



아주대학교

코딩 교육 봉사 동아리 SWeat x 00고등학교

2주차

아두이노 기초 문법과 LED + 스위치 제어

발표자 | 000 튜터, 000 튜터

1주차에 배운 내용 복습하기

Review



아두이노

: 물리적인 세계를 감지하고 제어할 수 있는 장치를 만들기 위한 도구

1주차에 배운 내용 복습하기

Review

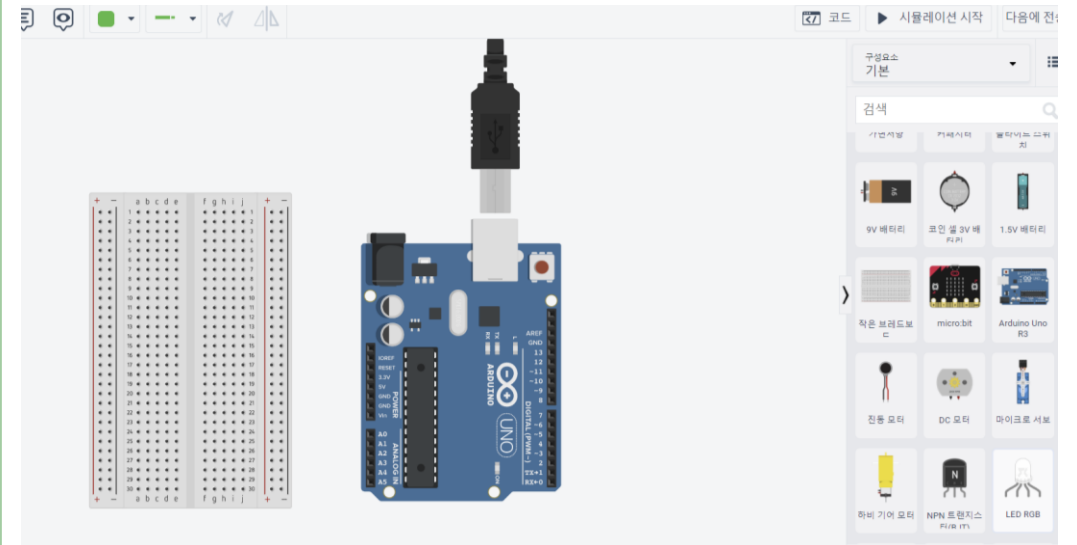
파일 편집 스케치 툴 도움말

```
sketch_jan24a
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:
}
```

