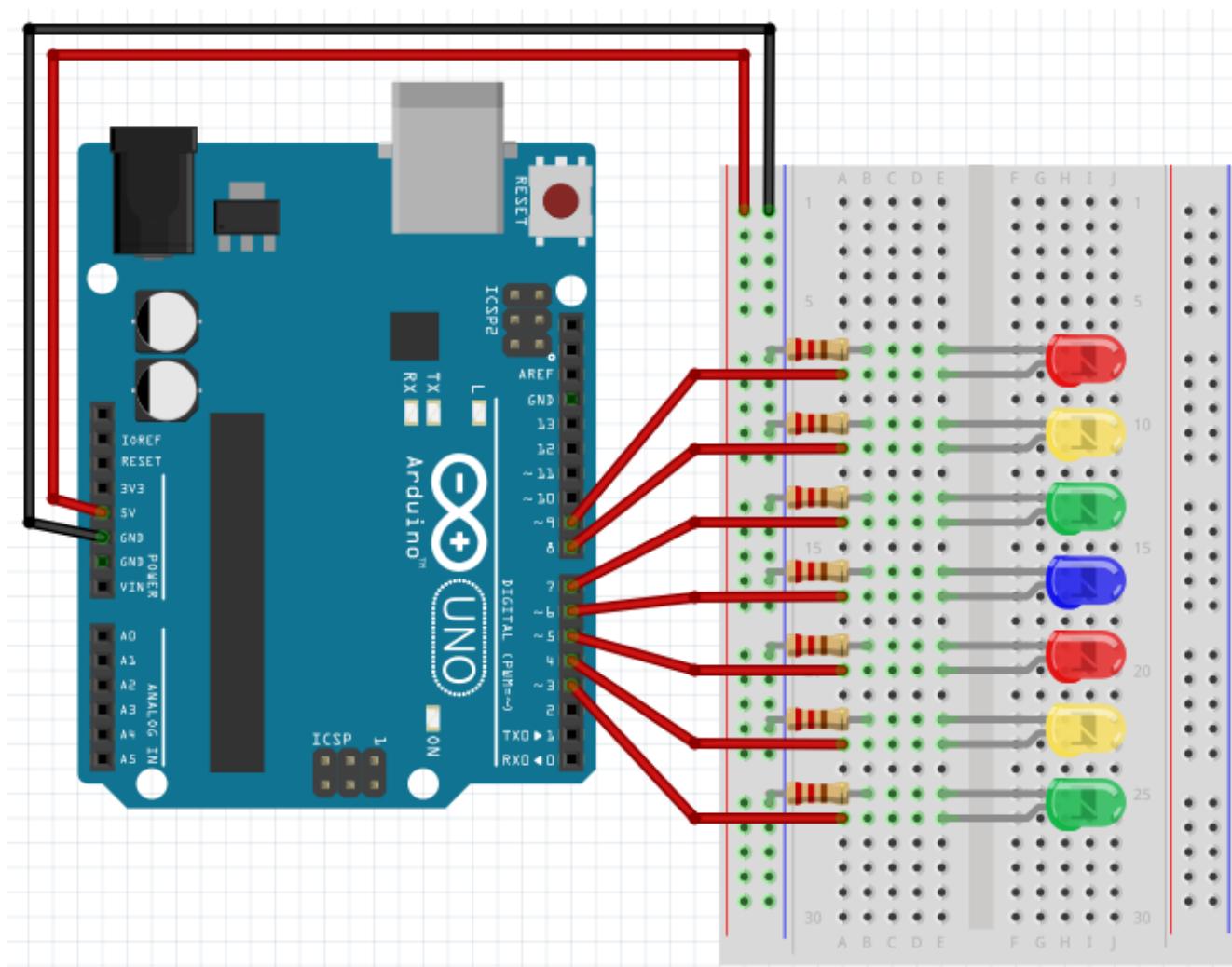


# 5. LED홀수 짹수 제어



스마트인재개발원  
Smart Human Resources Development

# 5. LED홀수 짹수 제어 - 배선도



# 5. LED홀수 짹수 제어 - 코드

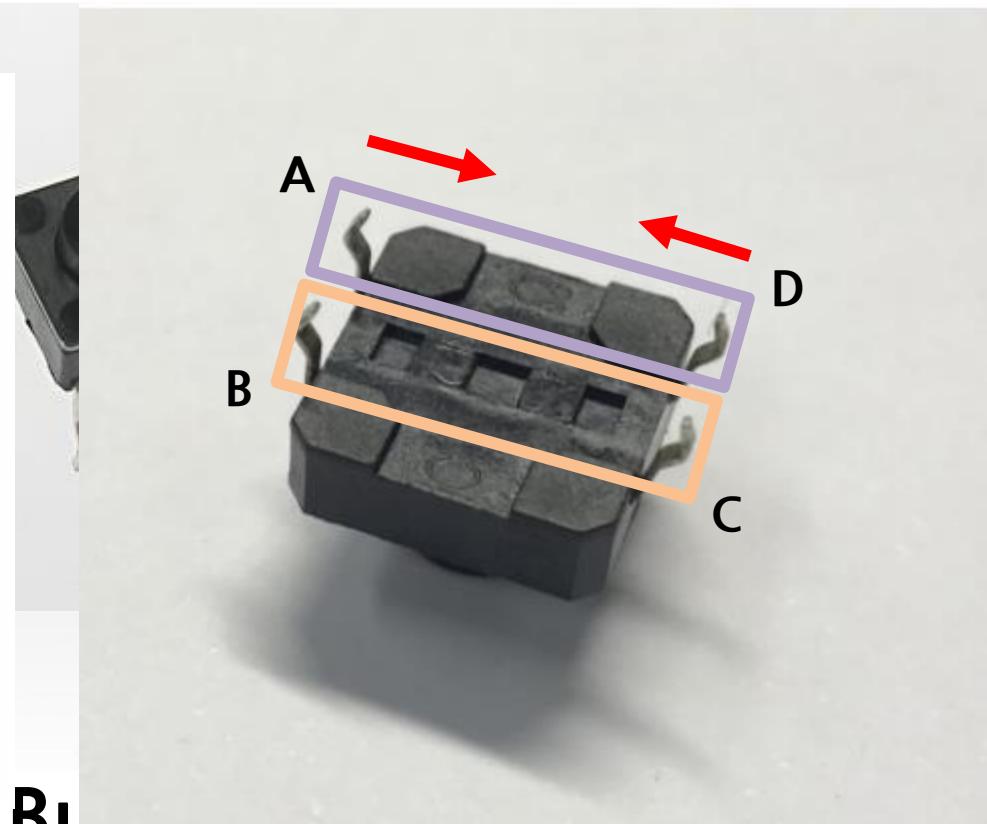
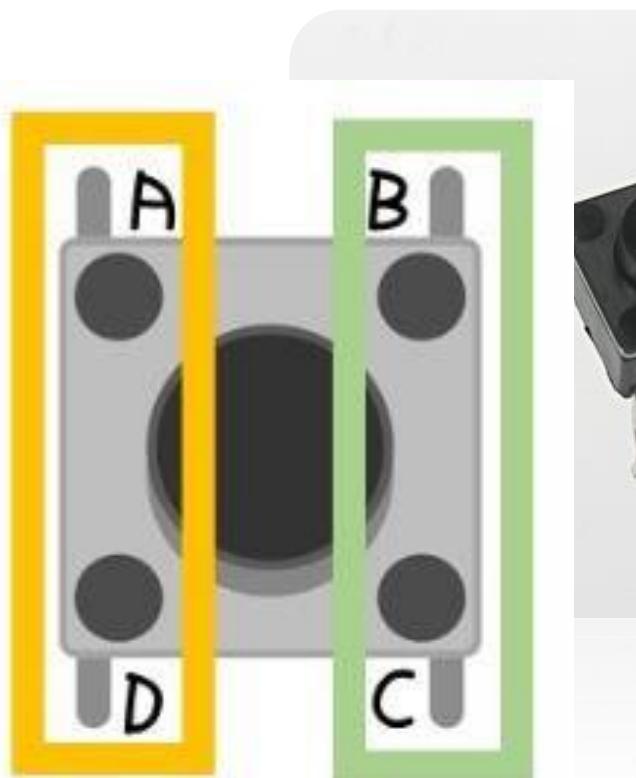


```
void setup() {  
    pinMode(3, OUTPUT);  
    pinMode(4, OUTPUT);  
    pinMode(5, OUTPUT);  
    pinMode(6, OUTPUT);  
    pinMode(7, OUTPUT);  
    pinMode(8, OUTPUT);  
    pinMode(9, OUTPUT);  
}  
  
void loop() {  
    digitalWrite(3, HIGH);  
    delay(300);  
    digitalWrite(3, LOW);  
    digitalWrite(5, HIGH);  
    delay(300);  
    digitalWrite(5, LOW);  
  
    digitalWrite(7, HIGH);  
    delay(300);  
    digitalWrite(7, LOW);  
    digitalWrite(9, HIGH);  
    delay(300);  
    digitalWrite(9, LOW);  
    digitalWrite(4, HIGH);  
    delay(300);  
    digitalWrite(4, LOW);  
    digitalWrite(6, HIGH);  
    delay(300);  
    digitalWrite(6, LOW);  
    digitalWrite(8, HIGH);  
    delay(300);  
    digitalWrite(8, LOW);  
}
```

