



# Arduino

# ▶ Arduino 입력과 출력

## ✓ 아두이노 입력, 출력

아두이노는 입력, 출력을 하기 위한 장비들이 존재하는데 이를 센서와 액추에이터로 구분한다.

- 센서 : 입력장치, 외부환경의 특정 이벤트에 의해 전기신호를 보내주는 것

- \* 세밀한 조율이 필요하여 거의 아날로그 신호로 처리함. 디지털신호 센서도 있음

- 액추에이터 : 출력장치, 입력받은 값을 이용하여 데이터를 출력하거나, 움직이는 행동을 하는것

- \* 디지털신호를 처리함

# ▶ Arduino 입력과 출력

## ✓ 입력장치(센서)



인체감지센서



DHT11센서  
온습도센서



Digital버튼

## ✓ 출력장치(액추에이터)



LED



피에조부저

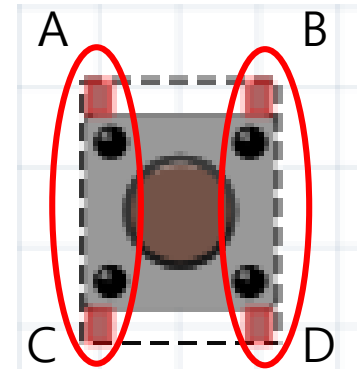


OLED모듈

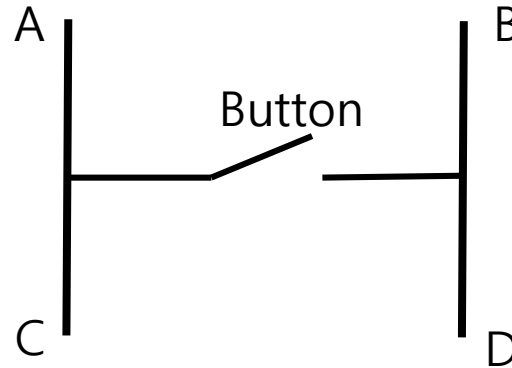
\* 지급된 장비를 기준으로 작성. 더 많은 장비들이 있음

# ▶ Arduino 입력받기

## ✓ 버튼을 이용하여 데이터 받기



서로 연결되어있는 전선



위 그림에서 봤을때 A,C와 B,D는 button에 눌림이나 땜의 행동에 따라서 전류가 흐르거나 흐르지 않게 제어할 수 있음

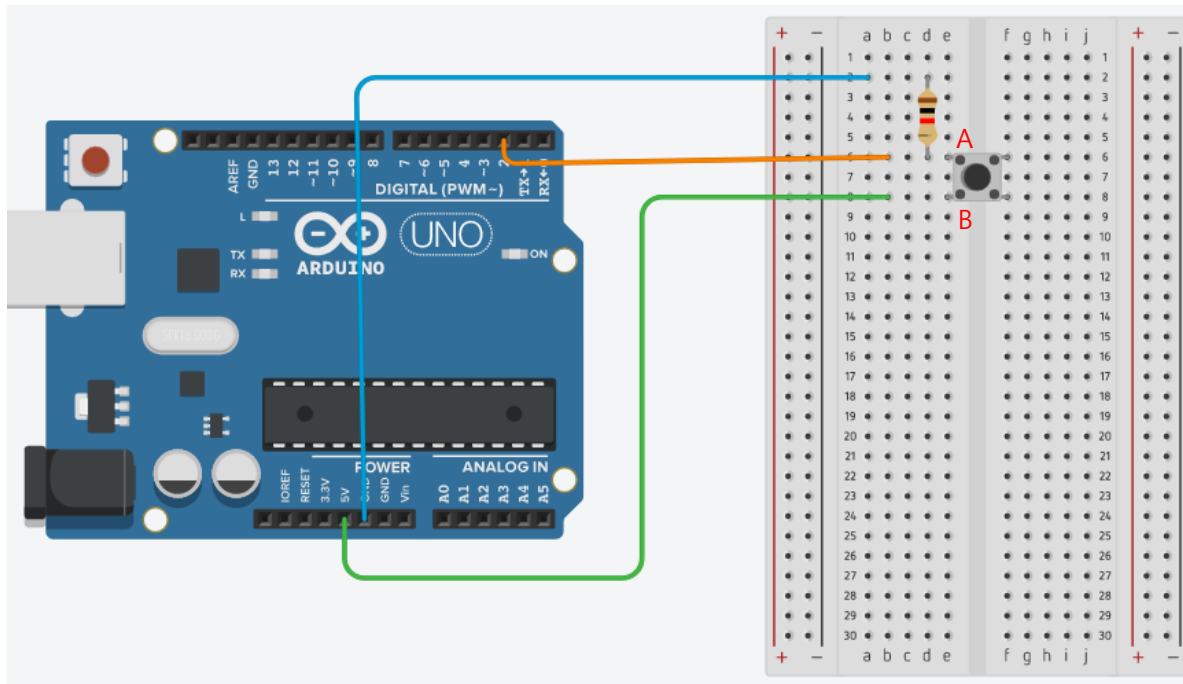
- Pull-down : 버튼이 눌렸을때 전류가 흐르게 하는 방식
- Pull-up : 버튼이 떨어져 있을때 전류가 흐르게 하는 방식

