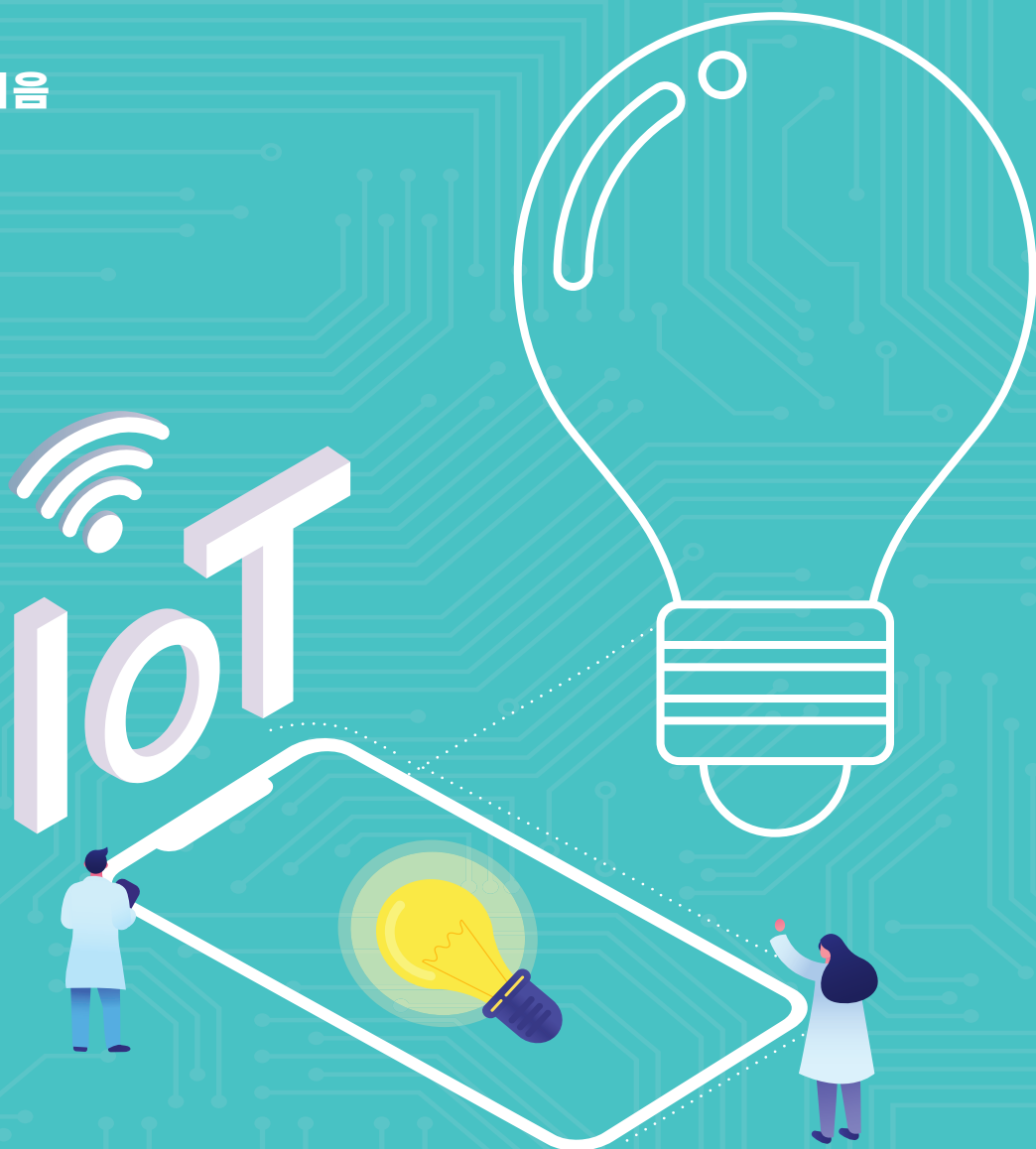


아두이노 IoT Smart Moodlight Kit 사용설명서

WEKIT 지음



아두이노 IoT 스마트 무드등 목차

구성품 확인하기	3
회로도 연결하기	4
아두이노 설치 및 소스 코드 업로드하기	6
아두이노 통합개발환경(IDE)	6
아두이노 통합개발환경(IDE) 설치	6
아두이노 통합개발환경(IDE) 기본 세팅	7
아두이노 소스 코드 업로드	9
블록코딩으로 코딩하기	13
mBlock 프로그램 설치	13
mBlock 프로그램 기본 세팅	14
mBlock 스마트 무드등 확장자 설치	17
mBlock 블록코딩	18
mBlock 블록코딩 업로드	21
APP 연결하기	22
안드로이드폰 버전	22
아이폰 버전	24
스마트 무드등 완성하기	25



구성품 확인하기

무드등 키트 구성품을 한눈에 살펴보아요!

1. 아두이노 IoT 스마트 무드등 키트 구성



1 무드등 외관

2 점퍼선 2종

3 브레드보드

4 WK 아두이노 나노 호환보드

5 블루투스 모듈

6 네오픽셀 LED 모듈

7 반투명 종이

8 양면 테이프

9 전원 어댑터



시작 전 확인하세요!