# PushButton

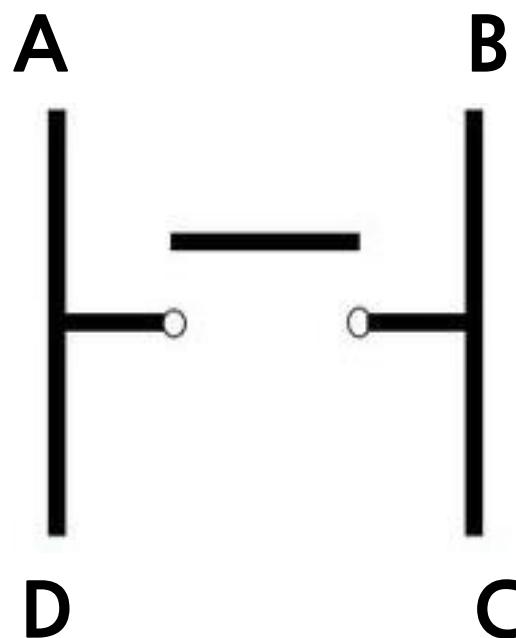


# PushButton

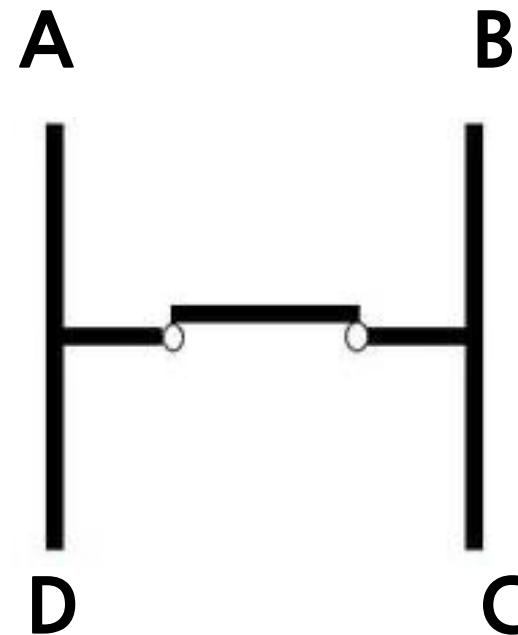


Button..