# ▶ Pull-Down방식 Button

## ✓ 준비장비

아두이노 Uno보드, digitalbutton 1개, 저항기 1개, USB연결 케이블, 연결케이블 3개, IDE도구

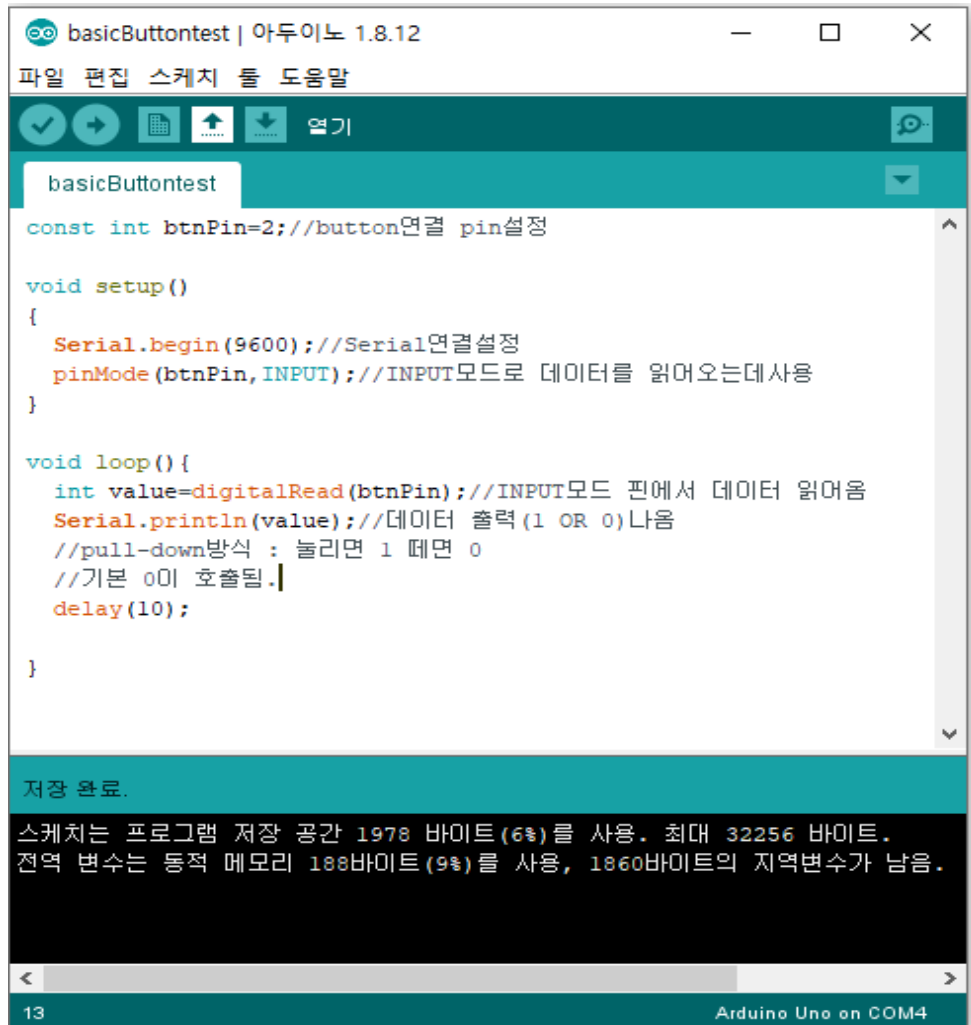
## ✓ 회로설계



- 버튼을 e열에 한쪽부분 A,B 또는 C,D를 연결
- A와 연결된 가로선에 저항기 한 부분을 연결, 반대쪽 저항기 부분에는 연결 케이블로 GND Pin과 연결함
- 버튼 A와 저항기가 같이 연결되어있는 곳을 digital2 pin을 연결함
- 버튼 B 부분을 5v핀에 연결케이블로 연결함
- USB케이블을 아두이노와 컴퓨터 USB포트에 연결

# ▶ Pull-Down방식 Button

## ✓ IDE코딩하기

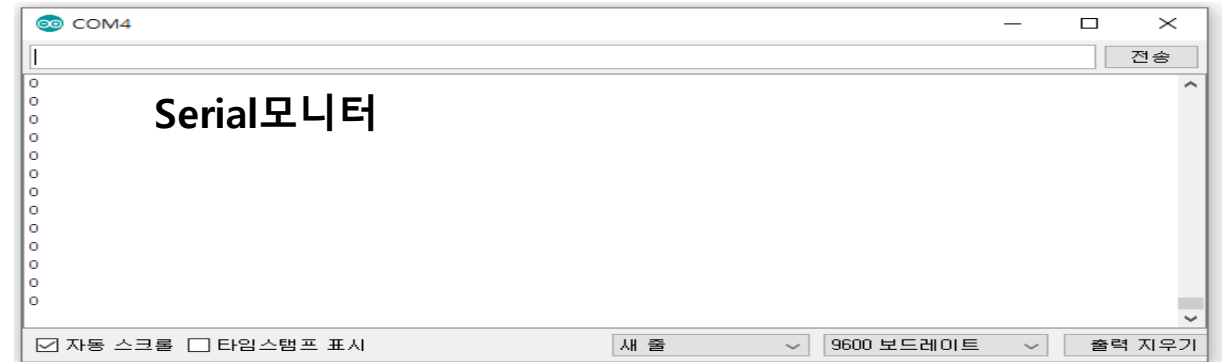


```
basicButtontest | 아두이노 1.8.12
파일 편집 스케치 툴 도움말
[Icons] 열기
basicButtontest
const int btnPin=2;//button연결 pin설정

void setup()
{
  Serial.begin(9600);//Serial연결설정
  pinMode(btnPin,INPUT);//INPUT모드로 데이터를 읽어오는데사용
}

void loop(){
  int value=digitalRead(btnPin);//INPUT모드 핀에서 데이터 읽어옴
  Serial.println(value);//데이터 출력(1 OR 0)나옴
  //pull-down방식 : 눌리면 1 떴면 0
  //기본 0이 호출됨.|
  delay(10);
}

저장 완료.
스케치는 프로그램 저장 공간 1978 바이트(6%)를 사용. 최대 32256 바이트.
전역 변수는 동적 메모리 188바이트(9%)를 사용, 1860바이트의 지역변수가 남음.
```



버튼동작확인하기 \* 버튼을 누르면 1 출력/기본 0 출력

- 코드를 작성하고
- V 버튼을 클릭하여 해당 코드를 컴파일함
- 컴파일을 하면 아래 출력창에 컴파일 결과가 출력됨
- → 버튼을 클릭하여 아두이노 Uno Board에 업로드를 실행함.
- 툴 -> 시리얼모니터 클릭(자바의 콘솔창)