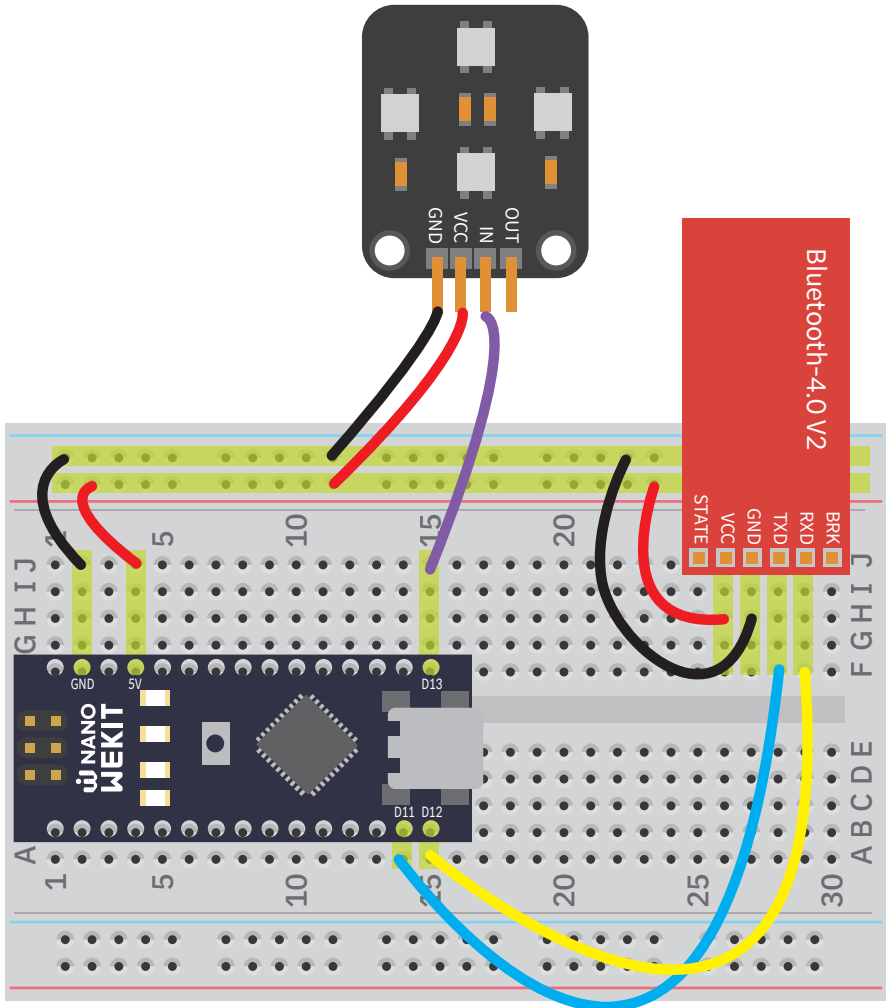
조립 필수 준비물 : 투명 테이프, 풀
무드등 외관(앞면) 원형 구멍에 반투명 종이 부착 시
고정력을 위해 풀 & 테이프를 같이 사용하세요.





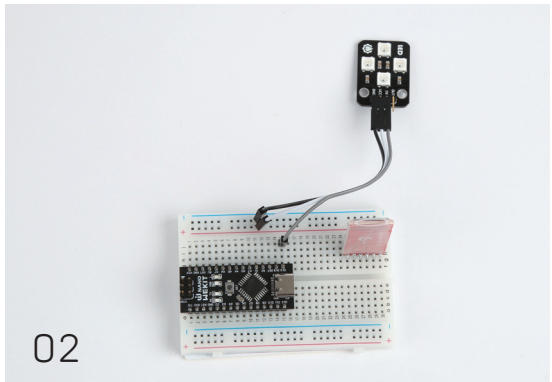
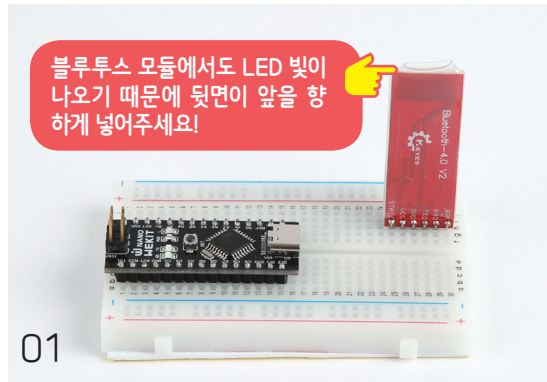
회로도 연결하기

회로도와 배선표를 보며 직접 연결해보아요!

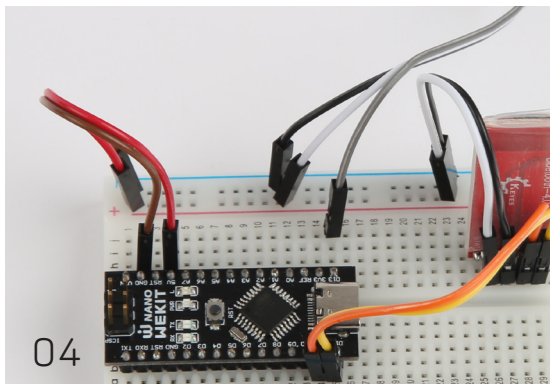
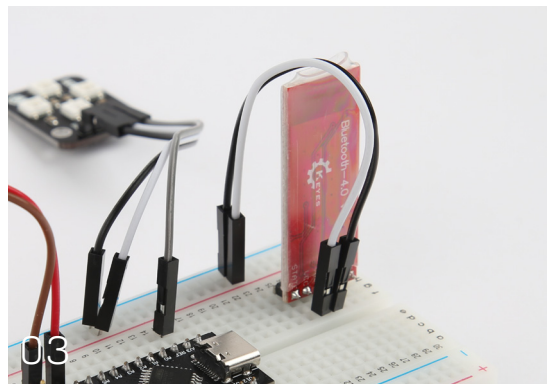


한 눈에 보는 연결 배선표

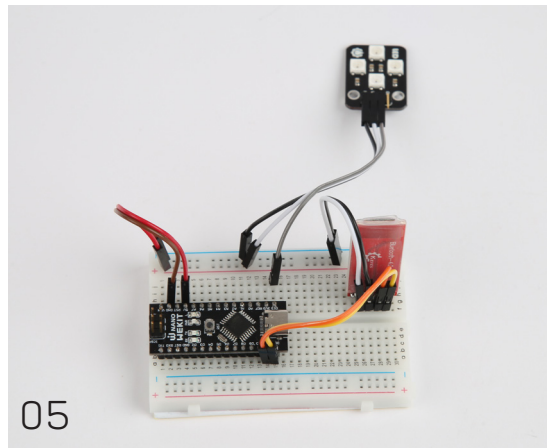
아두이노	네오픽셀 모듈	블루투스 모듈
+5V	VCC	VCC
GND	GND	GND
D13	IN	-
D12	-	RX
D11	-	TX



네오픽셀 모듈의 VCC핀은 브레드보드의 빨간줄에 연결, GND핀은 파란줄에 연결해 주세요. 그리고 네오픽셀 모듈의 IN핀을 아두이노 디지털 핀 13번에 연결해서 데이터를 입력할 준비를 해주세요.



