

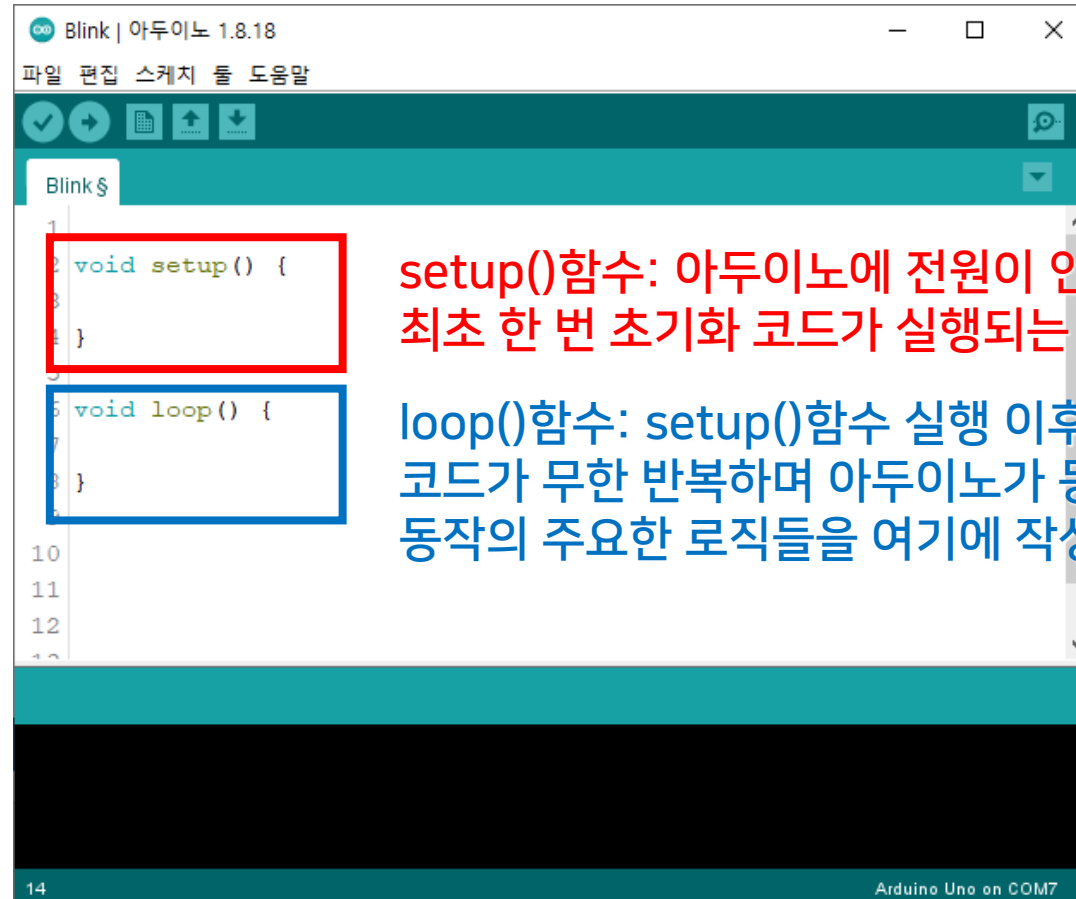
MakeUp  
아두이노

#MakeUp아두이노 #02

아두이노 기초

# 아두이노 기초

## 아두이노 스케치의 기본 구조

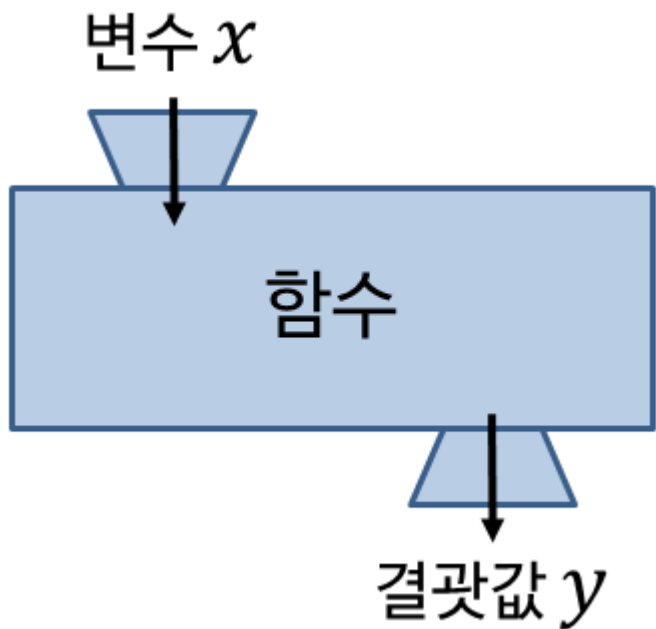


setup()함수: 아두이노에 전원이 인가되면  
최초 한 번 초기화 코드가 실행되는 부분

loop()함수: setup()함수 실행 이후 loop()함수에 작성한  
코드가 무한 반복하며 아두이노가 동작한다. (무한루프)  
동작의 주요한 로직들을 여기에 작성한다.