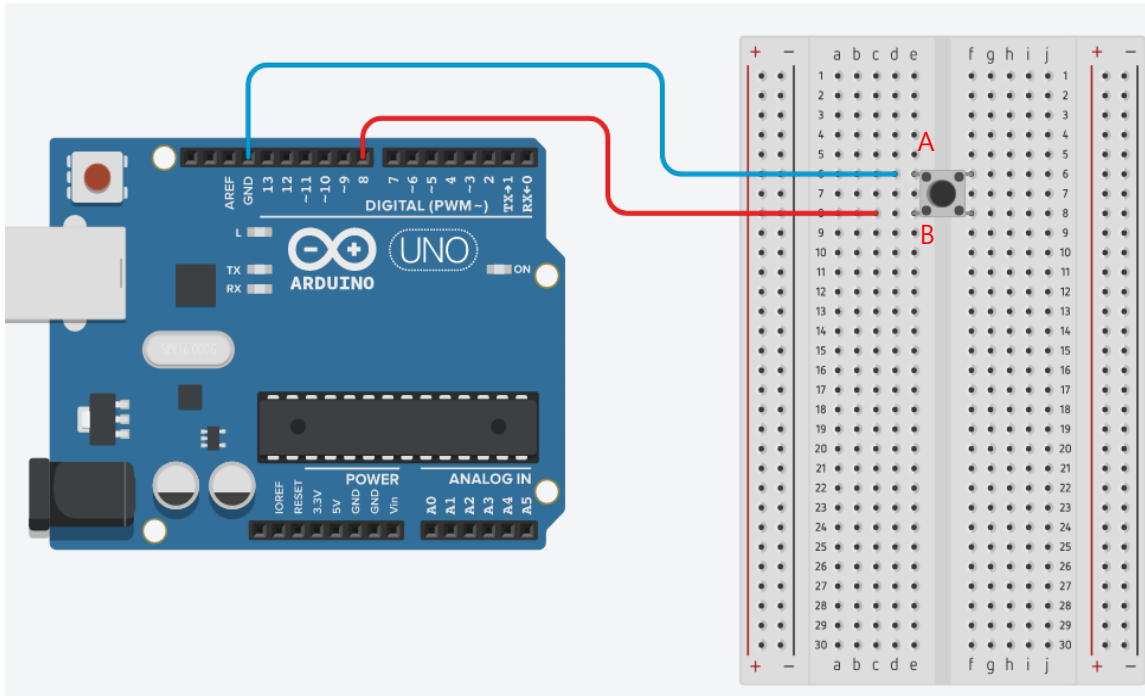
# ▶ Pull-up방식 Button

## ✓ 준비장비

아두이노 Uno보드, digitalbutton 1개, USB연결 케이블, 연결케이블 3개, IDE도구

\* 버튼 내부의 저항을 이용하는 방식으로 저항기가 필요가 없음

## ✓ 회로설계



- 버튼을 e열에 한쪽부분 A,B 또는 C,D를 연결
- A와 연결된 가로선에 연결 케이블로 GND Pin과 연결함
- 버튼 B 부분을 digital 8 핀에 연결케이블로 연결함
- USB케이블을 아두이노와 컴퓨터 USB포트에 연결

## ▶ Pull-up방식 Button

## ✓ IDE코딩하기

The screenshot shows the Arduino IDE with the following code in the main editor window:

```

const int btn=8;

void setup() {
  Serial.begin(9600);
  pinMode(btn, INPUT_PULLUP);
  //pull-up방식은 입력값을 INPUT_PULLUP으로 설정해야함
}

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

```

Below the code editor, a status bar displays the memory usage:

업로드 완료.  
 스케치는 프로그램 저장 공간 1832 바이트 (5%)를 사용. 최대 32256 바이트.  
 전역 변수는 동적 메모리 188바이트 (9%)를 사용, 1860바이트의 지역변수가 남음.