# PushButton 회로도

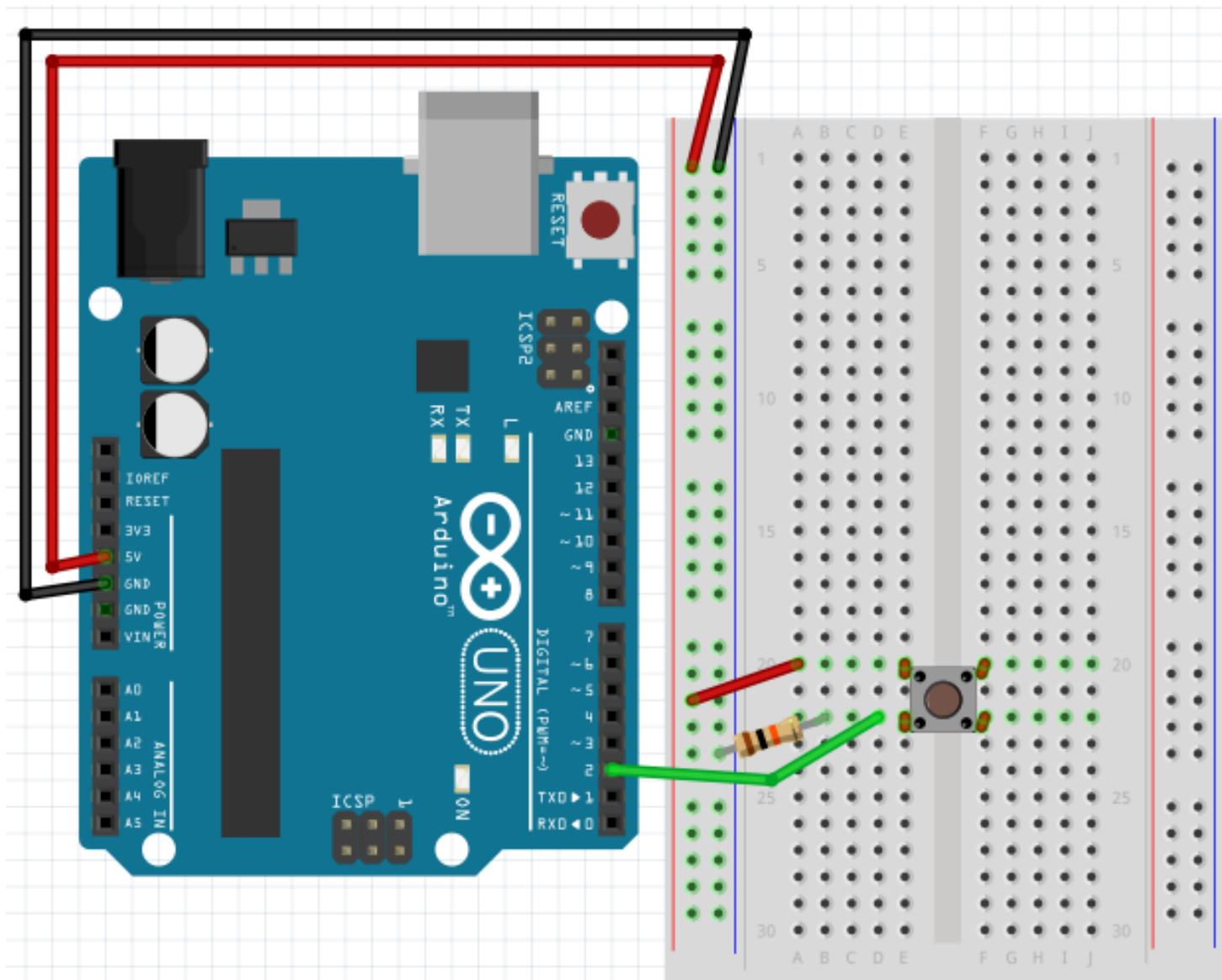


버튼을 누르지 않았을 때

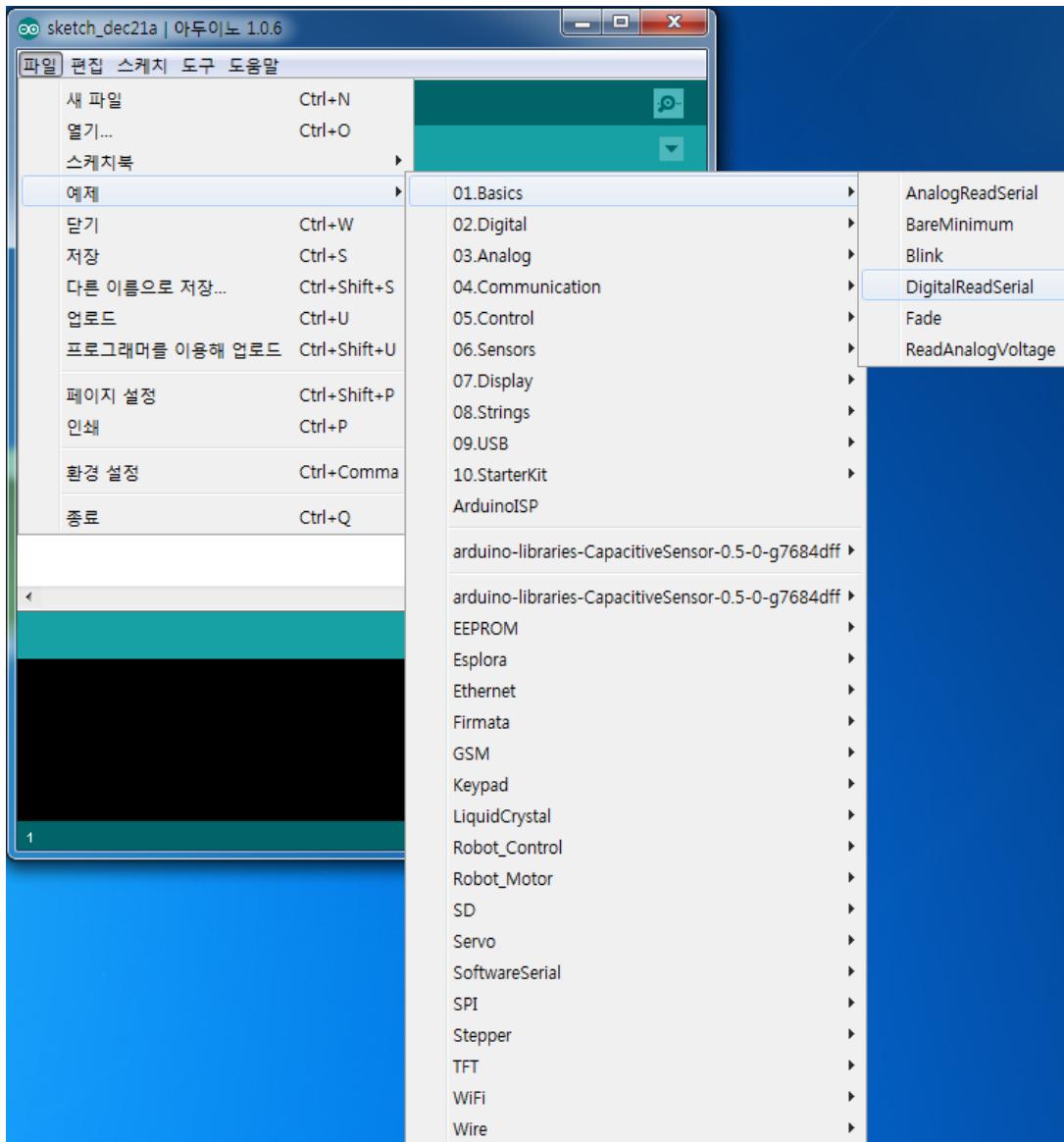


버튼을 눌렀을 때

# 6. PushButton



# 6. PushButton



## 1. 파일

## 2. 예제

## 3. Basics

## 4. Digital Read Serial

# 6. PushButton

sketch\_dec14a | 아두이노 1.6.5

```

int pushButton = 2;

void setup() {
    pinMode(pushButton, INPUT);
}

void loop() {
    digitalWrite(pushButton);
}

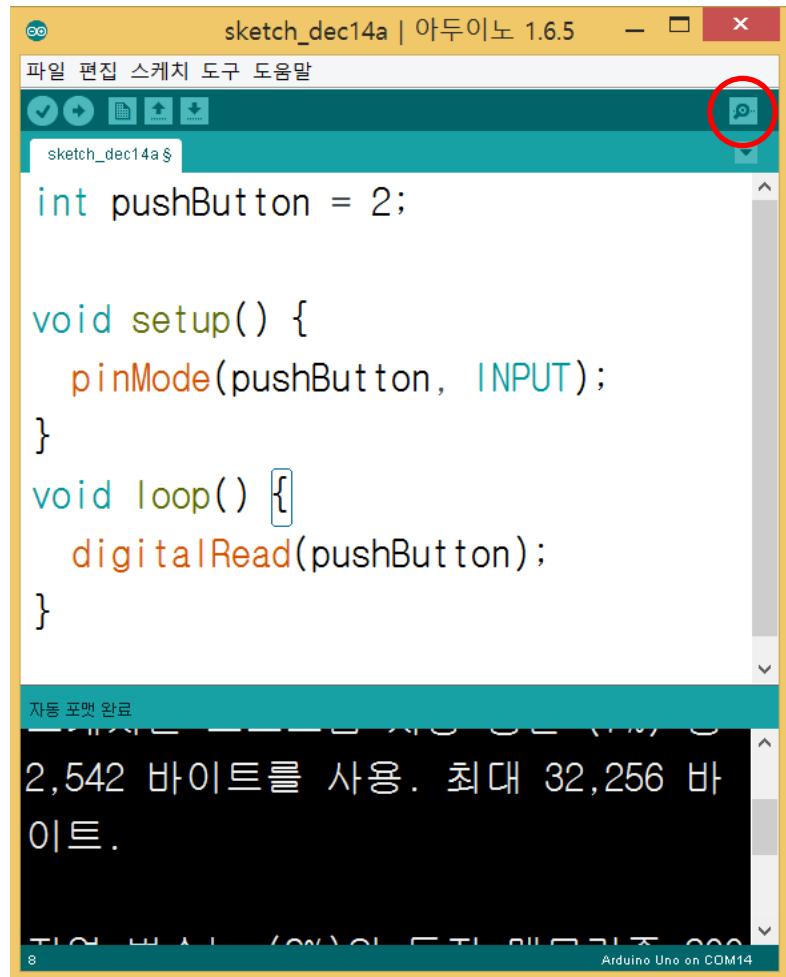
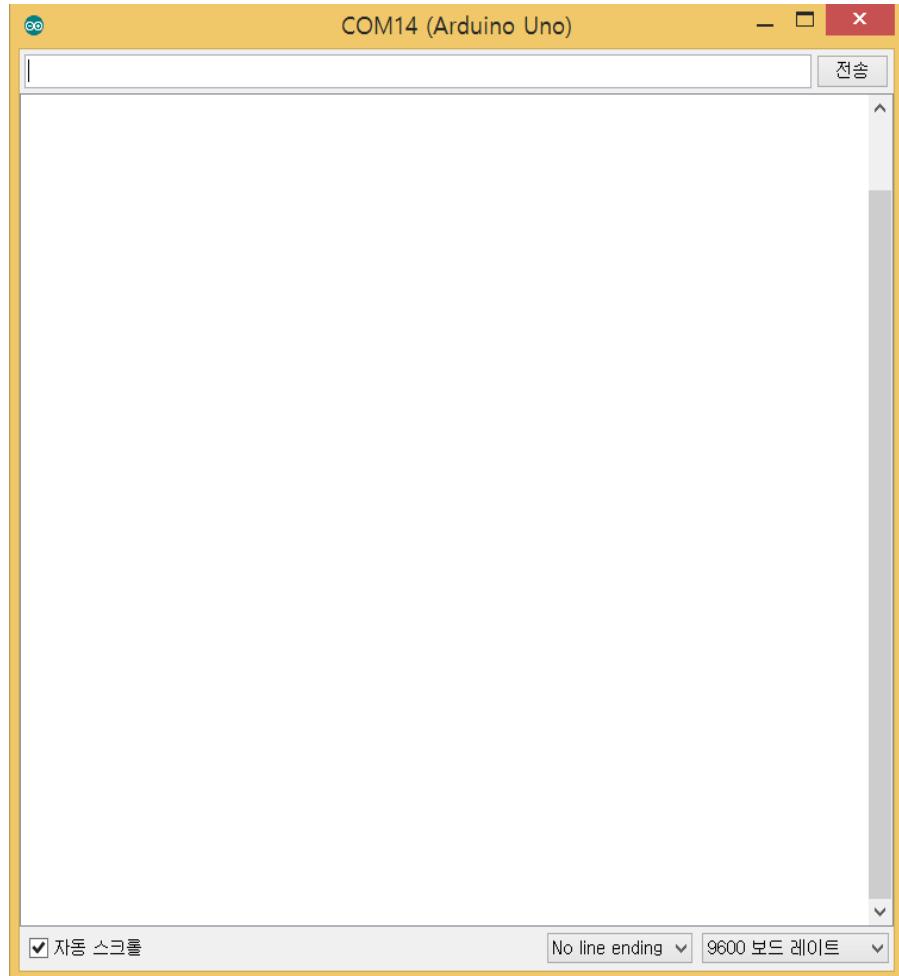
```

자동 포맷 완료

2,542 바이트를 사용. 최대 32,256 바이트.

8

Arduino Uno on COM14

## 6. PushButton



# 6. PushButton



- **pinMode(2, INPUT);**

연결된 pin번호 입·출력 설정

- **digitalRead(2);**

데이터를 읽어드릴 pin번호

- **Serial.begin(9600);**

통신속도

- **Serial.println("A");**

Serial모니터에 ()안에 들어있는 문구 출력

# 6. PushButton



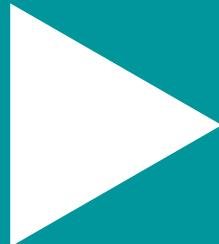
```
int pushButton = 2;
```

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

```
void loop() {  
    int buttonState = digitalRead(pushButton);  
    Serial.println(buttonState);  
    delay(1);  
}
```