# 아두이노 기초

## 코드 살펴보기 - 함수



수학에서의 함수

```
int increase(int x) {  
    return x + 1;  
}
```

```
int result = plus(1); // result = 2;
```

```
int plus(int a, int b) {  
    return a + b;  
}
```

```
int result = plus(1, 2); // result = 3;
```

```
void led_on() {  
    digitalWrite(13, HIGH);  
}
```

```
led_on(); // LED를 켜는 동작이 실행됨
```

코딩에서의 함수 → 특정한 연산이나 동작을 정의

# 아두이노 기초

## 코드 살펴보기 - 함수

반환(리턴) 자료형

함수 이름

매개변수(파라미터)

```
int increase(int x) {  
    return x + 1;  
}  
  
int result = plus(1); // result = 2;
```

매개변수와 리턴이 없는 함수

```
void led_on() {  
    digitalWrite(13, HIGH);  
}  
  
led_on(); // LED를 켜는 동작이 실행됨
```

# 아두이노 기초

## 코드 살펴보기 - 변수, 상수

변수: 프로그램 상에서 임의의 데이터(값)들을 일시적으로 저장하는 공간.

데이터의 수정 및 변경이 가능하다.

상수: 프로그램 상에서 변경되지 않는 데이터 (고정된 값)

```
const int pin_mode = 0; // const키워드 : 변수처럼 이름을 붙여 상수를 선언할 수 있다.
```

```
int pin = 13; // pin이라는 int형 변수에 13 값을 넣는다. (저장한다.)
```

변수 자료형      변수 이름      대입 연산자      상수



# 아두이노 기초

## 코드 살펴보기 - 변수, 상수

아두이노 스케치에서 사용할 수 있는 기본 데이터 타입 (C++)

자료형	크기(Byte)		비고	값의 범위	
	8bit 보드	32bit 보드		8bit 보드	32bit 보드
char	1		문자형	-128~127	
byte	1		부호 없음	0~255	
int	2	4		-32,768~32,767	(-2^31) ~ (2^31 -1)
word	2	4	부호 없음		
short	2			-32,768~32,767	
long	4			-2,147,483,648~2,147,483,647	
unsigned char	1		부호 없음	0~255	
unsigned int	2	4		0~65,535	0~(2^32 - 1)
unsigned short	2			0~65,535	
unsigned long	4			0~4,294,967,295	

# 아두이노 기초

## 코드 살펴보기 - 주석

주석은 코드에 삽입되는 코멘트로, 프로그램 로직과 실행 과정에 영향을 미치지 않는다.

- 코드에 대한 간략한 설명이나 메시지를 남기는데 사용
- 한 줄 주석(//), 여러 줄 주석(/\* \*/)이 있다.

```
1  /*
2   Blink
3   LED가 1초 간격으로 깜빡이는 코드
4   아두이노 기본 예제에 포함되어 있다.
5  */
6
7  // the setup function runs once when you press reset or power the board
8  void setup() {
9      pinMode(LED_BUILTIN, OUTPUT); // 내장 LED핀을 출력 모드로 설정한다.
10 }
11
12 // the loop function runs over and over again forever
13 void loop() {
14     digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
15     delay(1000); // wait for a second
16     digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
17     delay(1000); // wait for a second
18 }
```