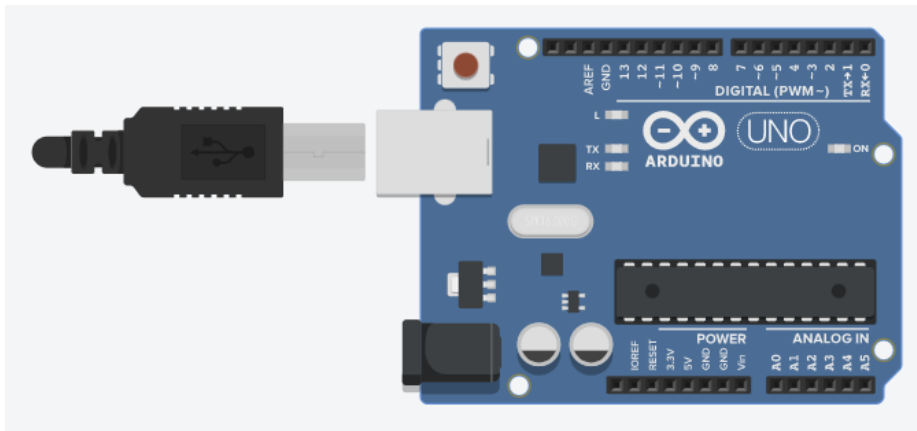
아두이노 IDE 설치하기



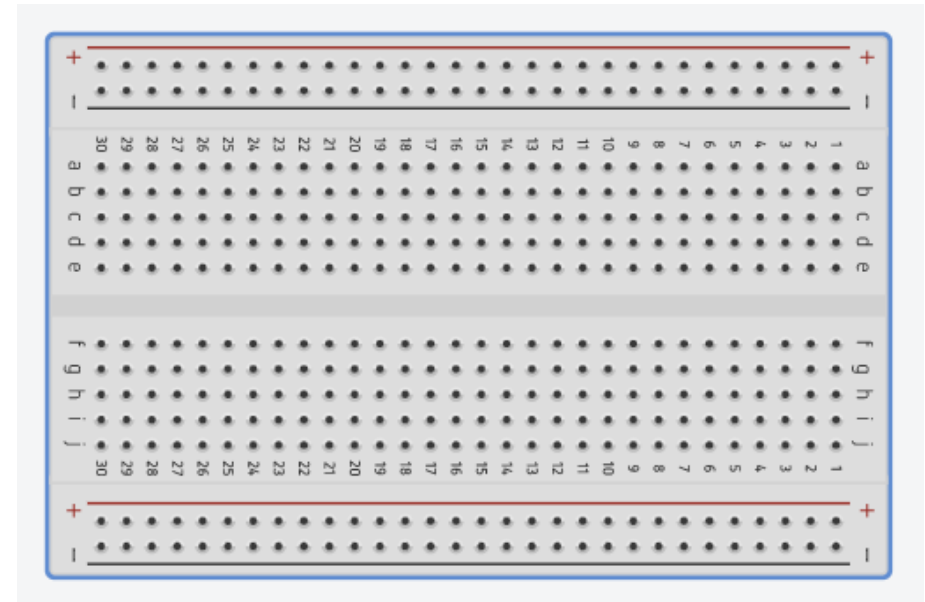
팅커캐드 접속해보기

1주차에 배운 내용 복습하기

Review



Uno 보드



브레드보드

1주차에 배운 내용 복습하기

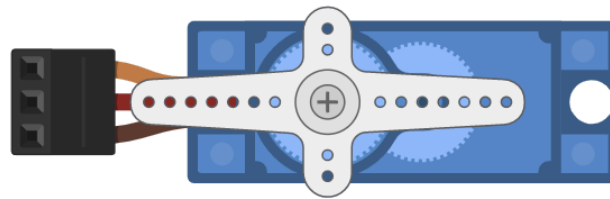
Review



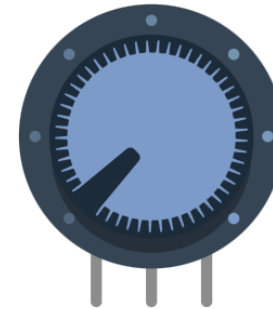
LED



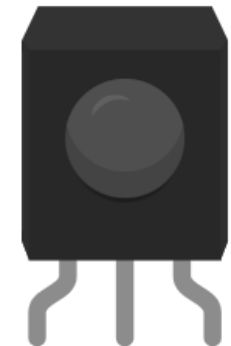
스위치



서보모터



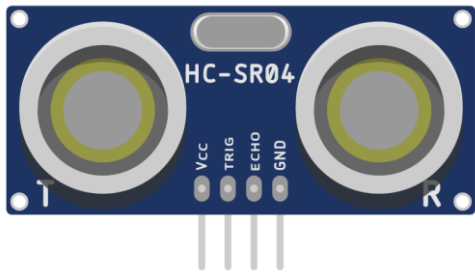
가변저항



적외선센서

1주차에 배운 내용 복습하기

Review



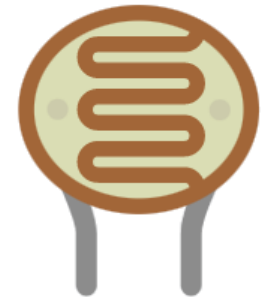
초음파센서



7세그먼트



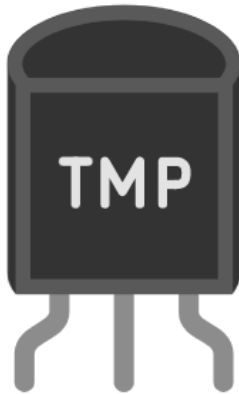
LCD



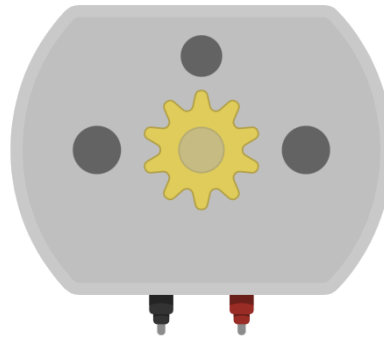
조도센서

1주차에 배운 내용 복습하기

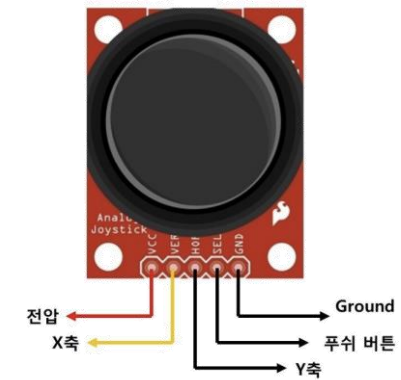
Review



온도센서



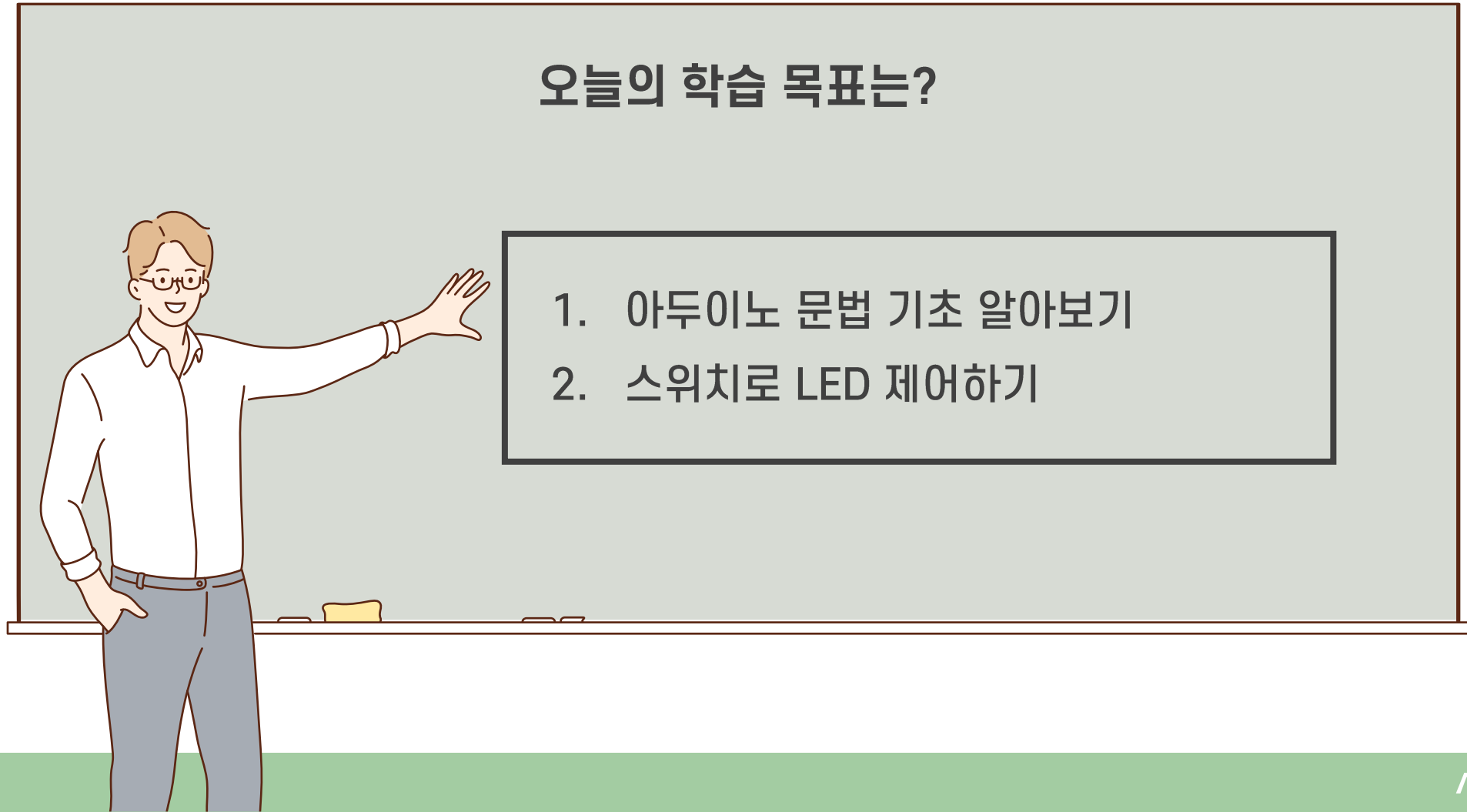
스텝모터



조이스틱

학습 목표

Learning Objectives

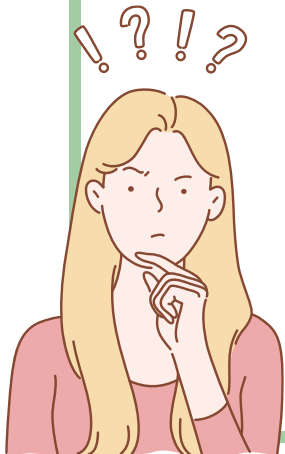


아두이노 IDE에서 사용하는 언어는 무엇일까요?

Q&A

아두이노 IDE에서는
어떤 언어를
사용하나요?

Q



아두이노 IDE에서
사용하는 언어는 C++입니다.
주로 사용할 C++의
문법을 배워볼까요?

A

실습하면서 다시 설명하므로
필요한 내용만 가볍게 알고 갑시다 ☺



C++ 문법 살펴보기1 : 변수 선언

아두이노 문법 기초

※ C++는 코드 구문이 끝날 때마다 ; (세미콜론)을 붙여 코드 한 구문이 끝났음을 명시해 주어야 합니다.

변수를 사용하려면 먼저 변수를 선언하고 값을 할당해주어야 합니다.

자료형

정수형	int
	char (문자1글자)
실수형	float
논리형	bool(참/거짓)

자료형 변수명; // 변수 선언
자료형 변수명 = 값; // 변수 선언 및 초기화 <추천방법!!>
변수명 = 값; // 초기화 및 대입(값 변경)

```
1 int ledButton = 13; // 정수형 ledButton 변수 선언 및 초기화
2 ledButton = 14; // ledButton 변수값 14 할당 (값 변경)
3
4 char alphabetA= 'a'; // 자료형(1글자) 변수 alphabetA 선언 및 초기화
5 // 문자는 따옴표로 묶어야 한다.
6
7 bool a = 0; // 논리형 변수 a 선언 및 초기화 0은 false와 같다.
8 bool b = false; // 논리형 변수 b 선언 및 초기화 0은 false와 같다. (a=b)
9 bool c = 1; // 논리형 변수 c 선언 및 초기화 1은 true와 같다.
10 bool d = true // 논리형 변수 d 선언 및 초기화 1은 true와 같다. (c=d)
11
12 float pi = 3.14 // 실수형 변수 pi 선언 및 초기화 (double도 쓸 수 있다.)
```

변수 선언이란?

어떤 변수를 사용할 것인지
변수 이름을 짓고 자료형을
매칭하는 것입니다.

초기화란?

처음으로 가질 값을 할당해
주는 것입니다. 초기화를
해주어야 변수를 사용할 수
있습니다.