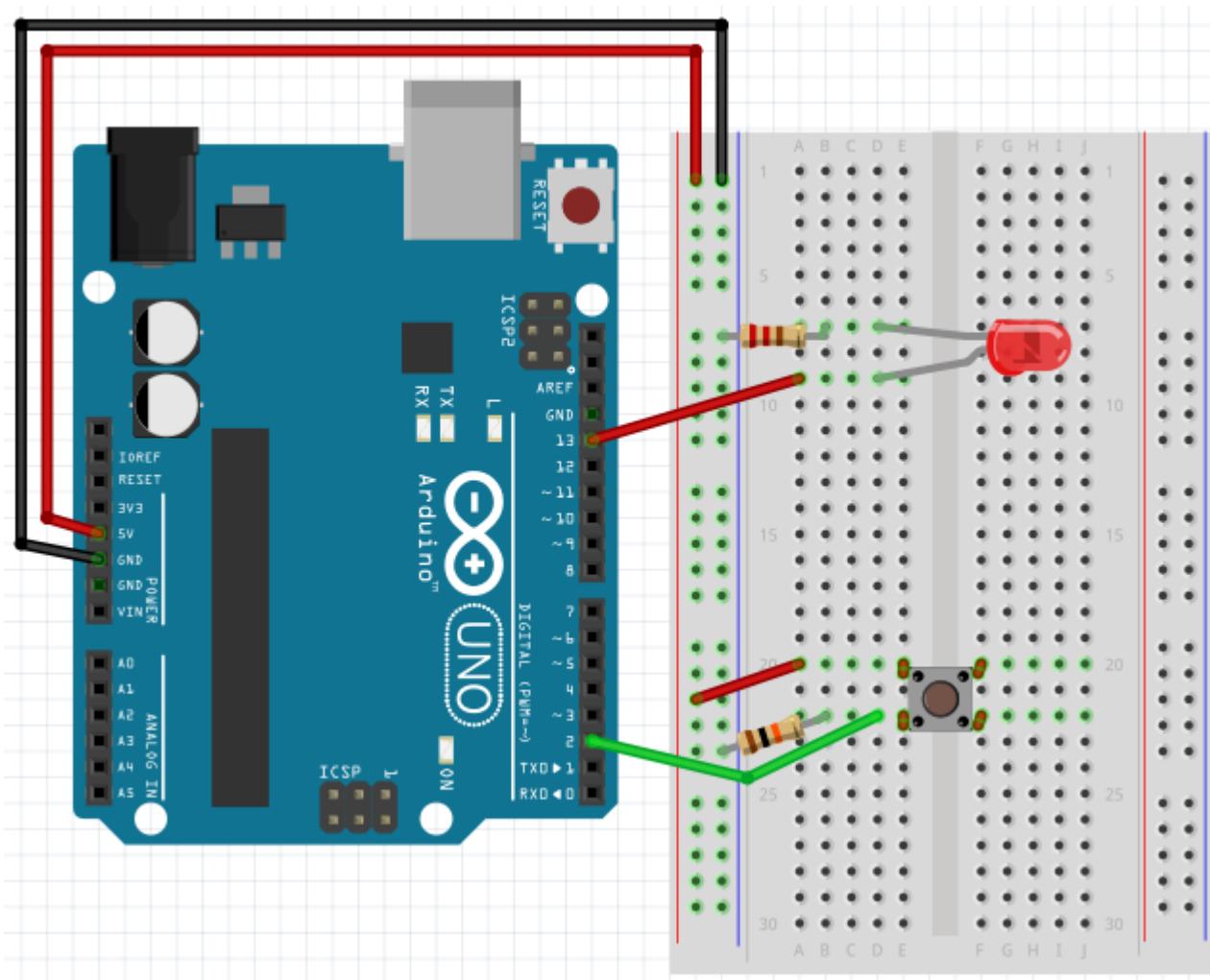


# 7. PushButton을 이용한 LED제어

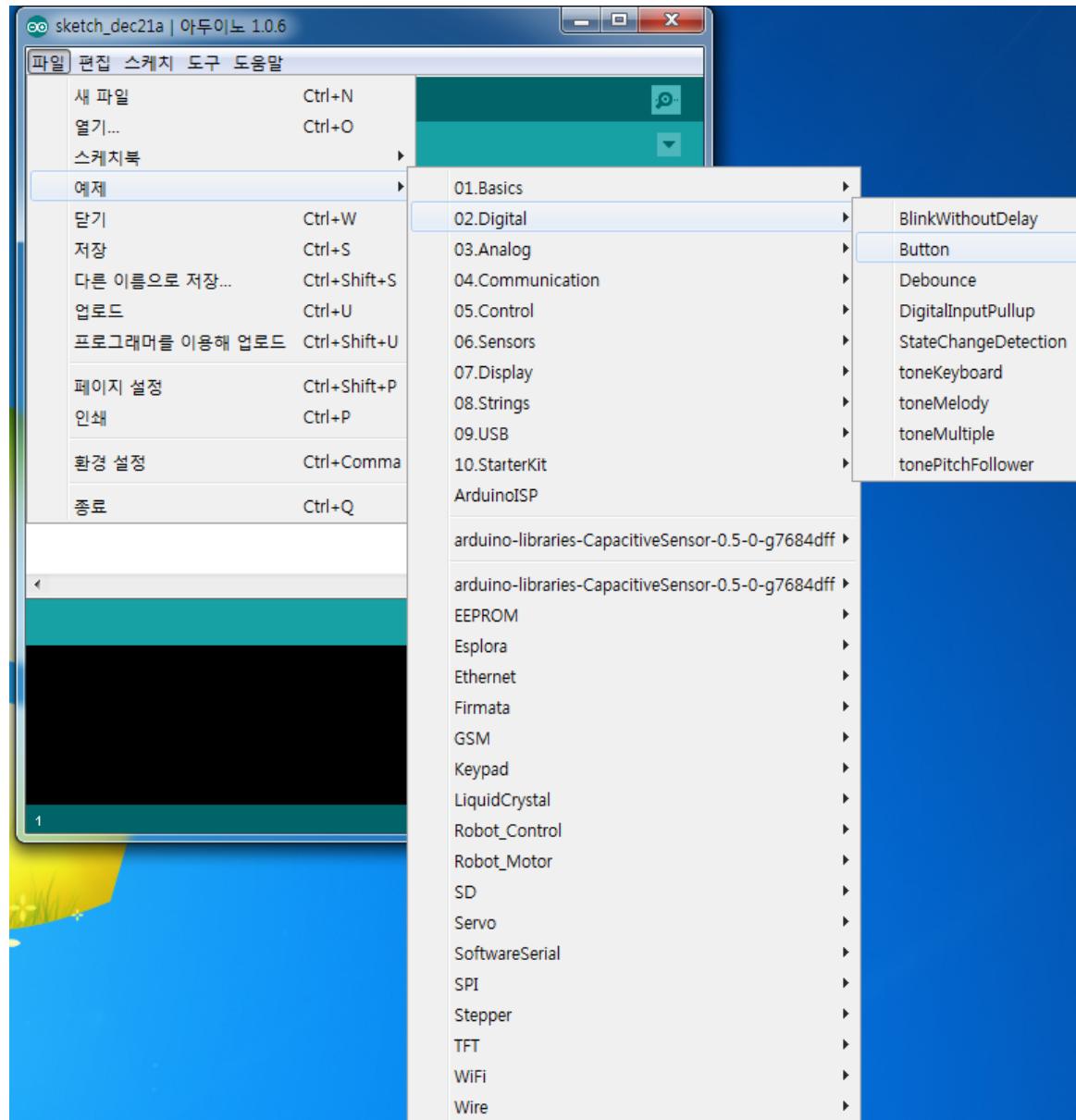


스마트인재개발원  
Smart Human Resources Development

# 7. PushButton을 이용한 LED제어



# 7. PushButton을 이용한 LED제어



## 1. 파일

## 2. 예제

## 3. Digital

## 4. Button

# 7. PushButton을 이용한 LED제어



The screenshot shows the Arduino IDE interface with a sketch named "Button". The code initializes pins 2 and 13 as INPUT and OUTPUT respectively, and sets up pin 13 as an output. The setup function also initializes the button state to 0. The code is as follows:

```
Button | 아두이노 1.0.6
파일 편집 스케치 도구 도움말
Button §
const int buttonPin = 2;
const int ledPin = 13;

int buttonState = 0;

void setup() {
  pinMode(ledPin, OUTPUT);
  pinMode(buttonPin, INPUT);
}
```

At the top of the window, there is a toolbar with various icons. One icon, which is a magnifying glass with three dots, is highlighted with a red circle. The status bar at the bottom shows "20" on the left and "Arduino Uno on COM5" on the right.