The bottom status bar shows the page number 14 and the board selection: Arduino Uno on COM4.

The screenshot displays a serial communication interface window titled "COM4". The main area contains a vertical stream of characters, all of which are the digit "1". At the top right, there is a "전송" (Send) button. The bottom status bar includes several controls: a checked checkbox for "자동 스크롤" (Auto Scroll), an unchecked checkbox for "타임스탬프 표시" (Show Timestamps), a dropdown menu currently showing "9600 보드레이트" (9600 Baud Rate), and a button labeled "출력 지우기" (Clear Output).

- 버튼 동작 확인하기 \* 버튼을 누르면 0 출력/기본 1 출력
- 코드를 작성하고
- V 버튼을 클릭하여 해당 코드를 컴파일함
- 컴파일을 하면 아래 출력창에 컴파일 결과가 출력됨
- → 버튼을 클릭하여 아두이노 Uno Board에 업로드를 실행함.
- 툴 -> 시리얼모니터 클릭(자바의 콘솔창)

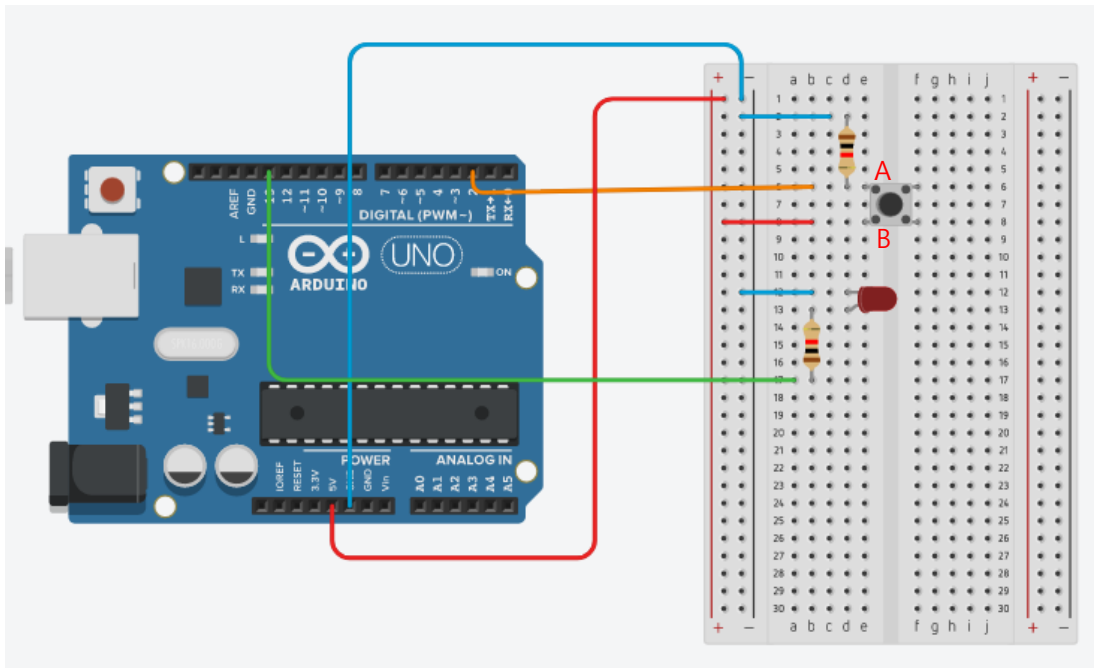


# ▶ Pull-down방식 Button 응용하기

## ✓ 준비장비

아두이노 Uno보드, digitalbutton 1개, 저항기 2개, LED 1개, USB연결 케이블, 연결케이블 3개, IDE도구

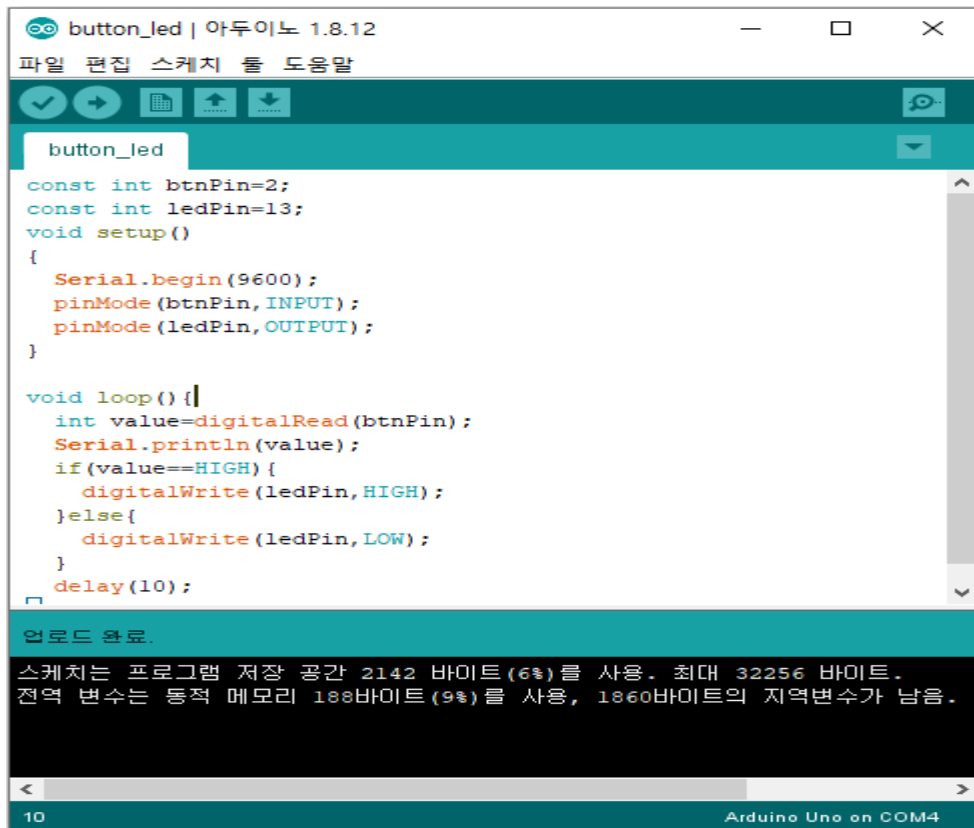