C++ 문법 살펴보기2 : 조건문 < if 문 >

아두이노 문법 기초

```
1  if ( /* 조건1 */ ){
2
3      /* 조건 1이 참이면 실행 */
4
5  } else if ( /* 조건2 */ ){
6
7      /* 조건 1이 거짓이고,
8       조건 2가 참이면 실행 */
9
10 } else if ( /* 조건3 */ ){
11
12     /* 조건 1이 거짓이고,
13     조건 2도 거짓이고,
14     조건 3이 참이면 실행 */
15
16 } else{
17
18     /* 조건 1, 2, 3 모두 거짓이면 실행 */
19
20 }
```

if(조건식), else if(조건식), else :

if()문은 괄호 안의 조건이 참(true)이 될 때 { }
안의 코드를 실행하는 조건문 함수입니다.

조건이 참이 아닌 경우 다음 조건문(else if()
또는 else)으로 넘어가며 else if()의 수는 제한
이 없습니다.

앞의 모든 조건을 만족하지 않을 때(거짓, false)
else 문이 실행됩니다. 이때, else 문이 없는 경우
다음 코드로 넘어갑니다.

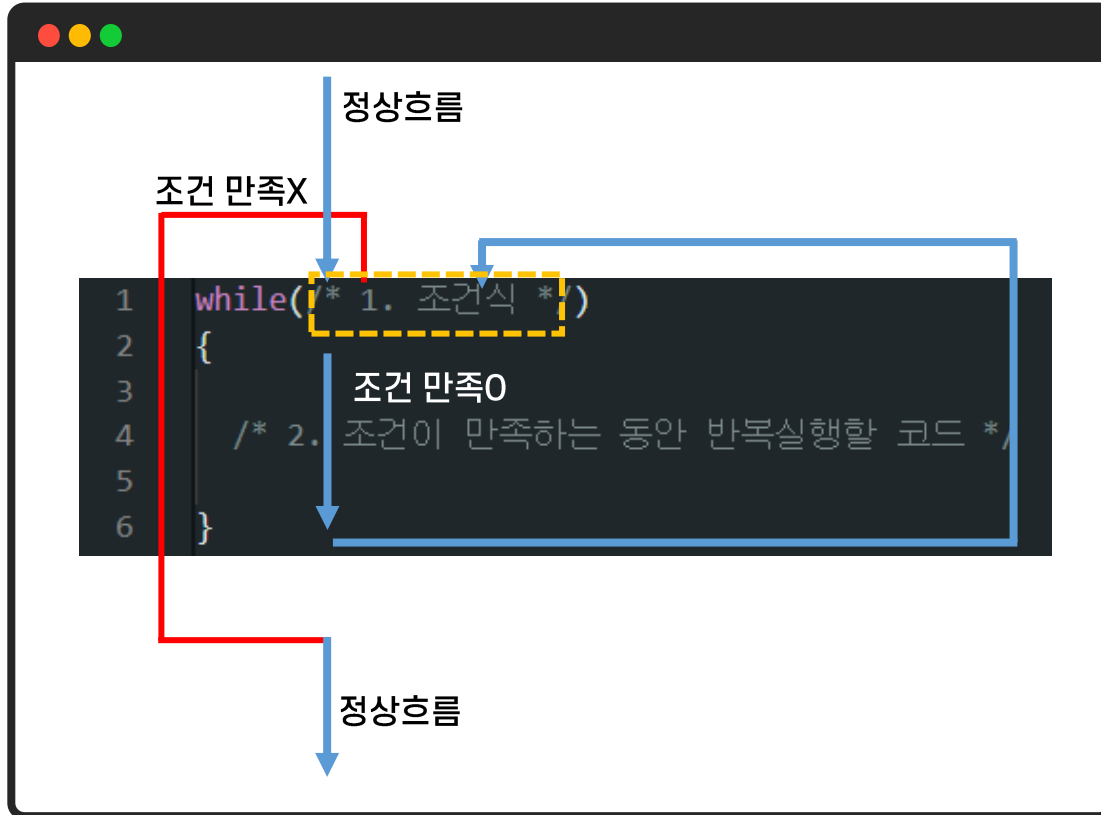
※ if()문을 단독으로 사용하는 것은 가능하지만, else if(), else문을 단독
으로 사용하는 것은 불가하다.

C++ 문법 살펴보기3 : 반복문 < while()문 >

아두이노 문법 기초

💡 Tip : while()문 예시

Q. 0부터 10까지 숫자를 화면에 출력!
while(스위치가 눌러있는가?)
{ led를 켜라! }



반복문이란, 반복적으로 동작해야 하는 명령어들을 모아 반복 조건을 설정하여 동작시키는 코드 구조입니다.

while(조건식):

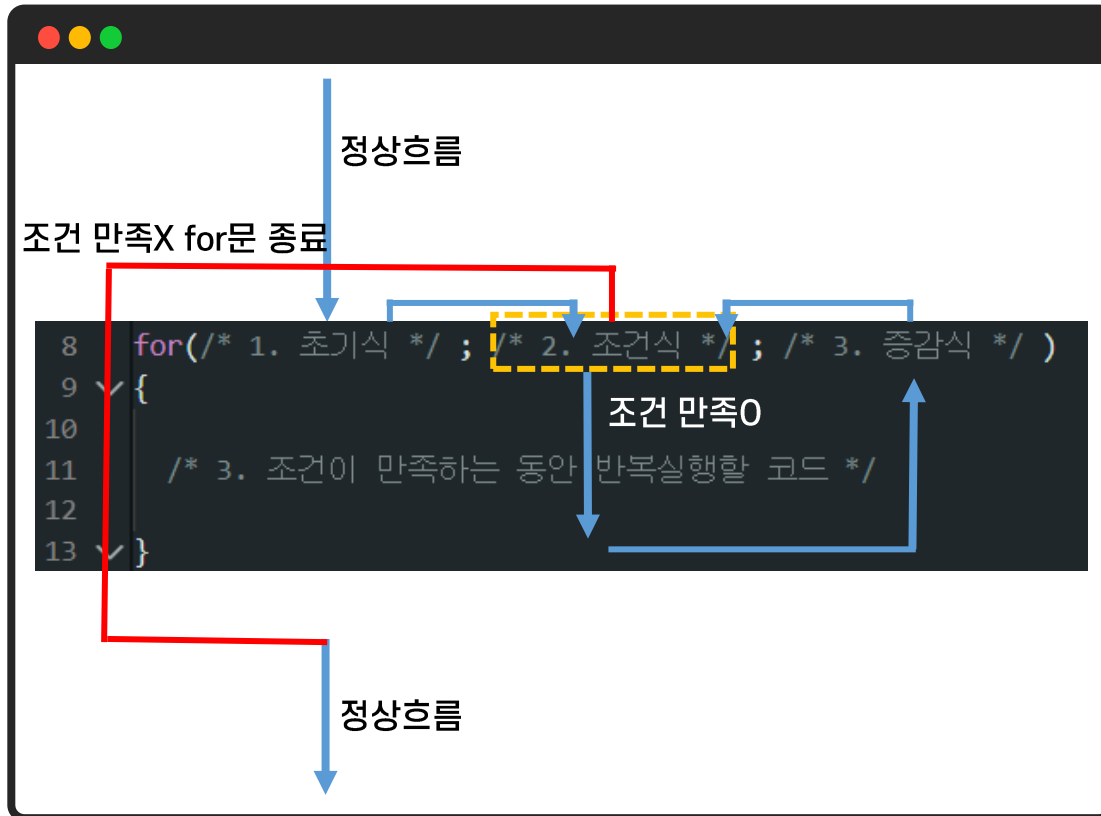
괄호 속 **조건을 만족하는 동안** { }안의 코드가 **반복 실행**됩니다. break문을 통해 중지할 수도 있습니다.