블루투스 모듈의 VCC핀과 GND 핀도 마찬가지로 브레드보드의 빨간줄, 파란줄에 연결해 주세요. 이후 아두이노의 +5V와 GND를 브레드보드의 전원부 빨간줄과 파란줄에 각각 연결해 주면 네오픽셀 모듈과 블루투스 모듈에 전원이 연결돼요.



마지막으로 블루투스 모듈의 RX핀에는 아두이노 디지털 핀 12번을 연결, TX핀에는 아두이노의 디지털 핀 11번을 연결하면 휴대폰과 통신할 준비 완료!



아두이노 설치 및 소스 코드 업로드하기

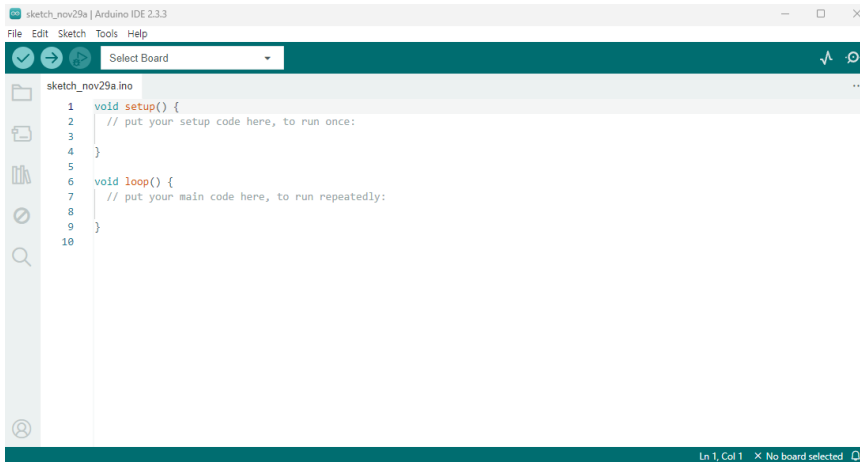
아두이노 IDE 업로드 또는 블록코딩을 통해 코딩해보아요!

회로도대로 연결이 완료됐으면 아두이노로 네오픽셀 LED를 제어할 수 있게 코딩작업을 진행해볼게요. 아두이노 IDE 업로드 방법과 블록코딩 2가지 방법 중 편한 방법을 선택해보세요!



따라해보세요!

1. 아두이노 통합개발환경(IDE)



아두이노 IDE(Integrated Development Environment)는 아두이노 프로그램을 작성하고 컴파일 할 수 있는 통합 개발 환경이에요. 아두이노 공식 홈페이지에서 무료로 제공하고 있어서 편리하게 사용할 수 있어요.

2. 아두이노 통합개발환경(IDE) 설치

아두이노 통합환경(IDE)은 아두이노 공식 홈페이지(<https://www.arduino.cc>)에서 다운로드 할 수 있어요. 아두이노 공식 홈페이지에 접속한 후 **SOFTWARE**를 클릭해 주세요.



Download the Arduino IDE 부분에서 각자 운영체제의 알맞은 버전의 파일을 선택해서 다운로드 하면 돼요. 제가 다운로드할 시점에서 최신 버전은 2.3.3이예요.

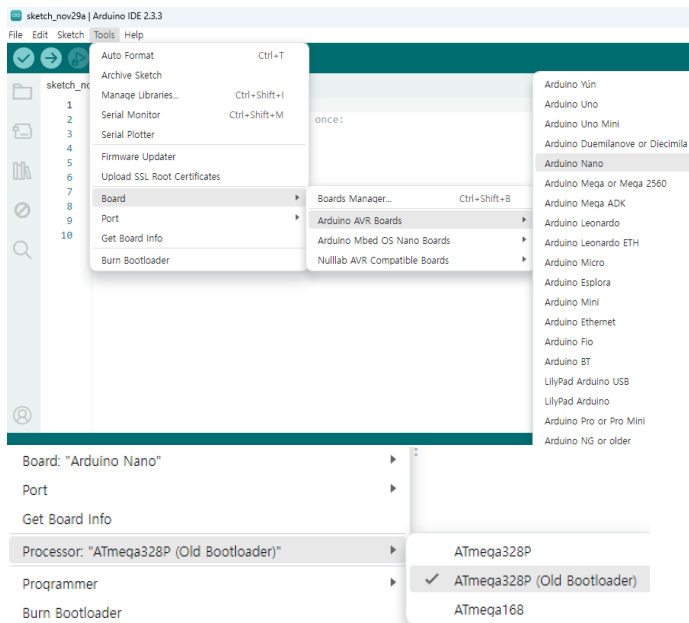
Contribute to the Arduino Software

Consider supporting the Arduino Software by contributing to its development. (US tax payers, please note this contribution is not tax deductible). [Learn more on how your contribution will be used.](#)