# 7. PushButton을 이용한 LED제어



```
const int buttonPin = 2;
```

```
const int ledPin = 13;
```

```
int buttonState = 0;
```

```
void setup() {
```

```
  pinMode(ledPin, OUTPUT);
```

```
  pinMode(buttonPin, INPUT);
```

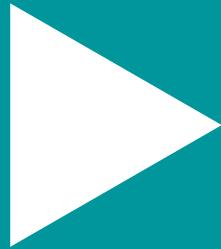
```
}
```

# 7. PushButton을 이용한 LED제어



```
void loop() {  
    buttonState = digitalRead(buttonPin);  
  
    if (buttonState == HIGH) {  
  
        digitalWrite(ledPin, HIGH);  
  
    }  
  
    else {  
  
        digitalWrite(ledPin, LOW);  
  
    }  
}
```

# 8. PushButton을 이용한 LED제어 2



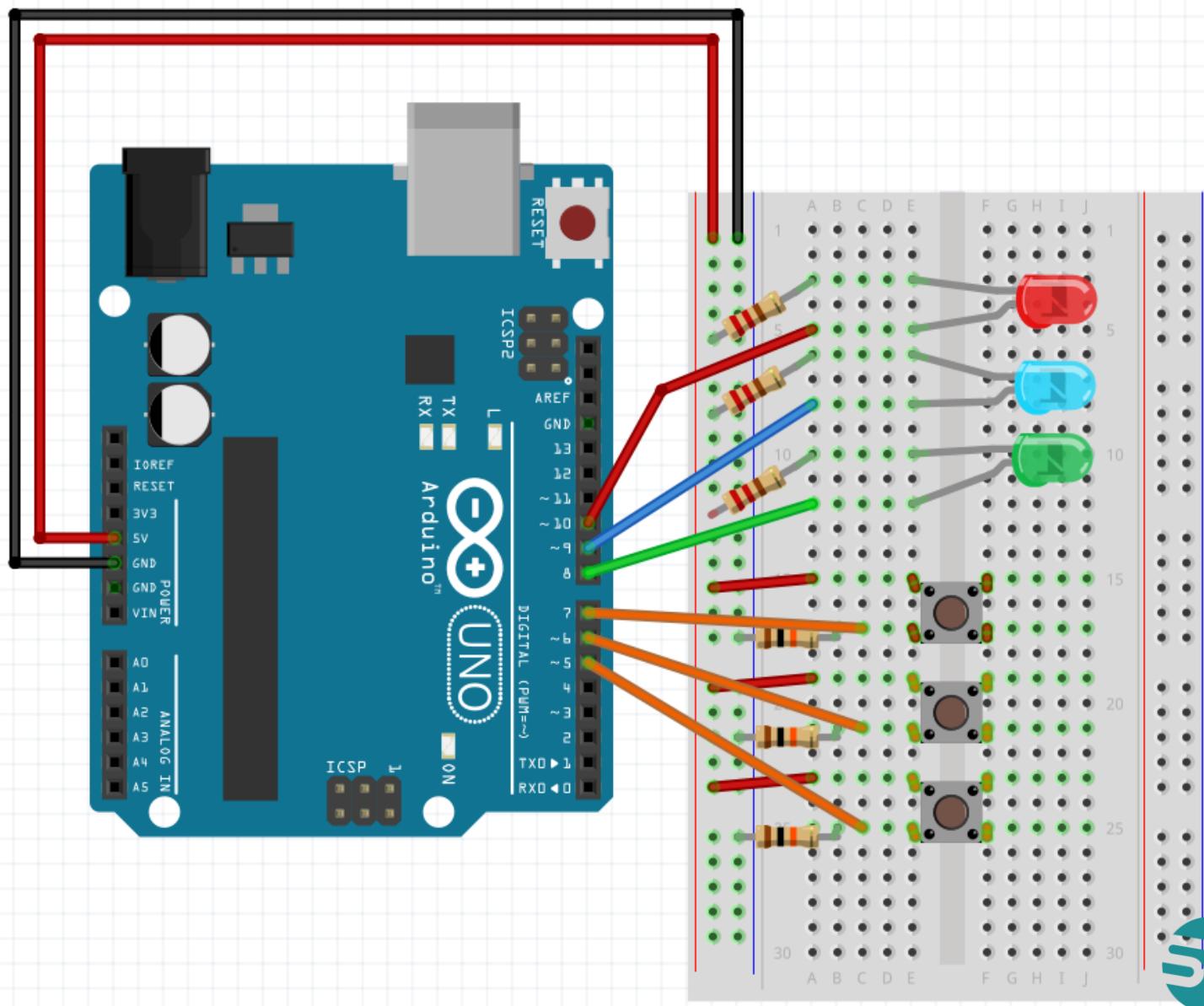
스마트인재개발원  
Smart Human Resources Development

# 8. PushButton을 이용한 LED제어 2



- Arduino Board
- R LED, G LED, Y LED
- 220-ohm resistor(\*3)
- push button(\*3)
- 10k-ohm resistor(\*3)

# 8. PushButton을 이용한 LED제어 2

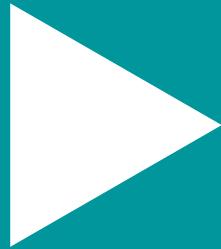


# 8. PushButton을 이용한 LED제어 2



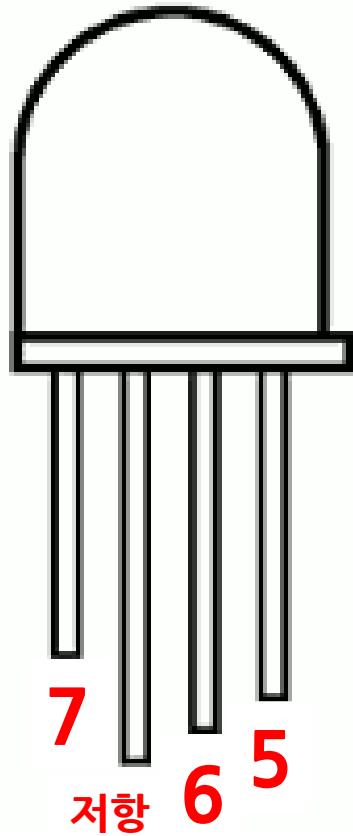
```
void setup() {  
    pinMode(10, OUTPUT);  
    pinMode(9, OUTPUT);  
    pinMode(8, OUTPUT);  
    pinMode(7, INPUT);  
    pinMode(6, INPUT);  
    pinMode(5, INPUT);  
}  
  
void loop() {  
    if(digitalRead(7) == HIGH){  
        digitalWrite(10, HIGH);  
    }else{  
        digitalWrite(10,LOW);  
    }  
    .  
    .  
    .  
}
```