C++ 문법 살펴보기3 : 반복문 < for()문 >

아두이노 문법 기초

💡 Tip : for()문 예시

Q. 0부터 10까지 숫자를 화면에 출력!
for(int i = 1; i < 11; i ++)
{ println(i); } // println은 출력함수!



반복문이란, 반복적으로 동작해야 하는 명령어들을 모아 반복 조건을 설정하여 동작시키는 코드 구조입니다.

for(초기식 ; 조건식 ; 증감식):

`while()`문과 같이 조건이 만족하는 동안 { }안의 코드가 반복 실행되며, 이에 더해 초기설정식과, 증감식 정보도 포함됩니다. (정보의 구분은 ; (세미콜론))

아두이노 코드 구조 살펴보기

아두이노 문법 기초

💡 Tip : 함수 구조

```
반환형 함수명( 매개변수1, ... ) {  
    // 실행할 코드  
}
```

```
1 void setup() {  
2  
3     pinMode(13, OUTPUT);  
4  
5 }  
6  
7 void loop() {  
8  
9     digitalWrite(13, HIGH);  
10    delay(1000);  
11    digitalWrite(13, LOW);  
12    delay(1000);  
13  
14 }
```

setup() :

아두이노를 사용하기 위해 초기설정을 하는 함수입니다. {} 안의 코드를 처음 한 번만 실행합니다.

loop() :

setup()이 실행된 이후 {} 안의 코드를 반복적으로 실행하는 함수입니다.

※ 함수 앞의 **void** 는 이 함수가 아무것도 반환하지 않도록 한다.

※ 코드 구문이 끝날 때마다 ; (세미콜론)을 붙여 코드 한 구문이 끝났음을 명시해 주어야 한다.

아두이노 함수 살펴보기

아두이노 문법 기초

```
1 void setup() {  
2  
3     pinMode(13, OUTPUT);  
4  
5 }  
6  
7 void loop() {  
8  
9     digitalWrite(13, HIGH);  
10    delay(1000);  
11    digitalWrite(13, LOW);  
12    delay(1000);  
13  
14 }
```

pinMode(핀 번호, 모드) :

해당 핀을 입력으로 사용할 것인지 출력으로 사용할 것인지 알려줍니다.

INPUT, OUTPUT :

INPUT은 입력모드, OUTPUT은 출력모드를 의미합니다.

※ pinMode(13, OUTPUT)
: 13번 핀을 출력 핀으로 설정한다는 의미이다.

아두이노 함수 살펴보기

아두이노 문법 기초

```
11 void loop()
12 {
13   int buttonState = digitalRead(buttonPin);
14
15   if(buttonState == HIGH) {
16     digitalWrite(ledPin, HIGH);
17   } else {
18     digitalWrite(ledPin, LOW);
19   }
20
21   delay(10);
22 }
23
```

digitalRead(핀 번호) :

해당 디지털 핀의 상태를 읽어오는 명령어입니다.

반환값: HIGH(=1) 또는 LOW 값 (=0)

digitalWrite(핀 번호, 값) :

해당 디지털 핀에 HIGH(전기 공급) 또는 LOW(전기 공급 안 함)의 값을 주는 명령어입니다.

※ digitalWrite(5, HIGH)

: 5번 핀에 전기를 공급한다는 의미이다.

아두이노 함수 살펴보기

아두이노 문법 기초

```
1 void loop() {  
2     int analogValue = analogRead(0);  
3  
4     analogWrite(11, analogValue/4);  
5  
6     delay(1000);  
7 }
```

analogRead(핀 번호) :

아두이노 보드의 아날로그 핀에 연결된 값을 읽어옵니다. 아날로그 값은 0부터 1023까지 1024단계로 측정합니다.

반환값: 0~1023 정수 값(int)

analogWrite(핀 번호, 값):

아날로그 값을 핀에 출력한다. 펄스 폭 조절(PWM) 기능을 지원하는 디지털 핀(아두이노 보드에서 ~ (물결) 표시가 되어 있는 핀) 번호와 0부터 255까지 입력 값을 지정하여 사용합니다.

delay(시간) :

일정 시간만큼 시간을 지연시키는 함수입니다.

시간에 들어갈 값은 ms단위 (1/1000초)

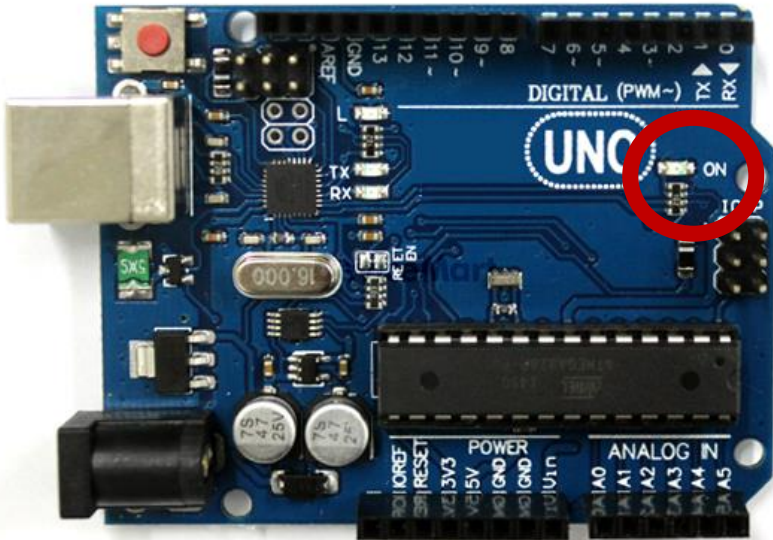
※ analogWrite(11, 250): 11번 핀에 250의 입력 값을 준다.

실습을 준비해볼까요?

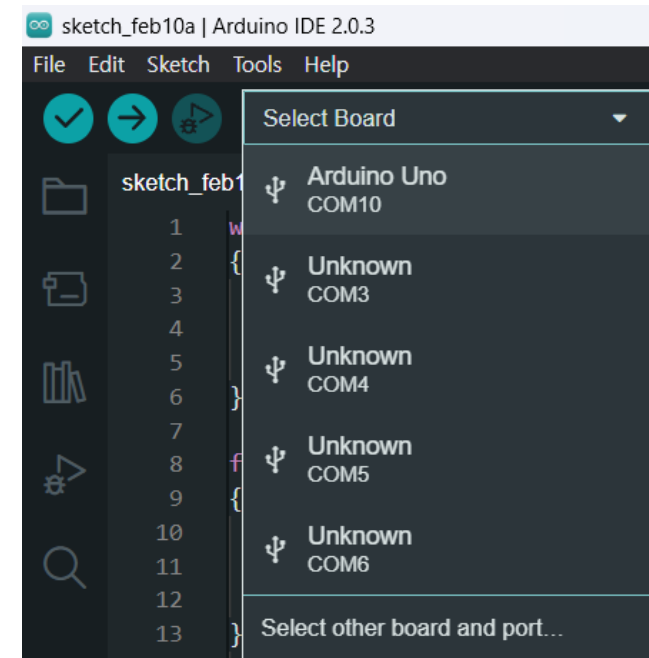
LED 제어하기

매번 실습 때 해주어야 해요 ~

컴퓨터에 아두이노 보드를 연결하고, 아두이노 IDE에서 연결된 Arduino Uno 보드와 포트번호를 선택합니다.