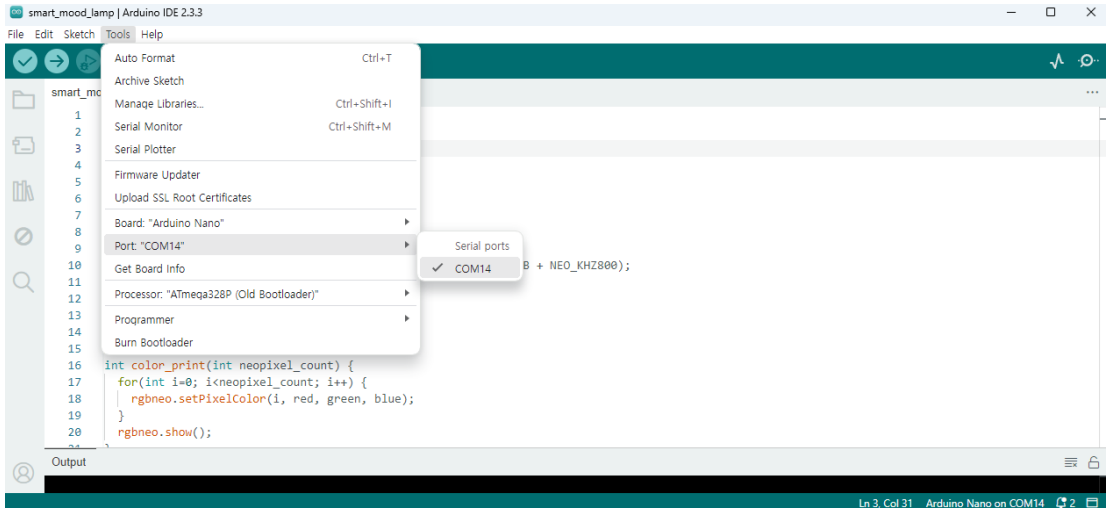
파일을 선택하면 아두이노 소프트웨어에 대한 기부를 권하는 창이 떠요. 기부를 하지 않으려면 JUST DOWNLOAD를 클릭하고 다운로드하면 돼요.

3. 아두이노 통합개발환경(IDE) 기본 세팅

IDE를 사용하는 방법에 대해 알려드릴게요. 아두이노와 PC를 USB로 연결한 후에 진행해야 되고, 이때 IDE 환경에서 아두이노 보드(Board)와 PC에 연결되는 COM포트(Port)를 맞춰줘야 해요.



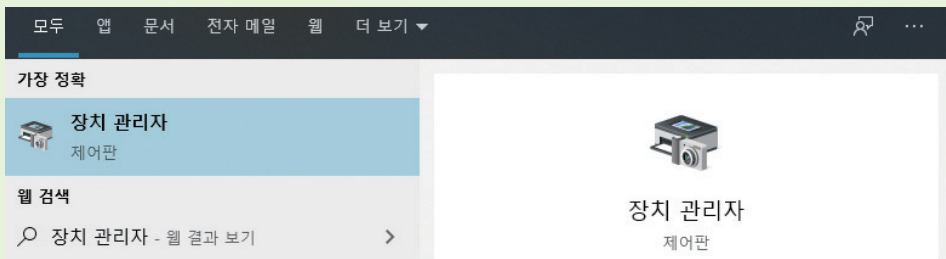
보드를 선택하는 방법은 메뉴바 - 툴 - 보드 - Arduino Nano 보드를 선택해 주면 돼요. 이 때, 프로세서는 ATmega328P (Old Bootloader)를 선택해 줘야 해요.



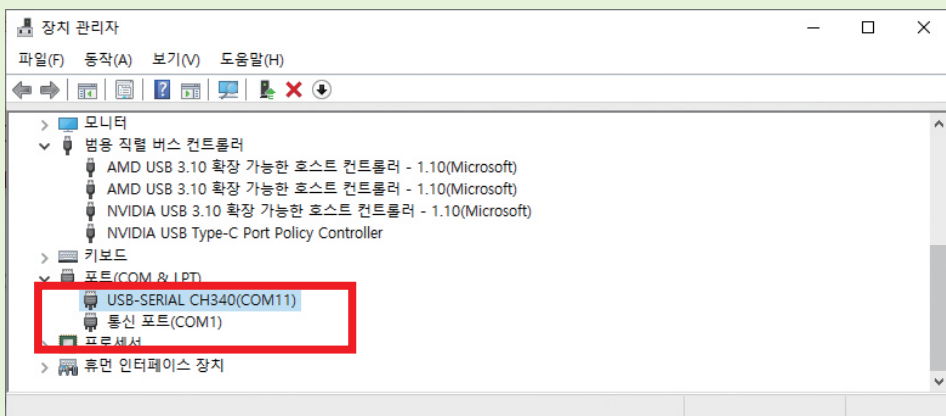
포트 선택하는 방법은 메뉴바 - 툴 - 포트에서 아두이노에 해당하는 COM 포트를 선택해 주면 돼요.

여기서 잠깐! 포트 번호는 어떻게 확인하나요?

포트 번호의 경우 PC환경에 따라 숫자가 달라요. 포트 번호를 찾아보는 방법은 윈도우 검색창에 '장치 관리자'를 검색해서 알아볼 수 있어요.



장치 관리자의 포트(COM & LPT)를 보면 Arduino Uno옆에 컴포트 번호가 있어요.