## ✓ 회로설계



- 버튼을 e열에 한쪽부분 A,B 또는 C,D를 연결
- A와 연결된 가로선에 저항기 한쪽을 연결하고 반대쪽은 - 세로열에 연결케이블을 이용하여 연결
- A와 저항기 한쪽이 연결되어있는 가로선에 연결선을 이용하여 digitalpin 2에 연결함.
- B의 가로선에 연결케이블을 이용하여 +세로열에 연결함
- 전구와 저항기를 연결하고 전구 -극(짧은선)을 -세로열에 연결케이블을 이용하여 연결
- 저항기 연결되어있는 가로선을 연결케이블을 이용하여 digitalpin 13에 연결함
- pin 5v를 연결케이블을 이용하여 + 세로열에 연결
- pin GND를 연결케이블을 이용하여 - 세로열에 연결

# ▶ Pull-down방식 Button 응용하기

## ✓ IDE코딩하기



```
button_led | 아두이노 1.8.12
파일 편집 스케치 툴 도움말

button_led
const int btnPin=2;
const int ledPin=13;
void setup()
{
  Serial.begin(9600);
  pinMode(btnPin, INPUT);
  pinMode(ledPin, OUTPUT);
}

void loop() {
  int value=digitalRead(btnPin);
  Serial.println(value);
  if(value==HIGH){
    digitalWrite(ledPin, HIGH);
  }else{
    digitalWrite(ledPin, LOW);
  }
  delay(10);
}

업로드 완료.
스케치는 프로그램 저장 공간 2142 바이트 (6%)를 사용. 최대 32256 바이트.
전역 변수는 동적 메모리 188바이트 (9%)를 사용, 1860바이트의 지역변수가 남음.

10 Arduino Uno on COM4
```