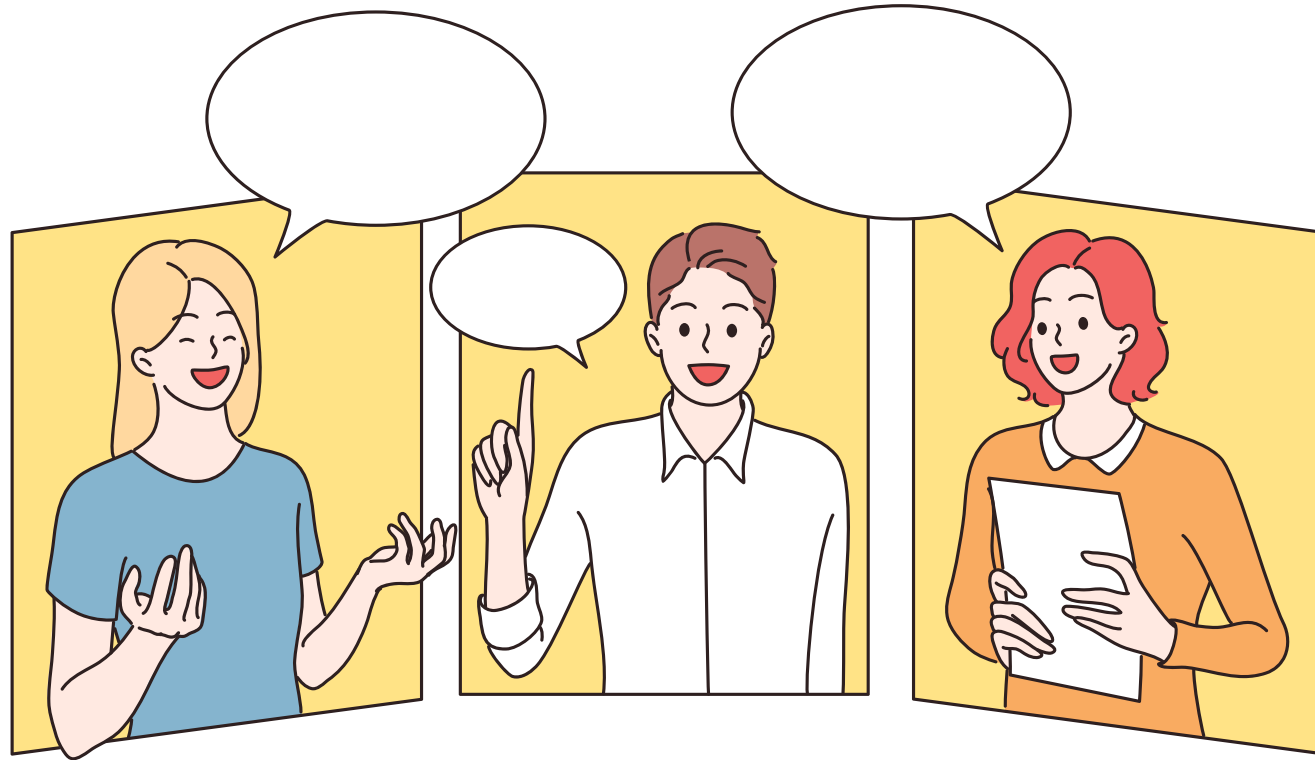


ON에 불이 들어와야 해요.



보드
포트번호

잠깐 쉬는 시간 !



아두이노 연결이 안되는 친구들은 쉬는 시간에 도움을 요청해주세요 ~ :D

<실습1> LED 제어하기

LED 제어하기

실습 준비물

LED x 1



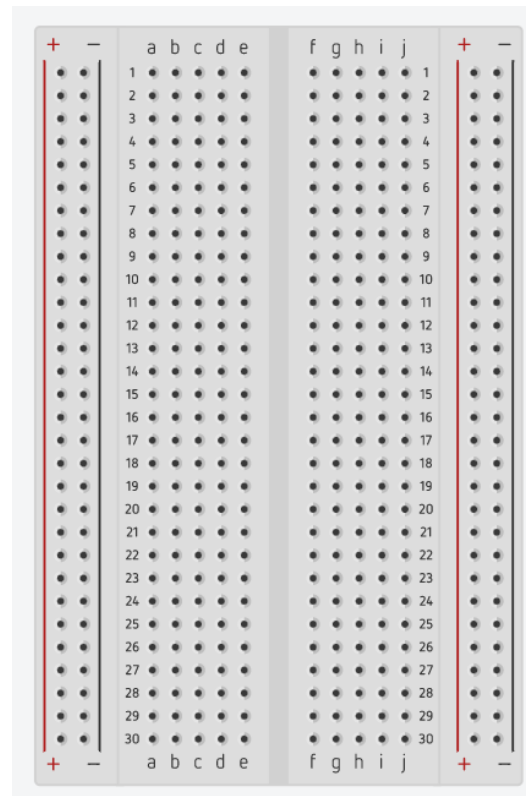
저항(220k옴) x 1



점퍼선 x 3

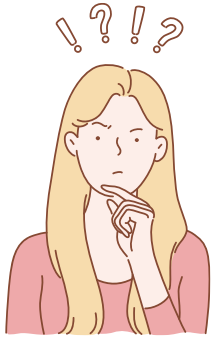


브레드보드 x 1



잠깐! 실습 전 Q&A

질문이 있어요!



점퍼선의 색깔을 꼭 똑같이 맞추어야 하나요?

Q

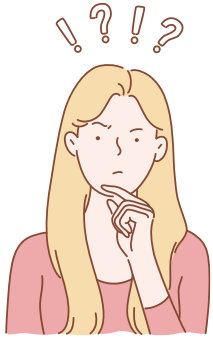
A

점퍼선의 색깔을 회로도와 똑같이 맞출 필요는 없습니다. 다만, 이해를 돕기 위해 일반적으로 +, 5V와 같은 전원선은 **빨간색 점퍼선**, - 또는 GND(Ground)는 **검은색 점퍼선**을 사용합니다. 😊



잠깐! 실습 전 Q&A

질문이 있어요!



저항이란 무엇인가요?

Q

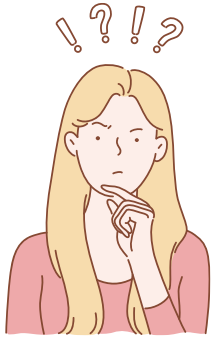
A

저항은 전류의 양을 제어합니다. 너무 많은 양의 전류가 흐를 경우 장치가 고장이 날 수 있어 이를 제어하는 역할을 합니다.

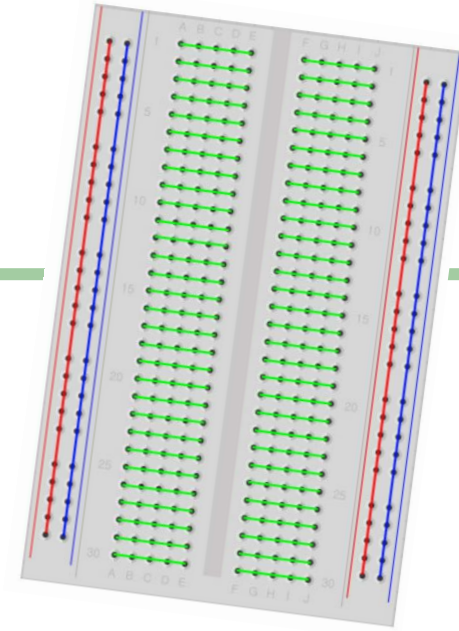


잠깐! 실습 전 Q&A

질문이 있어요!



브레드보드에 꽂을 때 유의할 점이 무엇인가요?



Q

A

A. 회로도와 실제 브레드보드가 달라 보기 어려울 시, 회로도와 똑같이 맞춰 꽂을 필요는 없습니다. 다만, A~E, 와 F~J의 줄의 경우 그림과 같이 가로로 이어져 있으므로 가로선상에 맞게 연결합니다. +/-는 극에 맞게 꽂으면 됩니다. 가운데 선을 기준으로 양쪽은 연결되어 있지 않으니 이점 유의하며 실습을 진행해 봅시다. 😊