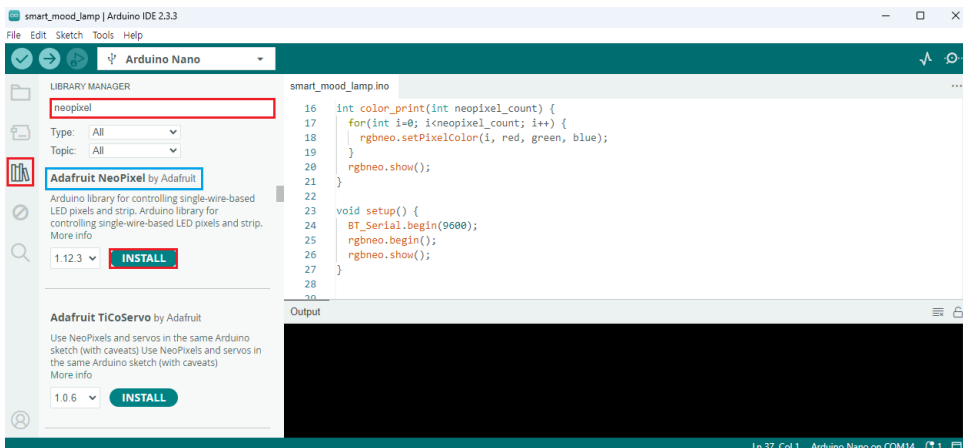




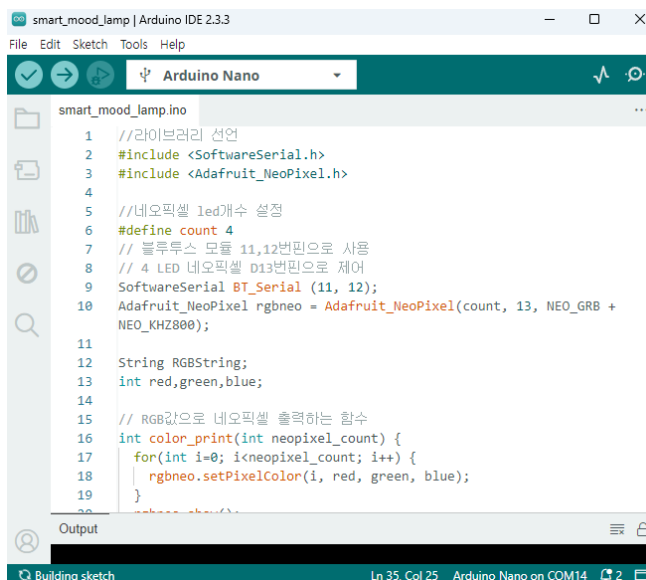
여기서 잠깐! 포트가 잡히지 않아요!

아두이노 ch340 칩 드라이버 설치를 진행해주세요. 설치 방법은 디바이스마트 블로그를 참고해주세요. http://blog.naver.com/no1_devicemart/221907461341

코드를 업로드하기에 앞서 네오픽셀을 사용하기 위해 라이브러리를 다운로드해야해요. 아두이노 IDE 라이브러리 매니저를 눌러 neopixel을 검색해주고, Adafruit NeoPixel 라이브러리를 설치해주면 끝!

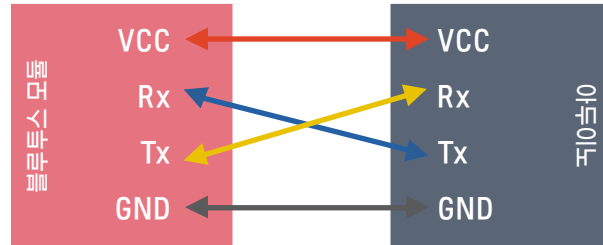


4. 아두이노 소스 코드 업로드



소프트웨어 시리얼(Software Serial) 코드 부분의 아두이노 <-> 블루투스 모듈 통신을 알아봐요.

Rx(Receive)는 데이터 수신을 의미하고, Tx(Transmit)은 데이터 송신을 의미해요. 따라서 통신을 하는 두 제품을 연결할 때 Rx와 Tx를 교차(Cross)해서 연결해야 해요.



블루투스 모듈의 Tx핀이 아두이노의 소프트웨어 시리얼(Software Serial) Rx포트로 지정된 디지털 11번 핀에 연결되고, 블루투스 모듈의 Rx핀은 아두이노의 소프트웨어 시리얼 Tx포트로 지정된 디지털핀 12번에 연결하면 서로 통신핀을 교차하면서 통신할 수 있어요.

아래는 아두이노 IDE 소스 코드예요. 해당 소스를 아두이노 IDE에 입력해주세요.

```
//라이브러리 선언
#include <SoftwareSerial.h>
#include <Adafruit_NeoPixel.h>

//네오피셀 led개수 설정
#define count 4
// 블루투스 모듈 11,12번핀으로 사용
// 4 LED 네오피셀 D13번핀으로 제어
SoftwareSerial BT_Serial (11, 12);
Adafruit_NeoPixel rgbneo = Adafruit_NeoPixel(count, 13, NEO_GRB + NEO_KHZ800
);

String RGBString;
int red,green,blue;

void setup() {
  BT_Serial.begin(9600);
  rgbneo.begin();
  rgbneo.show();
}
```

```

void loop() {
  if(BT_Serial.available()) {
    RGBString = BT_Serial.readStringUntil("\n");
    //Serial.println(RGBString);

    //블루투스 연결이 해제되어도 LED출력하기 위한 코드
    while(RGBString=='OK+LOST') { //color_print함수 출력
      color_print(count);
      if(BT_Serial.available()) {
        RGBString = BT_Serial.readStringUntil('\n');
        if(RGBString=='OK+CONN') {
          break;
        }
      }
    }
    // 블루투스 통신으로 온 데이터값을 RGB값으로 나눠주는 코드
    red = RGBString.substring(0,3).toInt();
    green = RGBString.substring(3,6).toInt();
    blue = RGBString.substring(6,9).toInt();
    //color_print함수 출력
    color_print(count);
  }
}
// RGB값으로 네오픽셀 출력하는 함수
int color_print(int neopixel_count) {
  for(int i=0; i<neopixel_count; i++) {
    rgbneo.setPixelColor(i, red, green, blue);
  }
  rgbneo.show();
}

```



주의사항

PDF에서 소스 코드 내용을 드래그하여 아두이노 IDE에 복사, 붙여넣기 할 경우, ‘, “, \ 기호 인식 문제로 오류가 발생할 수 있으니 활성화 여부를 꼭 확인해보세요. 활성화가 안 될 경우에는 해당 기호들을 지웠다가 직접 입력해보세요.