# 아두이노 기초

---

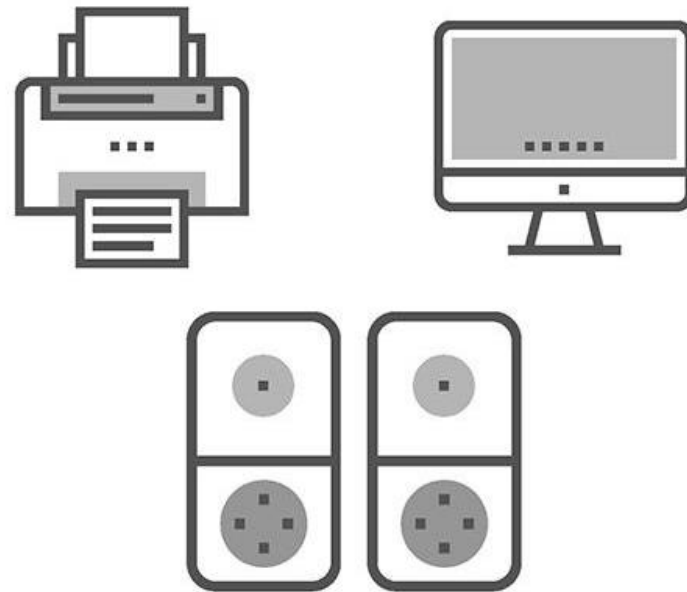
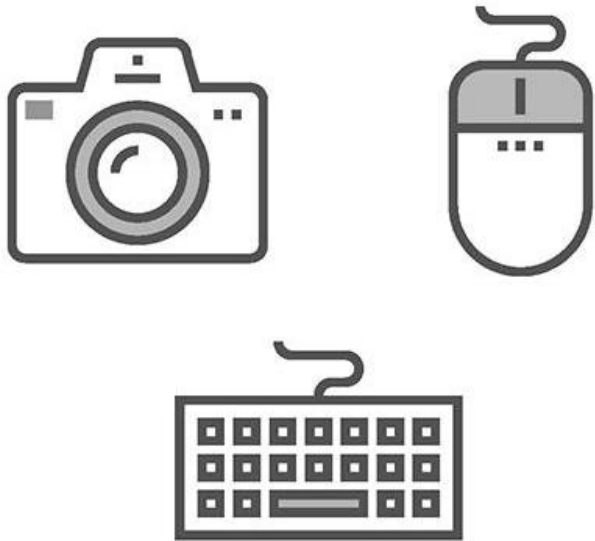
## 과제풀이

- LED를 3번 깜빡이는 함수를 작성하라 (flash3())



# 아두이노 기초

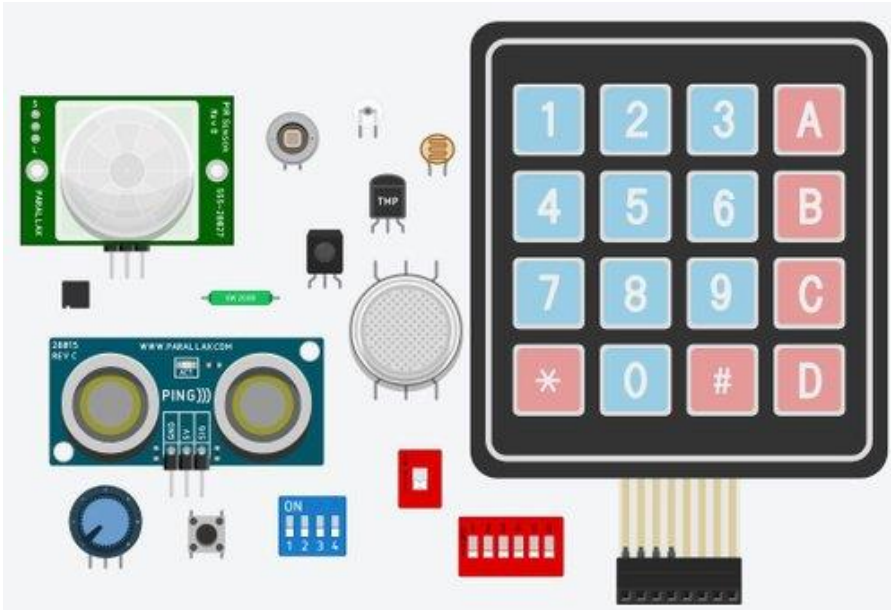
## 입력과 출력이란



입력장치 / 출력장치

# 아두이노 기초

## 아두이노의 입력과 출력장치



입력장치  
(각종 센서, 스위치 등)



출력장치  
(디스플레이, LED, 모터, 릴레이 등)

# 아두이노 기초

---

## digitalWrite(), digitalRead()

digitalWrite() :

- 디지털 핀의 **출력**을 제어한다.
- 0V/5V = ON/OFF = HIGH / LOW

digitalRead() :