# 9. Button을 이용한 LED제어 3



스마트인재개발원  
Smart Human Resources Development

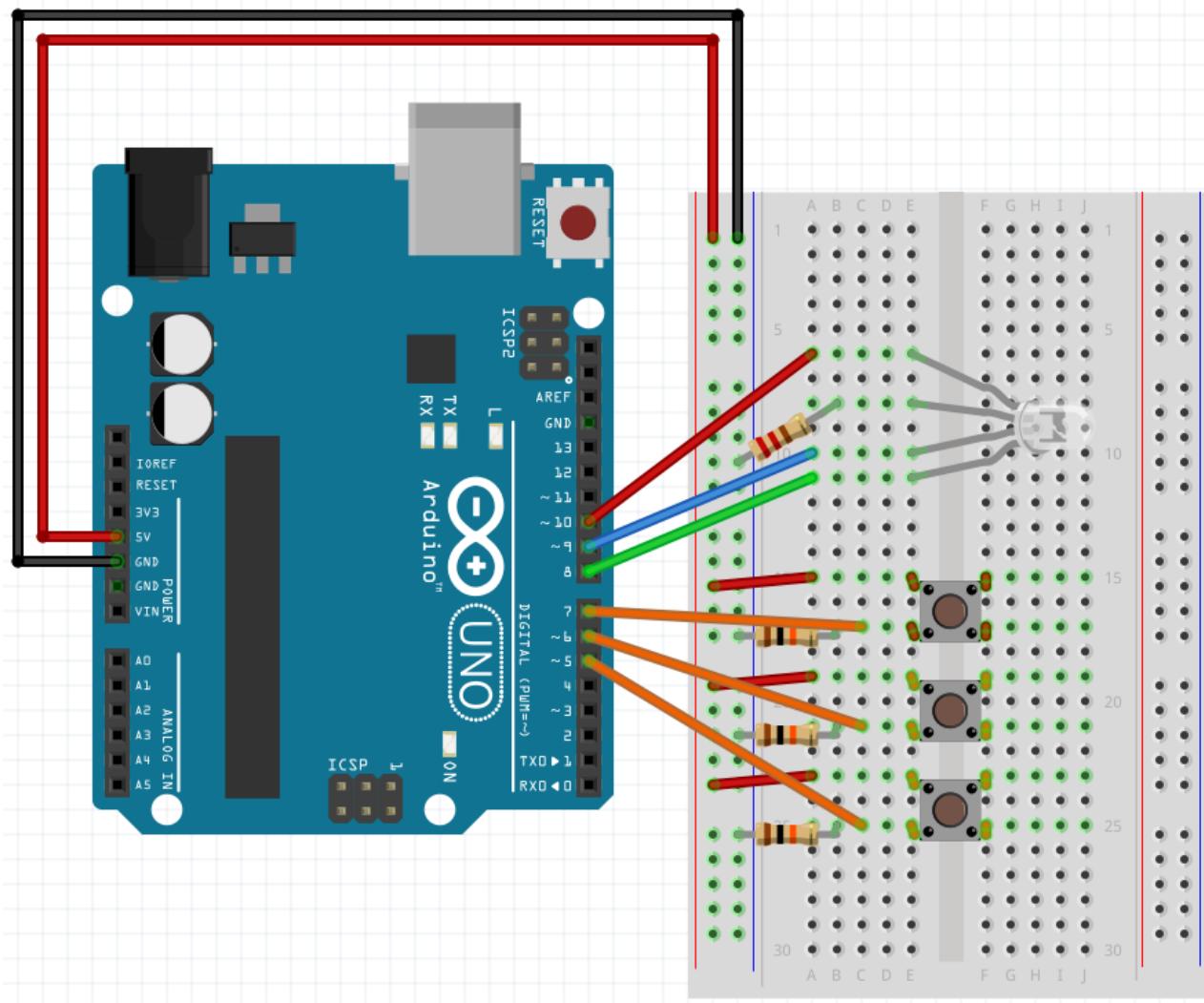
# 9. Button을 이용한 LED제어 3



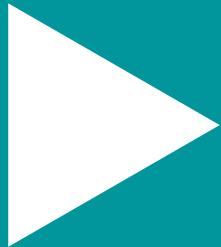
## RGB LED

- 1: Green (+)
- 2: Ground (-)
- 3: Blue (+)
- 4: Red (+)

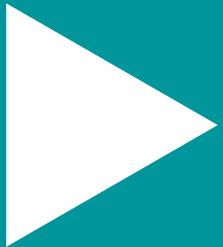
# 9. Button을 이용한 LED제어 3



# 10. 버튼을 이용한 스탠드



# 빛의 밝기 표현하기



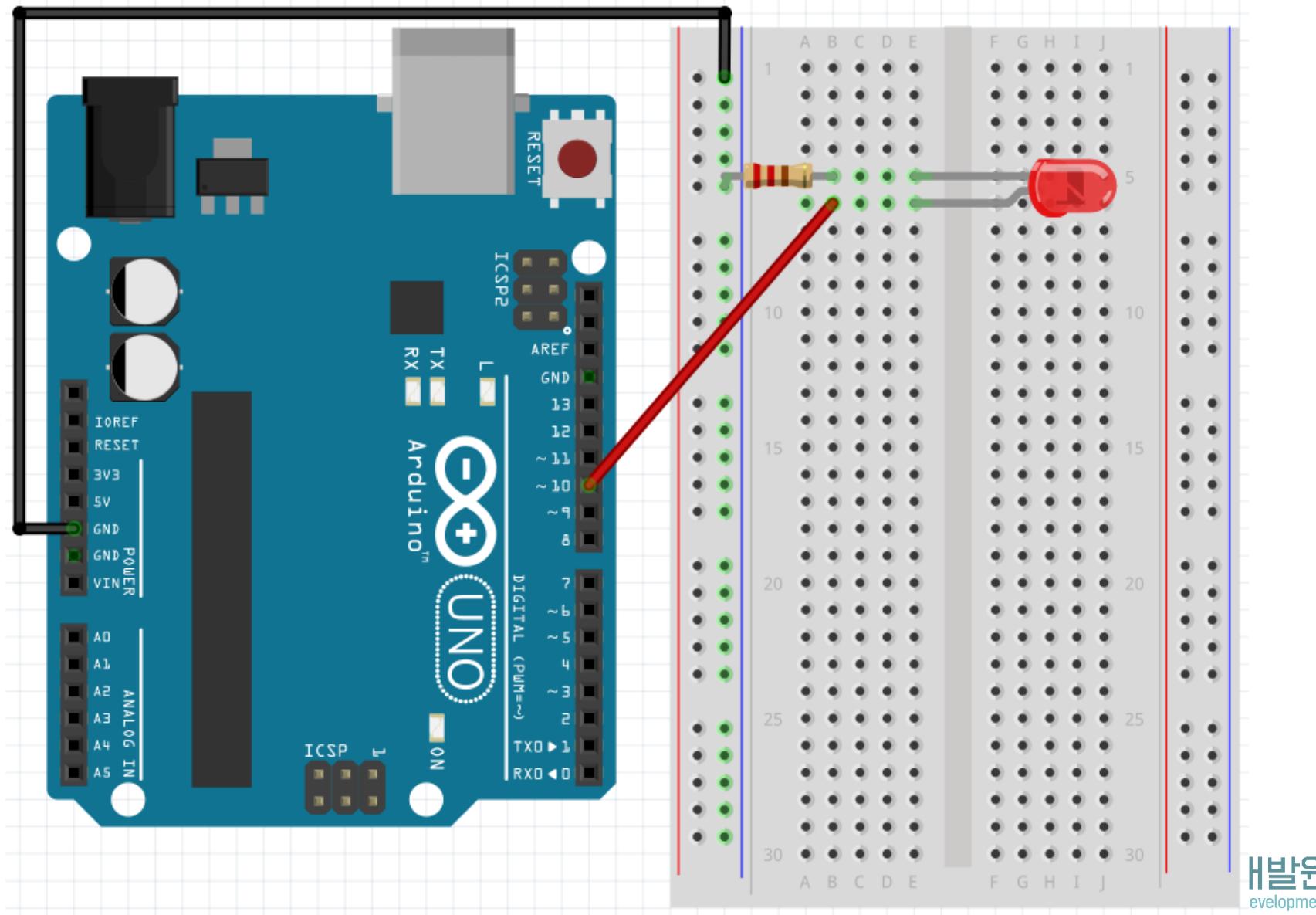
스마트인재개발원  
Smart Human Resources Development