아두이노 IDE 메뉴바 - 스케치 - 업로드를 누르거나, 왼쪽 상단에 있는 화살표 버튼을 눌러서 업로드할 수 있어요. 업로드가 완료되면 아래 하단에 업로드 완료라고 글씨가 떠요.



```
smart_mood_lamp | Arduino IDE 2.3.3
File Edit Sketch Tools Help
Arduino Nano
smart_mood_lamp.ino
1 //라이브러리 선언
2 #include <SoftwareSerial.h>
3 #include <Adafruit_NeoPixel.h>
4
5 //네오픽셀 led개수 설정
6 #define count 4
7 // 블루투스 모듈 11,12번핀으로 사용
8 // 4 LED 네오픽셀 D13번핀으로 제어
9 SoftwareSerial BT_Serial (11, 12);
10 Adafruit_NeoPixel rgbneo = Adafruit_NeoPixel(count, 13, NEO_GRB + NEO_KHZ800);
11
12 String RGBString;
13 int red,green,blue;
14
15 void setup() {
16   BT_Serial.begin(9600);
17   rgbneo.begin();
18   rgbneo.show();
19 }
20
21 void loop() {
22   if(BT_Serial.available()) {
23     RGBString = BT_Serial.readStringUntil('\n');
```

만약 업로드가 안된다면, 소스 코드에 오타가 없는지 보드 종류와 프로세서, 포트가 틀린 곳이 없는지 다시 확인해 주세요!



블록코딩으로 코딩하기

블록코딩으로 쉽게 명령을 내려요.

스크래치는 미국의 매사추세츠 공과 대학교(MIT)에서 만들어진 프로그램으로 청소년 및 코딩을 처음 배우는 사람들을 대상으로 만들어졌어요. 누구나 무료로 사용할 수 있고, 명령어를 레고 블록을 조립하듯이 쌓아가며 프로그래밍 할 수 있어서 쉽고 재미있게 코딩을 배울 수 있어요.

스크래치를 사용하는 방법에 대해 배워보기에 앞서 프로그램을 설치해야 해요. 일반적으로 웹에서 바로 작동하는 스크래치가 아닌, 아두이노에 사용할 수 있는 스크래치를 사용할 거예요.

우리가 사용할 프로그램은 mBlock이라는 프로그램이에요.

이 프로그램은 스크래치와 아두이노를 연동해서 스크래치로 아두이노에 코딩으로 쉽게 명령을 내릴 수 있게 도와주는 역할이에요.

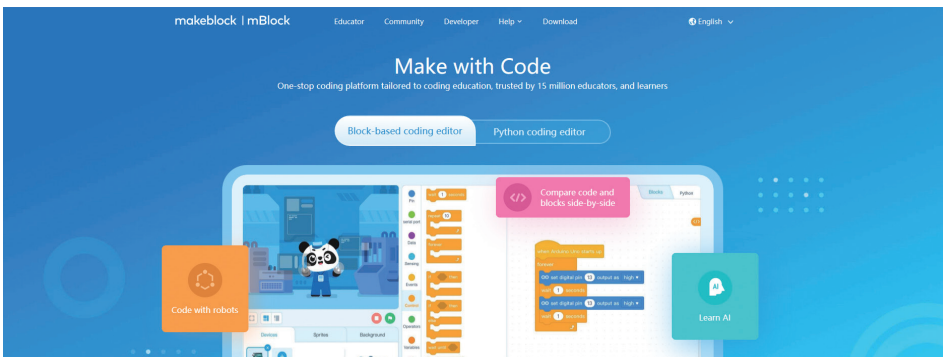
Go!



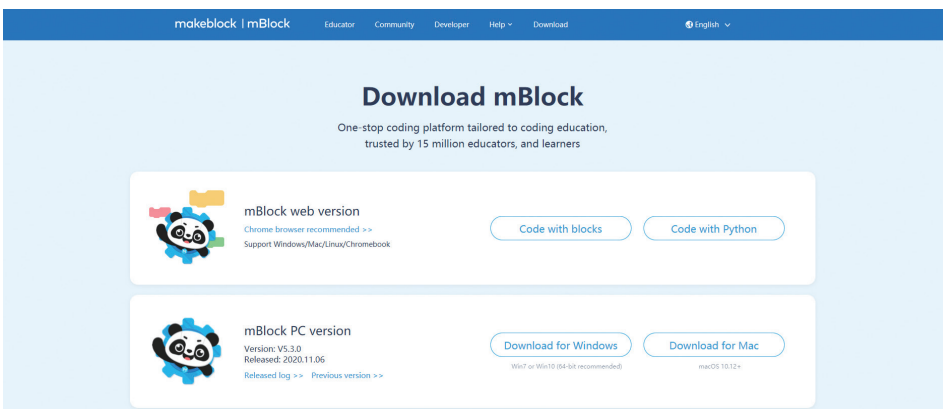
따라해보세요!

1. mBlock 프로그램 설치

인터넷 주소창에 www.mblock.cc 검색하여 mBlock 사이트에 들어가주세요.



mBlock 사이트에서 위쪽에 있는 Download 버튼을 클릭해 주세요.



각자 운영체제의 알맞은 버전의 파일을 선택해서 다운로드하면 돼요. 제가 다운로드할 시점에서 최신 버전은 5.3버전이에요.

2. mBlock 프로그램 기본 세팅

mBlock은 4가지 영역으로 구분할 수 있어요.



① 무대(stage)영역

무대에 팬더가 있고, 여기에서 스프라이트 영역의 스크립트가 재생돼요. 아래에는 장치 영역, 스프라이트 영역, 배경 영역에 연결할 수 있어요. '장치 영역'에는 우리가 사용할 아두이노도 있어요.

②, ③ 카테고리, 블록

가운데 부분을 카테고리(category)와 블록(Block)이라고 불러요. 동작, 모양, 소리 및 펜과 같은 다양한 색상의 카테고리가 있어요. 각각의 카테고리에는 코드 블록이 있으며, 드래그해 오른쪽 작업 영역에 끌어 놓을 수 있어요.