버튼동작확인하기 \* 버튼을 누르면 1 출력/기본 0 출력

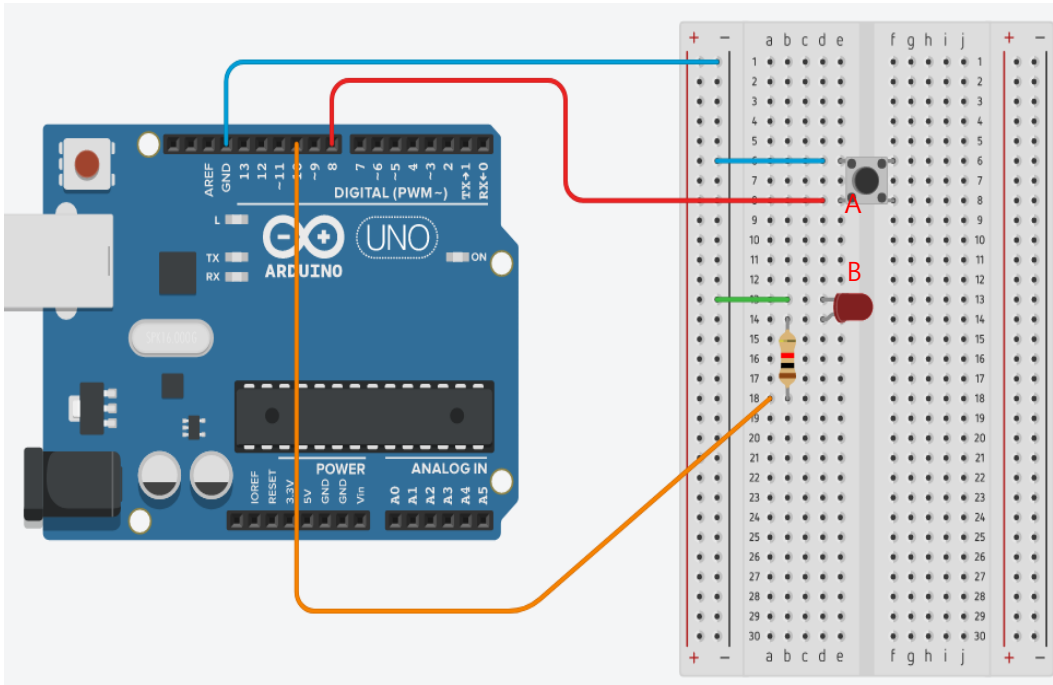
- 코드를 작성하고
- V 버튼을 클릭하여 해당 코드를 컴파일함
- 컴파일을 하면 아래 출력창에 컴파일 결과가 출력됨
- → 버튼을 클릭하여 아두이노 Uno Board에 업로드를 실행함.
- 툴 -> 시리얼모니터 클릭(자바의 콘솔창)