# 빛의 밝기 표현하기



LED : 0 ~ 255

# 빛의 밝기 표현하기 - 배선도



- **analogWrite(10, 200);**

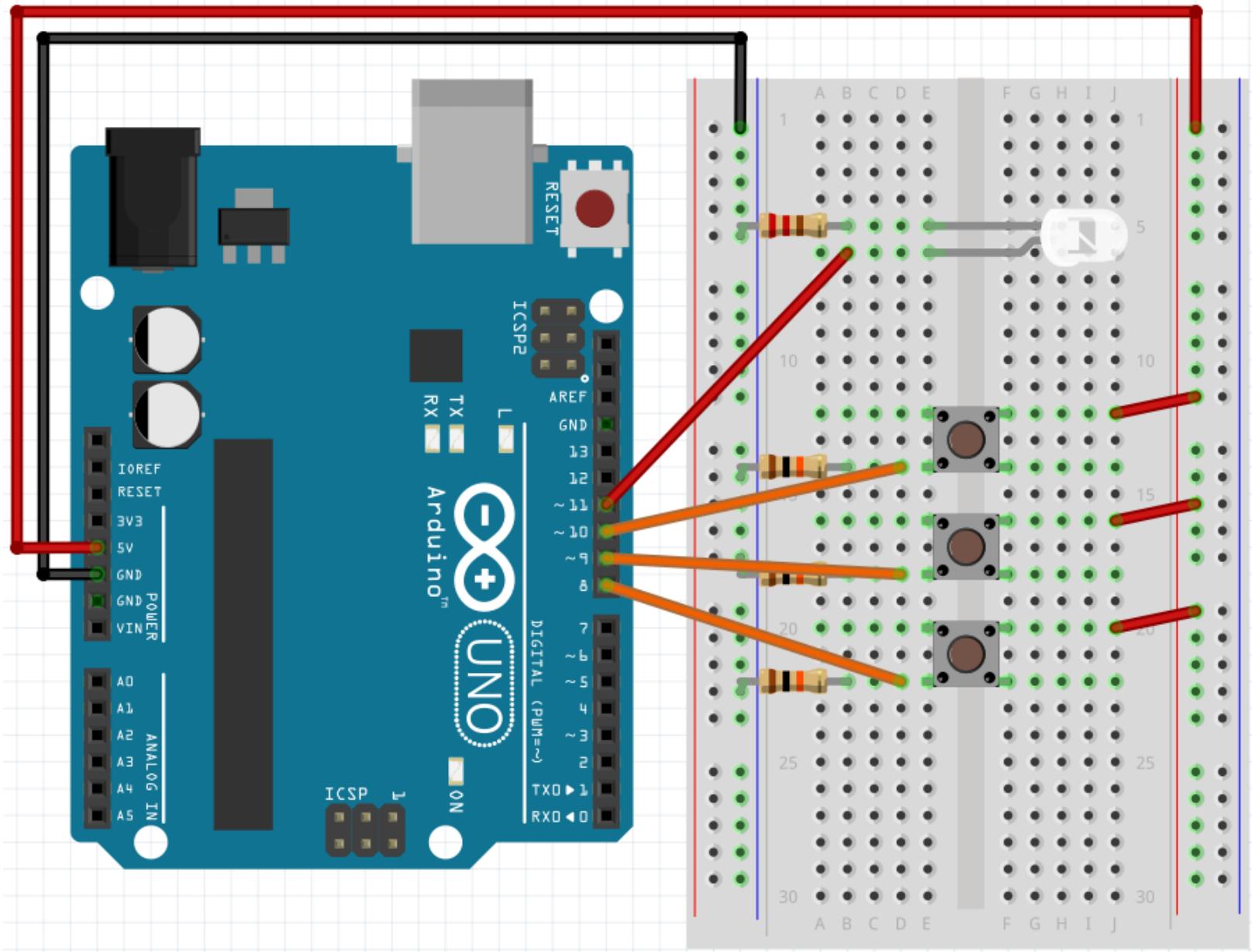
Pin번호 앞에 ~이 붙어 있는 pin 사용      출력 할 값

```
void setup() {  
    pinMode(10, OUTPUT);  
}  
 
```

```
void loop(){  
    analogWrite(10, 0);  
    delay(500);  
    analogWrite(10, 50);  
    delay(500);  
    analogWrite(10, 250);  
    delay(500);  
}  
 
```



# 10. 버튼을 이용한 스탠드- 배선도



# 10. 버튼을 이용한 스탠드 - 코드



```
void setup() {  
    pinMode(11, OUTPUT);  
    pinMode(10, INPUT);  
    pinMode(9, INPUT);  
    pinMode(8, INPUT);  
}
```

# 10. 버튼을 이용한 스탠드 - 코드



```
void loop() {  
    if(digitalRead(10)==1){  
        analogWrite(11, 30);  
    }  
    else if(digitalRead(9)==1){  
        analogWrite(11, 255);  
    }  
    else if(digitalRead(8)==1){  
        analogWrite(11, 0);  
    }  
}
```



스마트인재개발원  
Smart Human Resources Development



감사합니다!



스마트인재개발원  
나예호 선임연구원