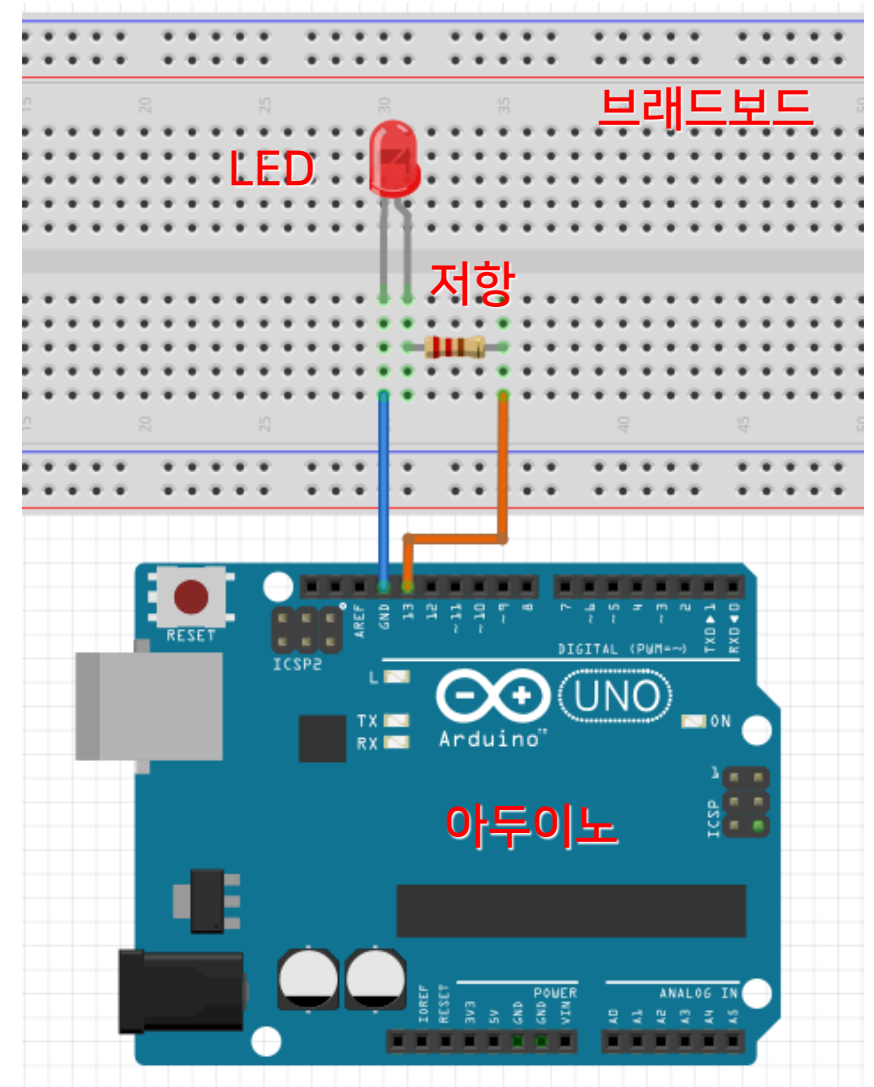
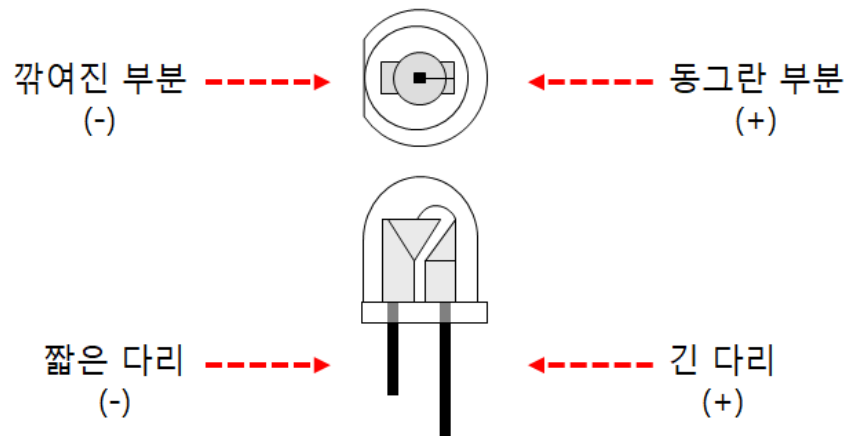
- 디지털 핀의 **입력** 상태를 읽어온다.
- 스위치의 ON/OFF상태, 센서의 상태값