# ▶ Pull-up방식 Button 응용하기

## ✓ 준비장비

아두이노 Uno보드, digitalbutton 1개, 저항기 1개, LED 1개, USB연결 케이블, 연결케이블 3개, IDE도구

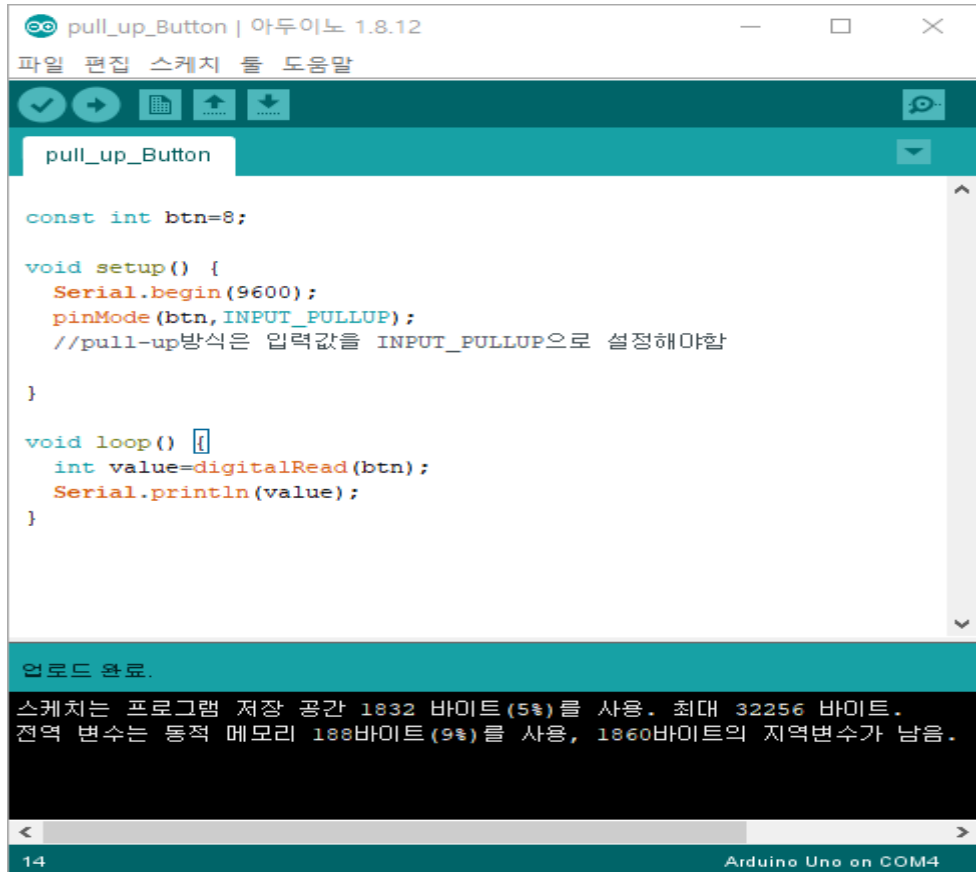
## ✓ 회로설계



- 버튼을 e열에 한쪽부분 A,B 또는 C,D를 연결
- A의 가로선을 – 세로열에 연결케이블을 이용하여 연결
- B의 가로선에 연결케이블을 이용하여 digitalpin 8에 연결함
- 전구와 저항기를 연결하고 전구 – 극(짧은선)을 – 세로열에 연결케이블을 이용하여 연결
- 저항기 연결되어있는 가로선을 연결케이블을 이용하여 digitalpin 10에 연결함
- pin GND를 연결케이블을 이용하여 – 세로열에 연결

# ▶ Pull-up방식 Button 응용하기

## ✓ IDE코딩하기



```
pull_up_Button | 아두이노 1.8.12
파일 편집 스케치 툴 도움말

pull_up_Button

const int btn=8;

void setup() {
  Serial.begin(9600);
  pinMode(btn, INPUT_PULLUP);
  //pull-up방식은 입력값을 INPUT_PULLUP으로 설정해야함
}

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

업로드 완료.

스케치는 프로그램 저장 공간 1832 바이트 (5%)를 사용. 최대 32256 바이트.  
전역 변수는 동적 메모리 188바이트 (9%)를 사용, 1860바이트의 지역변수가 남음.

14 Arduino Uno on COM4

버튼동작확인하기 \* 버튼을 누르면 0 출력/기본 1 출력

- 코드를 작성하고
- V 버튼을 클릭하여 해당 코드를 컴파일함
- 컴파일을 하면 아래 출력창에 컴파일 결과가 출력됨
- → 버튼을 클릭하여 아두이노 Uno Board에 업로드를 실행함.
- 툴 -> 시리얼모니터 클릭(자바의 콘솔창)

# ▶ 아두이노 Button이용하기 실습

## ✓ 과 제 1

버튼과 LED 각각 2개를 이용하여 하나는 버튼을 누르면 켜지고 하나는 버튼을 누르면 꺼지게 하기

## ✓ 과 제 2

toggle방식으로 버튼을 이용하여 LED on/off 하기