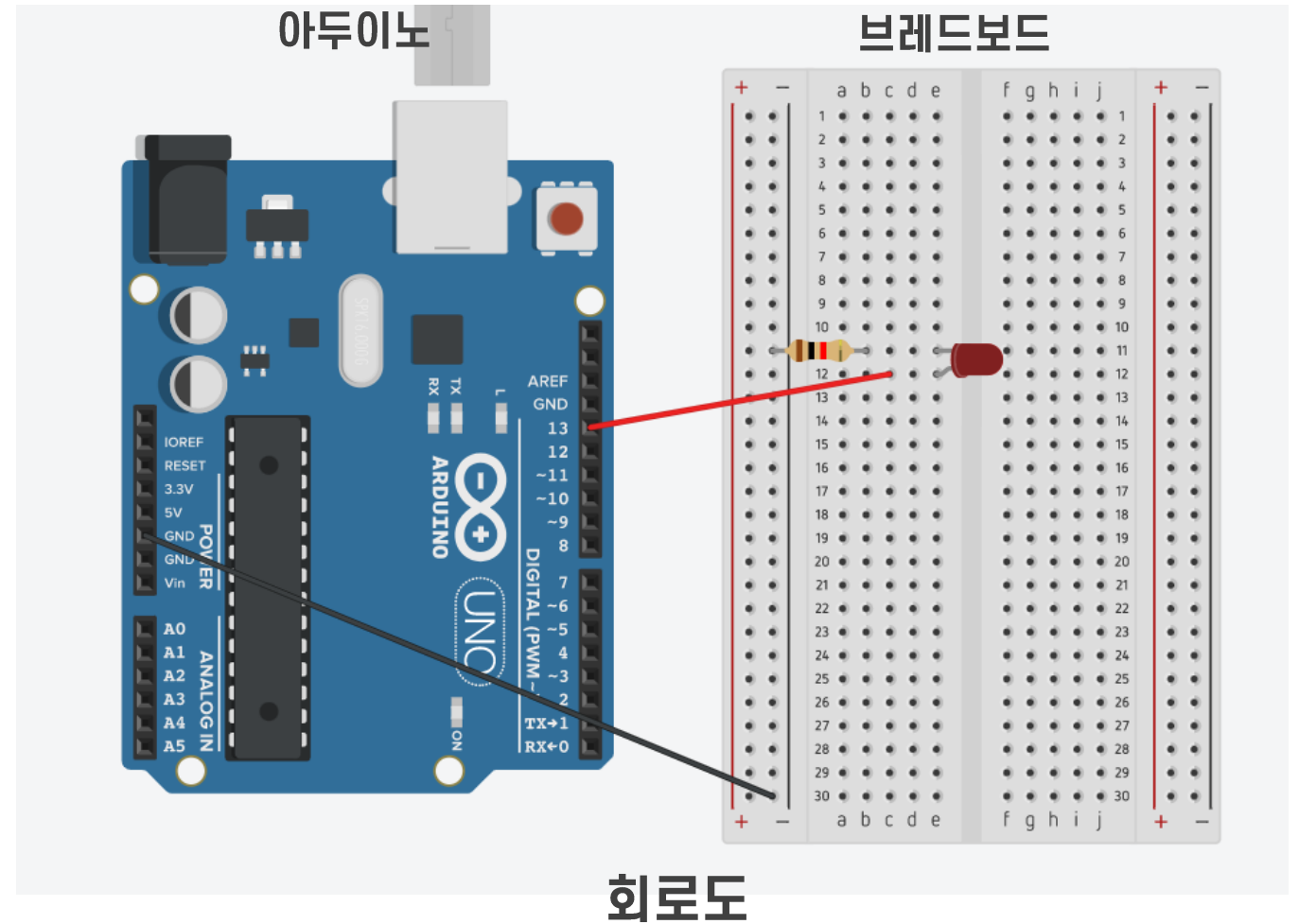


<실습1> LED 제어하기: 아두이노에 LED 1개를 연결하고 1초 간격으로 켜졌다가 꺼지게 해봅시다.

LED 제어하기

이렇게 해봐요!

1. 브레드보드에 LED 1개를 연결합니다.
2. LED의 -핀과 같은 줄에 저항 1개를 연결합니다.
3. 저항의 한 쪽 핀과 아두이노 보드의 전원 영역의 GND를 검은색 점퍼선으로 연결합니다. (각각을 -극에 연결합니다)
4. LED +핀과 아두이노 보드의 디지털 13번 핀을 빨간색 점퍼선으로 연결합니다.



스케치 작성

Sketch

1. 디지털 13번 핀을 출력모드로 설정합니다. 이제 디지털 핀 13번에서 전기를 공급할 수 있습니다.
2. 디지털 13번 핀에 전기를 줍니다. (HIGH) 전기를 끊으려면 LOW를 입력합니다.
3. 1000 ms, 즉, 1초 동안 기다립니다

사용된 함수

pinMode(핀 번호, 모드)

: 해당 핀을 출력(OUTPUT) 또는 입력(INPUT) 모드로 설정한다.

digitalWrite(핀 번호, 값)

: 해당 디지털 핀에 HIGH(전기공급) 또는 LOW(전기차단)의 값을 주는 명령어

delay(시간)

: 일정 시간만큼 시간을 지연시키는 함수 (단위: 1ms = 1/1000초)

```
1 void setup()
2 {
3   pinMode(13, OUTPUT);
4 }
5
6
7 void loop()
8 {
9   digitalWrite(LED_BUILTIN, HIGH);
10
11  delay(1000); // Wait for 1000 millisecond(s) = 1초
12
13  digitalWrite(LED_BUILTIN, LOW);
14
15  delay(1000); // Wait for 1000 millisecond(s) = 1초
16 }
```

Review: setup(), loop()

setup()은 초기설정으로 한번만 실행되며, loop()는 반복 실행된다.

<실습2> 스위치 + LED 제어하기 (실습1과 이어집니다. 저항 1개, 푸쉬버튼 1개, 점퍼선 2개 추가)