# 아두이노 기초

## 출력 회로 구성하기

아두이노 우노의 디지털 13번 핀에 LED를 연결

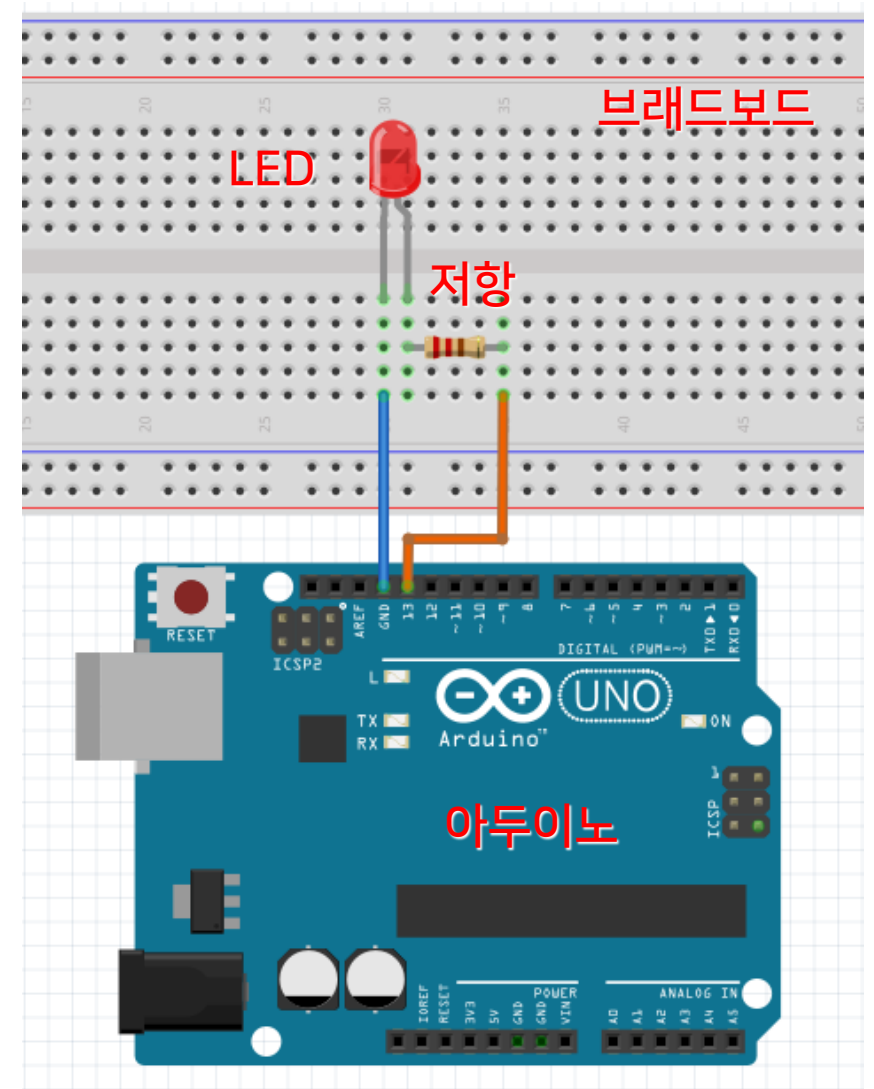
- LED는 바로 연결하지 않고 저항을 통해 연결한다.
- LED는 +- 극성이 있다.
- (D13핀)-(저항)-(LED+) / (LED-)-(GND핀) 과 같이 연결



# 아두이노 기초

## 디지털 출력 (LED 켜기/끄기)

- LED가 연결된 핀을 출력 모드로 설정
  - `pinMode(13, OUTPUT);`
- LED가 연결된 핀에 HIGH출력
  - `digitalWrite(13, HIGH);`