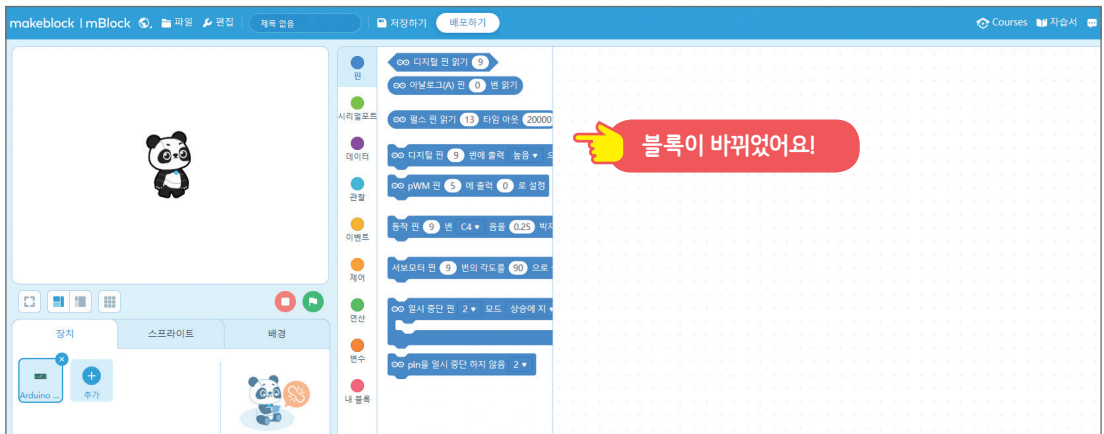
④작업(script) 영역

작업 영역에 블록을 끌어다가 놓음으로써 코드를 배치하는 공간이에요. 이 영역에 있는 모든 것은 무대 혹은 아두이노에 영향을 줄 수 있어요.

블록코딩으로 아두이노를 제어할 수 있도록 mBlock과 아두이노를 연결해 줄 거예요. 무대에 있는 장치 영역을 클릭한 후에 [추가] 버튼을 눌러주세요.



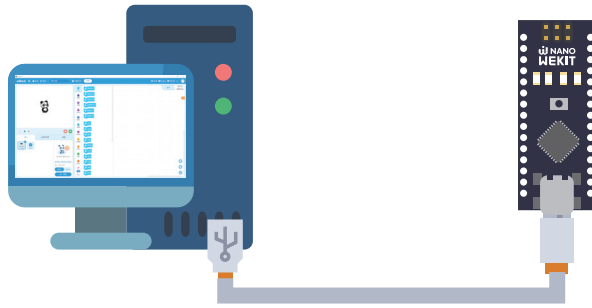
장치 라이브러리에서 아두이노 나노(Arduino Nano(old))를 찾아서 파란색 플러스 버튼을 클릭하여 다운로드 후 다시 선택하고 [확인]을 눌러주세요.



그럼 카테고리에 있는 블록들이 아두이노 나노를 제어할 수 있는 블록으로 바뀌게 보이시죠?

3. 아두이노 연결 방법

블록 코딩을 하기 전 아두이노를 PC와 USB 케이블을 통해 연결해 줘야 해요.



USB A 타입은 PC에 연결하고, USB C타입 쪽은 아두이노에 연결해 주세요.

1번 무대 영역에 **연결** 버튼을 누르면 위 이미지와 같은 창이 뜨고 아두이노에 연결된 포트번호 (COM11)가 뜰 거예요!

다음으로 [접속 가능한 모든 기기 표시] 를 체크해주세요. 이후 **연결** 을 클릭해주면 연결이 돼요.



여기서 잠깐! 인식이 안되요!

인식이 되지 않을 경우, 아두이노에서 케이블을 뺐다가 다시 꽂아주세요!

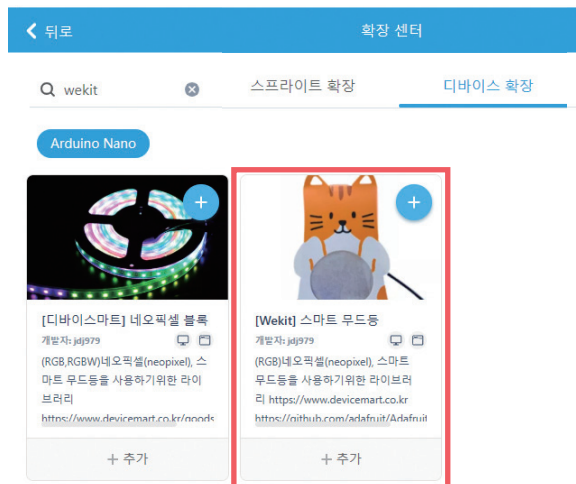
※ 포트 번호는 PC 환경에 따라 번호가 다르며, 해당 번호는 윈도우 검색창에 '장치 관리자'를 검색 후 포트 (COM & LPT) > USB-SERIAL CH340 (포트번호)를 확인해서 해당 번호로 연결해 주세요.

4. mBlock 스마트 무드등 확장자 설치

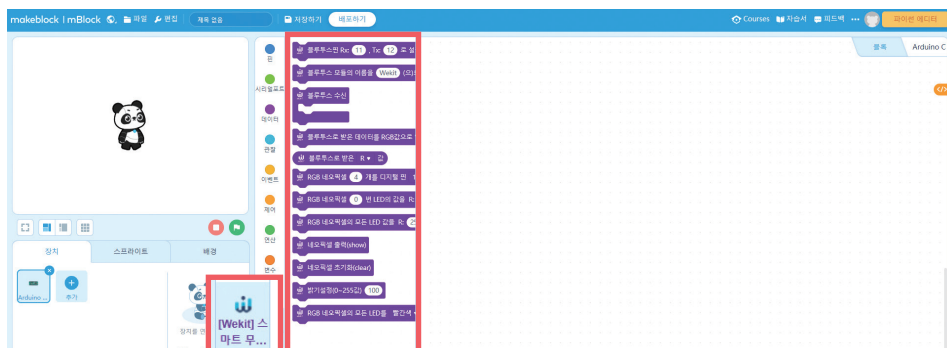
mBlock에서 장치를 아두이노 나노(old)로 선택해 줬다면, 이제 스마트 무드등을 제어하기 위한 블록을 사용할 수 있게 확장자를 설치해야 해요.