스위치 + LED 제어하기

실습 준비물

LED x 1



브레드보드 x 1

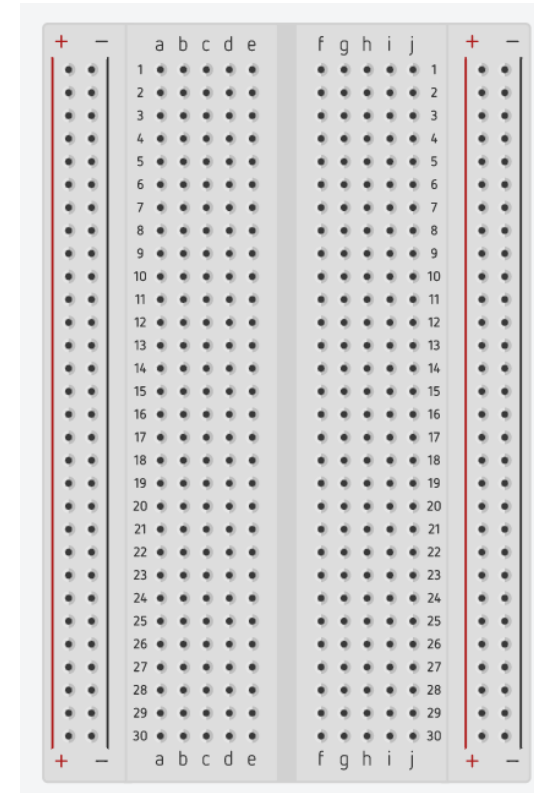
저항(220k옴) x 2



푸쉬버튼 x 1



점퍼선 x 5

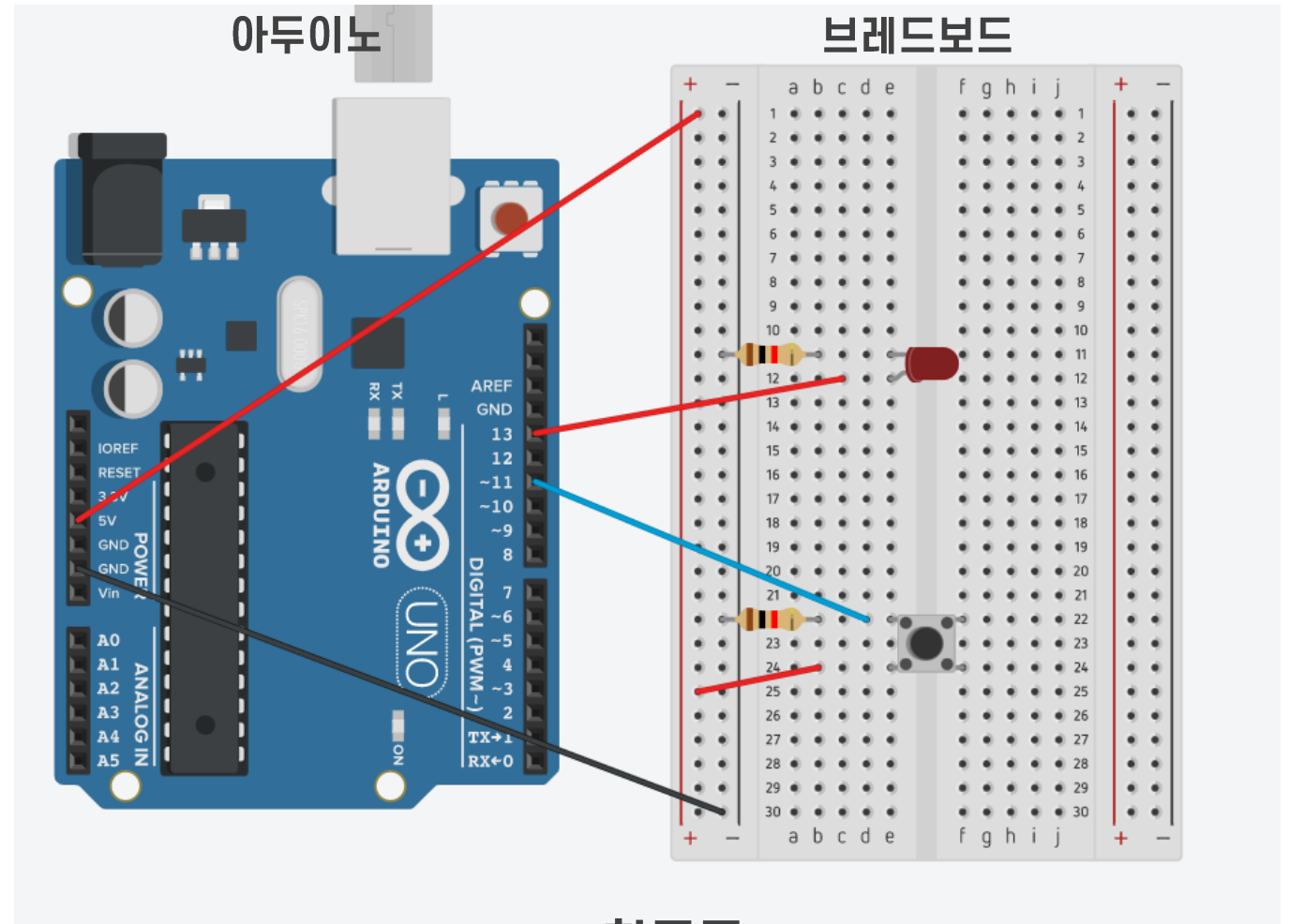


<실습2> 스위치 + LED 제어하기: 푸시버튼을 누르면 LED가 켜지도록 만들어 봅시다.

스위치 + LED 제어하기

이렇게 해봐요!

1. 실습1의 과정을 완료합니다.
2. 아두이노 보드의 전원영역의 5V에 +선을 연결합니다.
3. 브레드 보드에 푸시버튼을 연결한 후, 왼쪽 위에 있는 핀을 저항을 이용해 -선에 연결합니다. 그 사이에서 점퍼선을 이용하여 아두이노 보드의 디지털 11번 핀에 연결합니다. 왼쪽 아래의 핀은 점퍼선을 이용하여 +선에 연결합니다.



회로도

스케치 작성

Sketch

1. 변수 선언을 통해 buttonPin이라는 변수에 11번 핀 번호를 할당하고, ledPin 변수에 13번 핀 번호를 할당합니다.
2. 변수 선언과 동시에 buttonPin의 핀에서 digitalRead()로 읽어온 값을 대입합니다
3. 읽어온 값이 1 (HIGH) 이면(버튼을 눌렀다면) led에 HIGH를, 0 (LOW) 이면 LOW값을 줍니다.

사용된 함수

pinMode(핀 번호, 모드)

: 해당 핀을 출력(OUTPUT) 또는 입력(INPUT) 모드로 설정한다.

digitalRead(핀 번호, 값)

: 해당 디지털 핀에 HIGH(전기공급) 또는 LOW(전기차단)의 값을 주는 명령어

```
1  int buttonPin = 11;
2  int ledPin = 13;
3
4  void setup()
5  {
6      pinMode(buttonPin, INPUT);
7      pinMode(ledPin, OUTPUT);
8  }
9
10
11 void loop()
12 {
13     int buttonState = digitalRead(buttonPin);
14
15     if(buttonState == HIGH) {
16         digitalWrite(ledPin, HIGH);
17     } else {
18         digitalWrite(ledPin, LOW);
19     }
20
21     delay(10);
22 }
23
24
25
```

Tip : 비교연산자

A == B : 등호역할, A와 B가 같은가?

(주의!! ' = ' 은 값을 대입하는 기호!)

A >= B : a가 b보다 크거나 같은가?

A > B : a가 b보다 큰가?

A != B : a와 b가 다른가?

참(true, 1)/거짓(false, 0)을 반환한다.

Review: 조건문 if()

If(조건) {

// 조건이 참일 때 실행될 코드

} else {

// 조건이 참이 아니면 실행될 코드

}

Quiz를 맞춰 보아요!

오리엔테이션

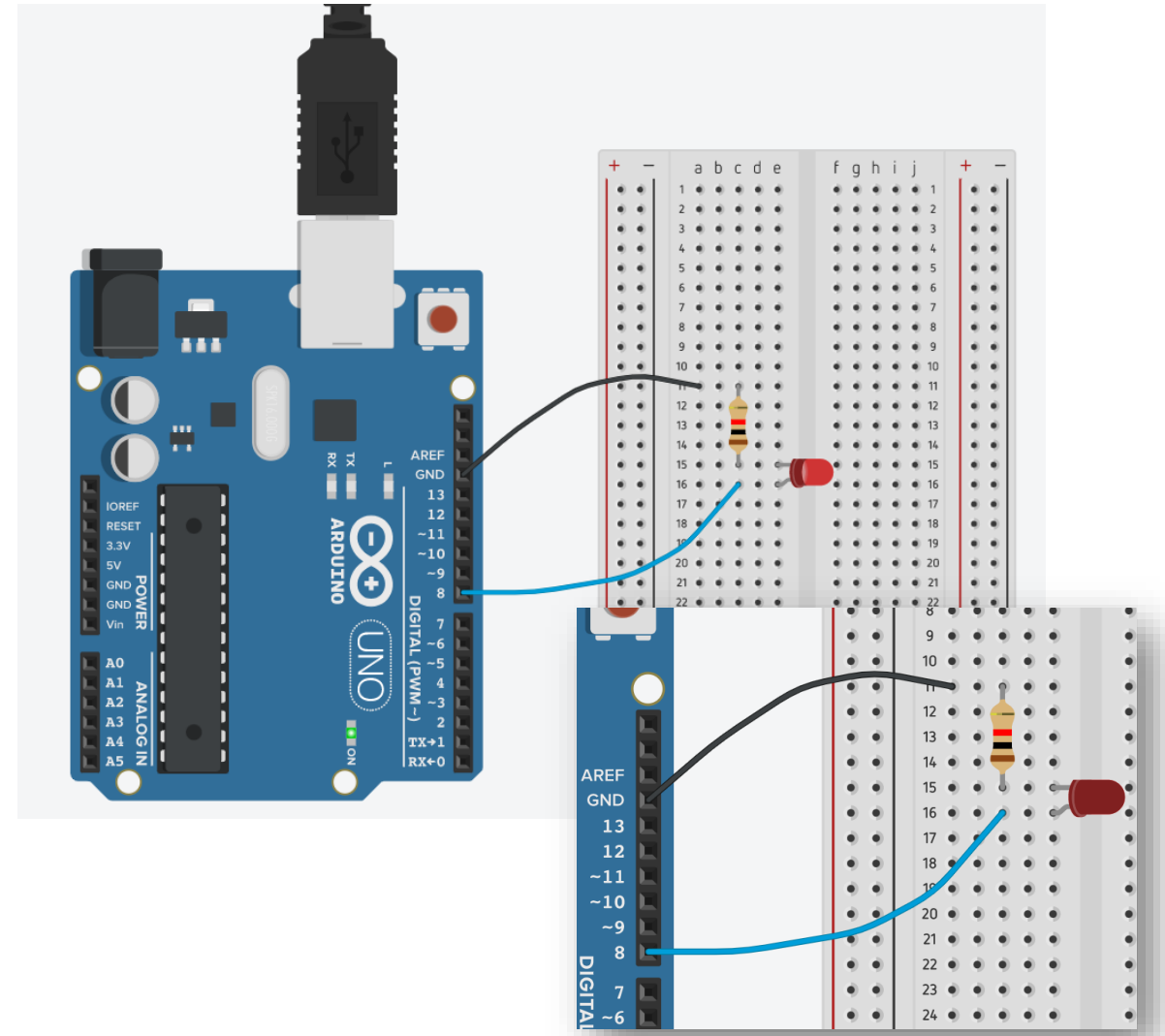


오늘 배운 내용을
잘 이해했는지
퀴즈를 풀어보아요 !