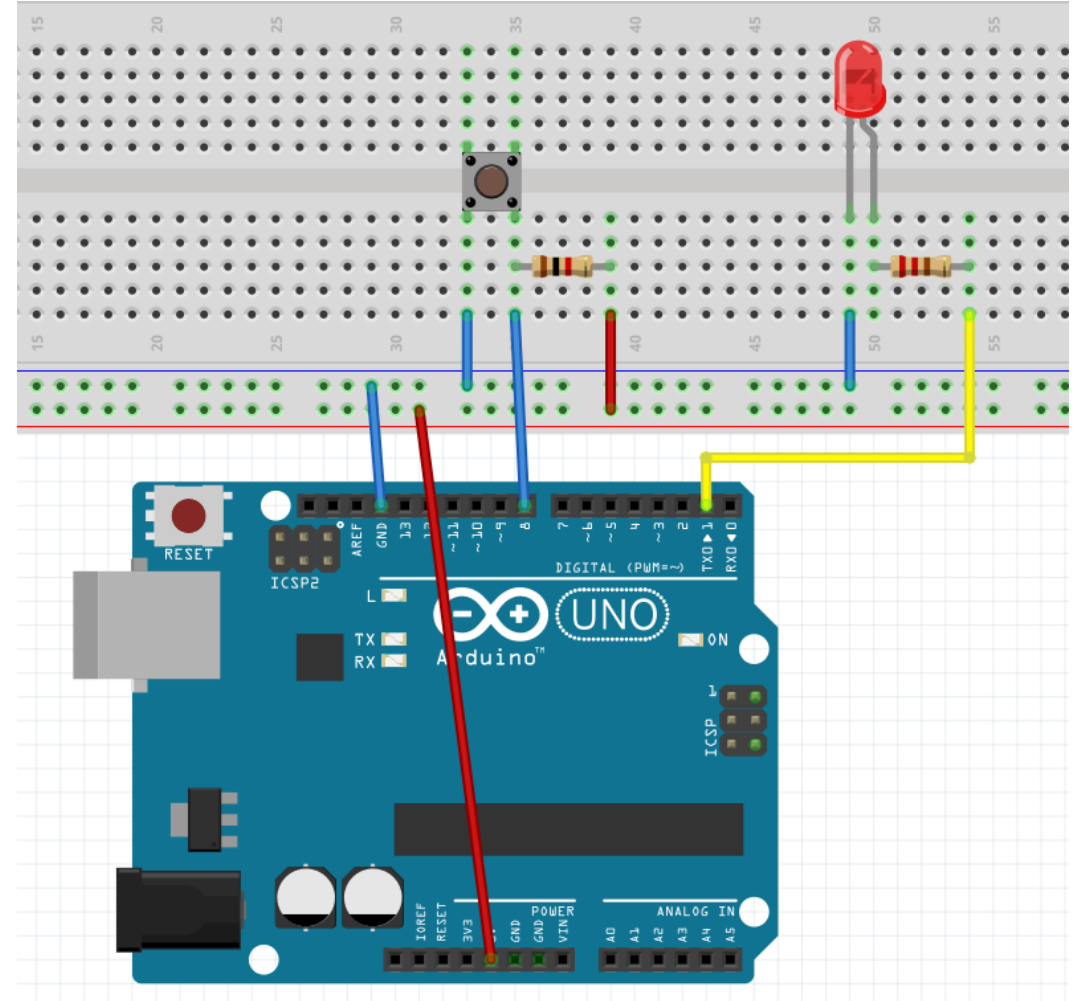


# 아두이노 기초

## 버튼 입력 받기

버튼의 ON/OFF 상태를  
digitalRead() 함수로 받아보자.

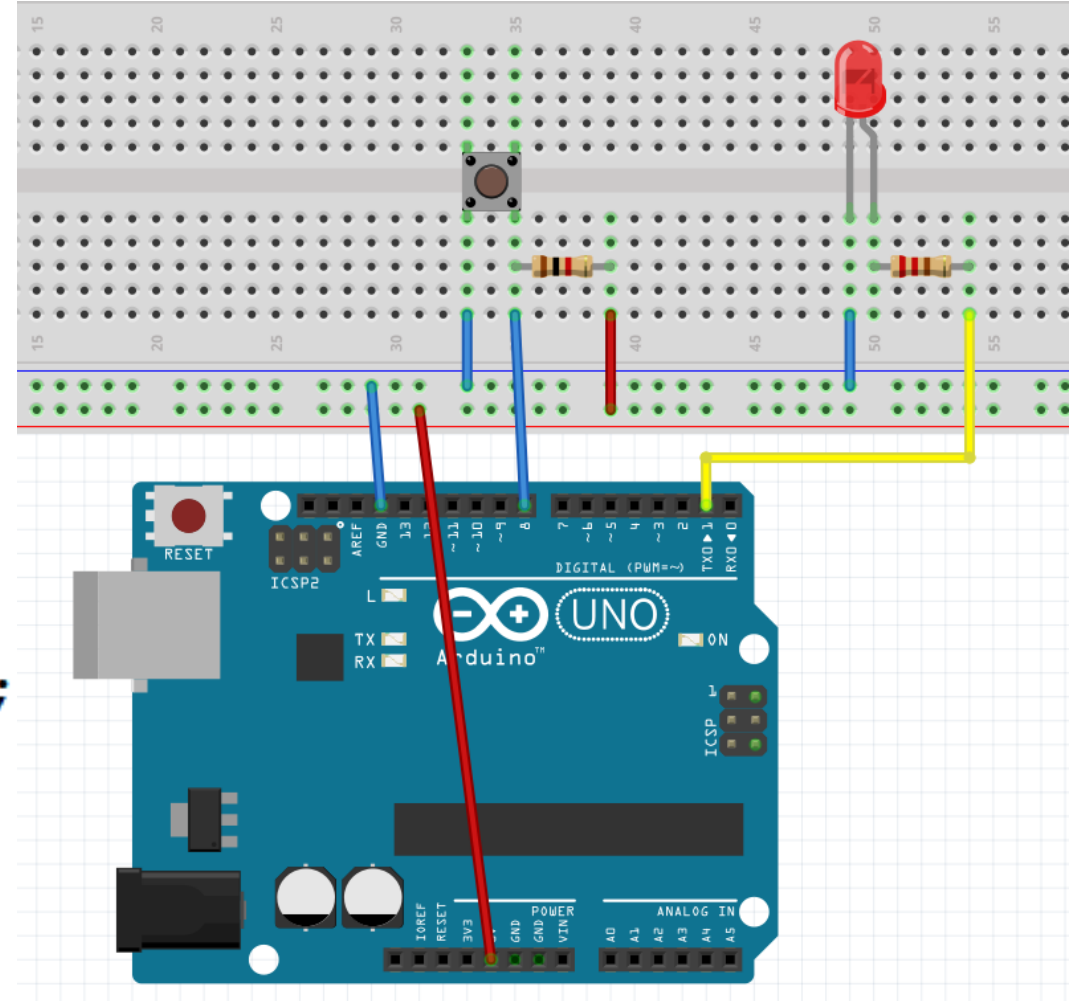
- 오른쪽과 같이 회로를 구성
- 버튼에 연결된 핀을 입력(INPUT)으로 설정
- digitalRead(8) 함수를 사용해보자.
- digitalRead(8) 에서 반환된 값을  
변수 readedValue에 저장
- 변수에 저장된 값을  
digitalWrite(1, readedValue); 출력



# 아두이노 기초

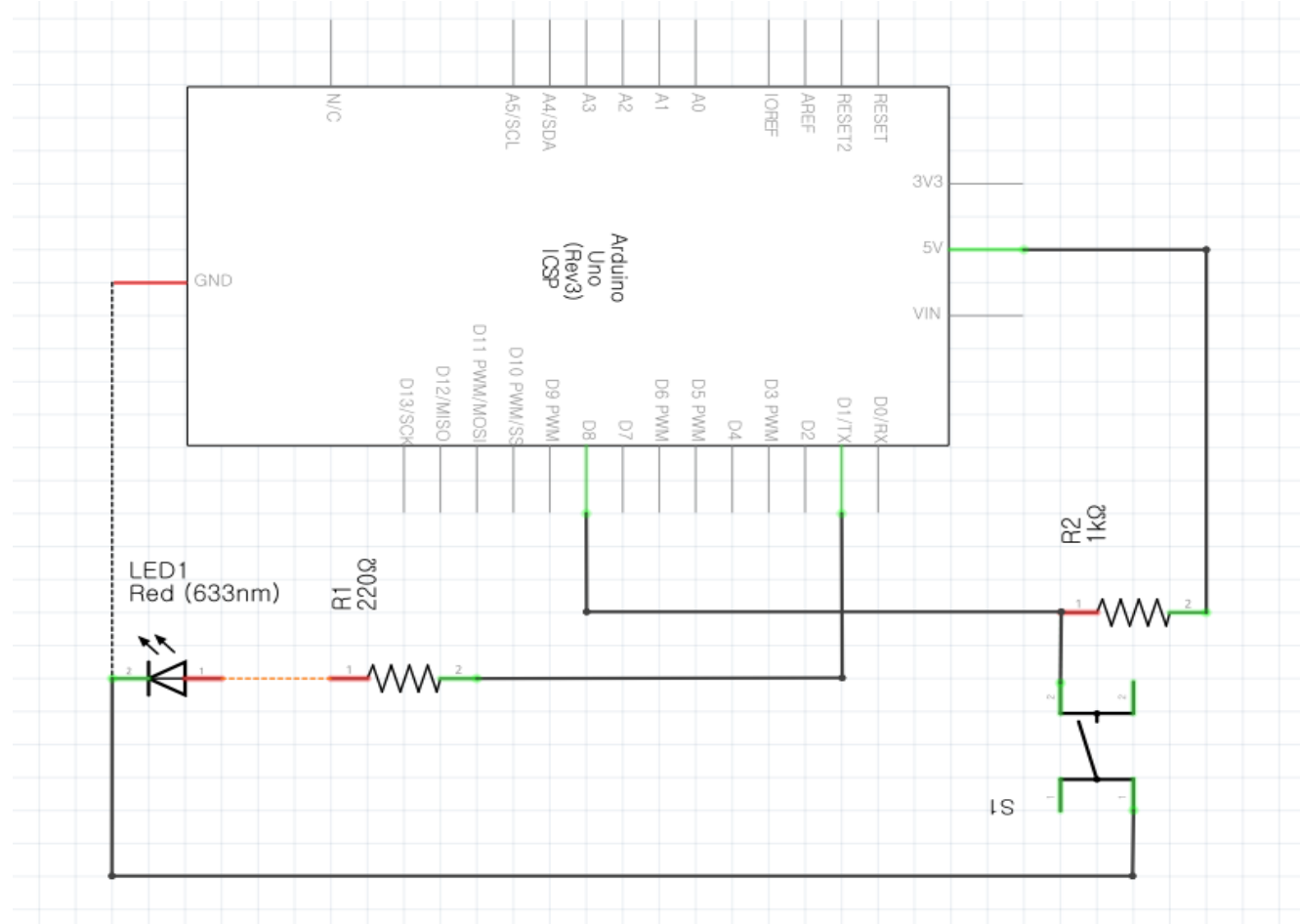
## 버튼 입력 받기

```
1 void setup() {  
2   pinMode(8, INPUT);  
3   pinMode(1, OUTPUT);  
4 }  
5  
6 void loop() {  
7   int readedValue = digitalRead(8);  
8   digitalWrite(1, readedValue);  
9 }
```