Quiz. LED가 1초 동안 켜지고 2초 동안 꺼지도록 만든 회로도입니다. 빈칸에 들어갈 숫자는 무엇일까요?

```
1 void setup()
2 {
3   pinMode( ① , OUTPUT);
4 }
5
6 void loop()
7 {
8   digitalWrite( ① , HIGH);
9
10  delay( ② );
11
12  digitalWrite( ① , LOW);
13
14  delay( ③ );
15 }
```

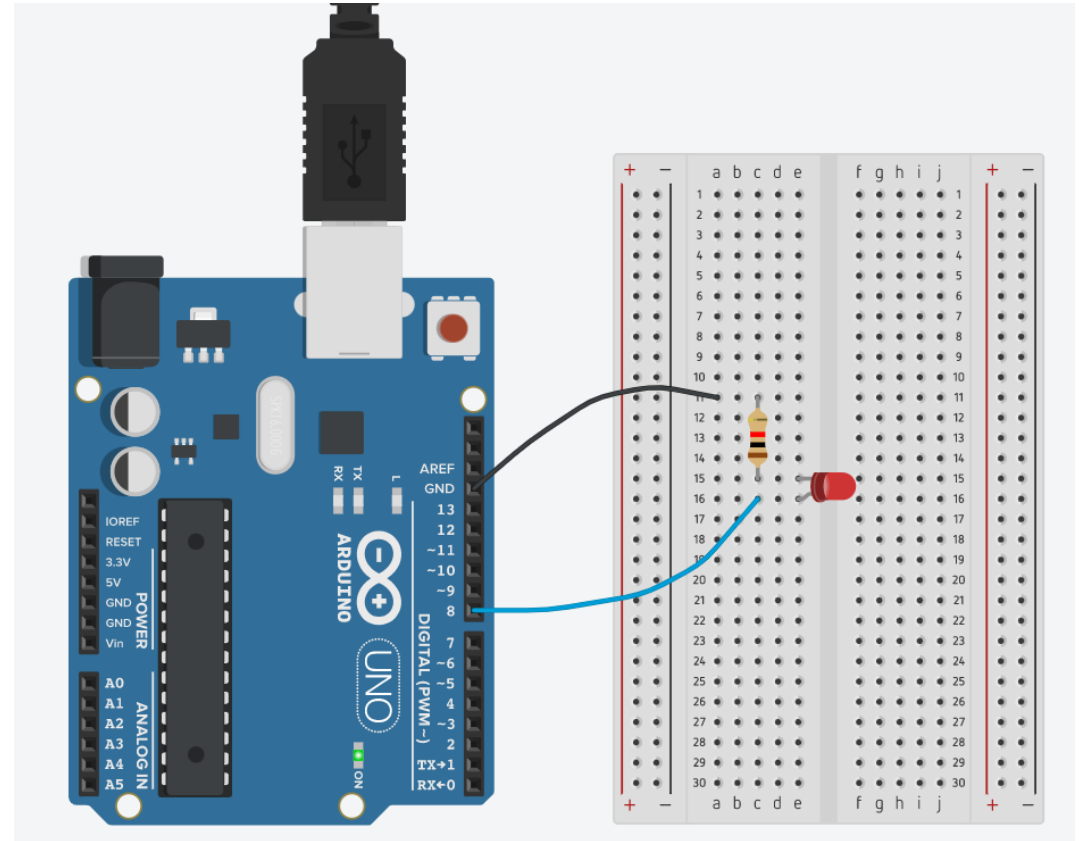
① : , ② : , ③ :



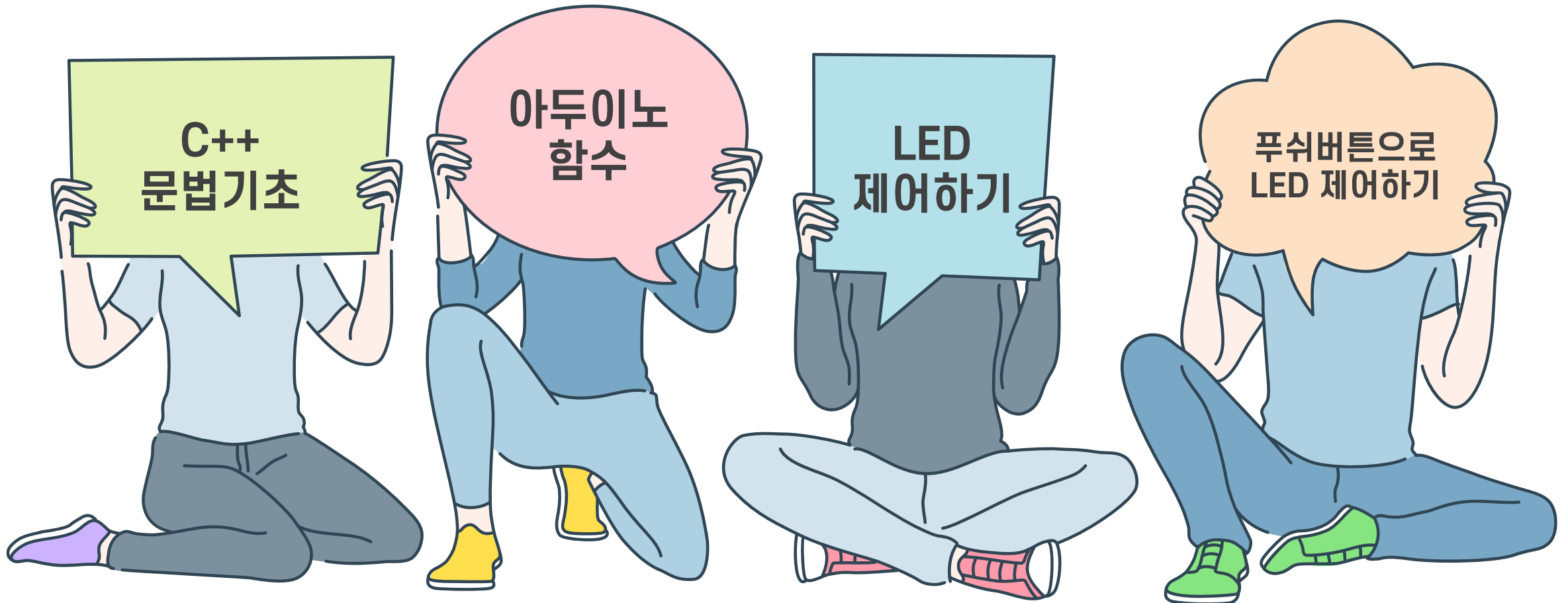
Quiz. LED가 1초 동안 켜지고 2초 동안 꺼지도록 만든 회로도입니다. 빈칸에 들어갈 숫자는 무엇일까요?

```
1 void setup()
2 {
3   pinMode( 8 , OUTPUT);
4 }
5
6 void loop()
7 {
8   digitalWrite( 8 , HIGH);
9
10  delay( 1000 );
11
12  digitalWrite( 8 , LOW);
13
14  delay( 2000 );
15 }
```

정답: ① : 8, ② : 1000, ③ : 2000



오늘 수업 Point!



오늘 배운 내용에서 질문 있나요?