# 아두이노 기초

## 버튼 입력 받기 (회로도)





# 아두이노 기초

---

## 조건문

- 특정한 조건이 만족되었을 때에만 코드를 실행한다.
- if문, switch case문

# 아두이노 기초

## 조건문 (if문)

```
1 void setup() {  
2   pinMode(8, INPUT);  
3   pinMode(13, OUTPUT);  
4 }  
5  
6 void loop() {  
7   if (digitalRead(8) == HIGH) {  
8     digitalWrite(13, HIGH);  
9   }  
10  
11   if (digitalRead(8) == LOW) {  
12     digitalWrite(13, LOW);  
13   }  
14 }
```

// 만약 ~라면 ...

# 아두이노 기초

## 조건문 (if else문)

```
1 void setup() {  
2     pinMode(8, INPUT);  
3     pinMode(13, OUTPUT);  
4 }  
5  
6 void loop() {  
7     if (digitalRead(8) == HIGH) { // 만약 ~라면 ...  
8         digitalWrite(13, HIGH);  
9     } else if (digitalRead(8) == LOW) { // 아니면 ~라면 ...  
10        digitalWrite(13, LOW);  
11    }  
12 }
```

# 아두이노 기초

## 조건문 (if else문)

```
1 void setup() {  
2     pinMode(8, INPUT);  
3     pinMode(13, OUTPUT);  
4 }  
5  
6 void loop() {  
7     if (digitalRead(8) == HIGH) { // 만약 ~라면 ...  
8         digitalWrite(13, HIGH);  
9     }else { // 아니면 ...  
10        digitalWrite(13, LOW);  
11    }  
12 }
```

# 아두이노 기초

---

## 응용해보기1

- 버튼을 누르면 LED가 3초간 켜졌다가 꺼지는 동작을 작성하시오.

# 아두이노 기초

## 응용해보기2

- LED 6개, 버튼 1개로 주사위 게임을 구현하시오.
- 버튼을 누르면 1~6개의 LED가 랜덤한 개수로 켜진다.
- 랜덤(난수)함수:  
setup에 작성: `randomSeed(analogRead(0));`  
랜덤 생성 시: `int result = random(0, 6);` // 0~5까지의 난수 생성

# 아두이노 기초

---

## 과제 1

- 버튼 1개와 LED 1개의 회로를 구성하여 다음 동작을 작성하시오.
- 버튼 1을 누르면 LED 1이 켜진다.
- 다시 버튼 1을 누르면 LED 1이 꺼진다.
- 즉, 버튼을 누를 때 마다 LED가 켜지고 꺼진다.

# 아두이노 기초

---

## 과제2

- 버튼 2개와 LED 2개의 회로를 구성하여 다음 동작을 작성하시오.
- 버튼 1을 누르면 LED 1이 켜지고 꺼진다.
- 버튼 2를 누르면 LED 2가 켜지고 꺼진다.
- 버튼 1과 2를 동시에 누르면 모든 LED가 꺼진다.