먼저 mBlock 프로그램 아래에 있는 확장 버튼을 눌러주세요.



스마트 무드등 혹은 Wekit으로 검색하면 [Wekit] 스마트 무드등 확장자가 있어요. **파란색 플러스 버튼을 클릭하여 다운로드 후 다시 검색하여 [추가]를 눌러주면 이미지와 같이 스마트 무드등 카테고리**가 추가될 거예요.



5. mBlock 블록코딩

mBlock 블록 코딩에서는 휴대폰 APP을 이용해서 스마트 무드등을 제어할 수 있는 블록 코딩 방법과 휴대폰 APP과 연동하지 않고 스마트 무드등 색상을 변경할 수 있는 블록 코딩 방법 이렇게 두 가지 방법을 소개할게요.


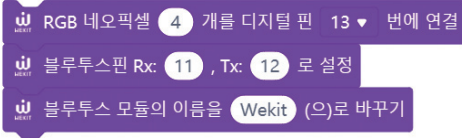
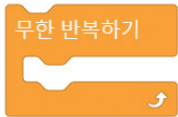


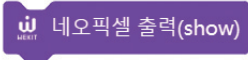
① 휴대폰 APP을 이용한 스마트 무드등 색상변경

스마트 무드등을 제어하기 위한 블록을 먼저 보여줄게요. 아래처럼 만들면 휴대폰을 통해 블루투스로 제어가 가능해요. 블록들을 드래그해 순서대로 끌어와주세요.

The image shows the mBlock software interface with a block-based code for controlling an RGB NeoPixel LED strip via Bluetooth. The code is as follows:

- when Arduino starts up
 - RGB 네오픽셀 4 개를 디지털 핀 13 번에 연결
 - 블루투스핀 Rx: 11, Tx: 12 로 설정
 - 블루투스 모듈의 이름을 wekit (으)로 바꾸기
- 계속 반복하기
 - 블루투스 수신
 - 블루투스로 받은 데이터를 RGB값으로 읽어오기
 - 빨강 을 블루투스로 받은 R 값 로(으)로 설정하기
 - 초록 을 블루투스로 받은 G 값 로(으)로 설정하기
 - 파랑 을 블루투스로 받은 B 값 로(으)로 설정하기
 - RGB 네오픽셀의 모든 LED 값을 R: 빨강 G: 초록 B: 파랑 값으로 설정
 - 블루투스 출력(show)

블록 설명 (순서대로 따라해보세요!)

카테고리	블록	블록설명
이벤트		아두이노가 켜졌을 때, 시작하는 블록이 예요.
스마트 무드등		첫번째와 두번째 블록은 네오픽셀 모듈 과 블루투스 모듈을 아두이노의 디지털 핀에 연결해주는 블록이고, 세번째 블록 은 블루투스 모듈의 이름을 바꿔주는 블 록이에요. 블루투스 모듈의 이름을 바꾸 고 싶지 않다면 마지막 블록은 사용하지 않아도 돼요. (*블루투스 모듈의 기본 이 름 HMSoft)
제어		이 블록 안에 있는 모든 블록을 실행하 고 나면, 다시 무한 반복하기 블록의 처 음으로 돌아가서 계속 반복해요.
스마트 무드등		블루투스 모듈에서 데이터가 들어오는 걸 인식하는 블록이에요.
스마트 무드등 + 변수		블루투스를 통해 들어온 데이터를 각각 R, G, B값으로 받아서 네오픽셀의 값으로 설정해주는 블록이에요. 변수 만드는 방법은 아래를 참고해 주세요.
스마트 무드등		네오픽셀의 설정된 값을 출력하여 빛나게 해주는 블록이에요.

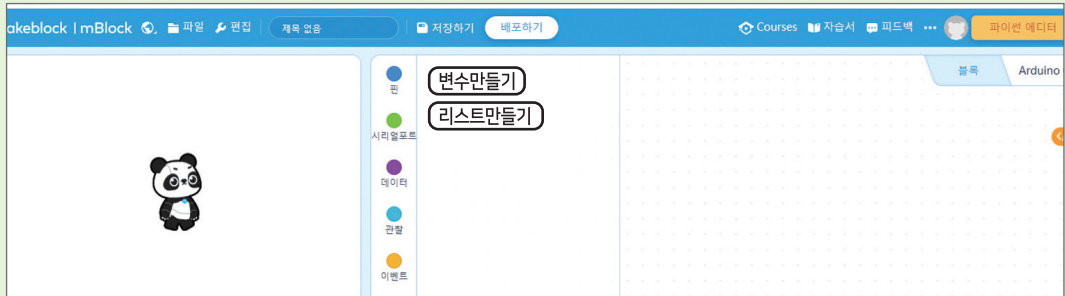


시작 전 확인하세요!

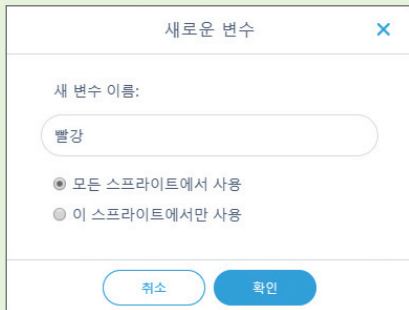
블루투스 모듈의 이름을 바꾸는 블록을 사용해서 블루투스 모듈의 이름을 변경시켰을
경우엔 아두이노의 전원을 껐다가 켜야 이름이 변경돼요.



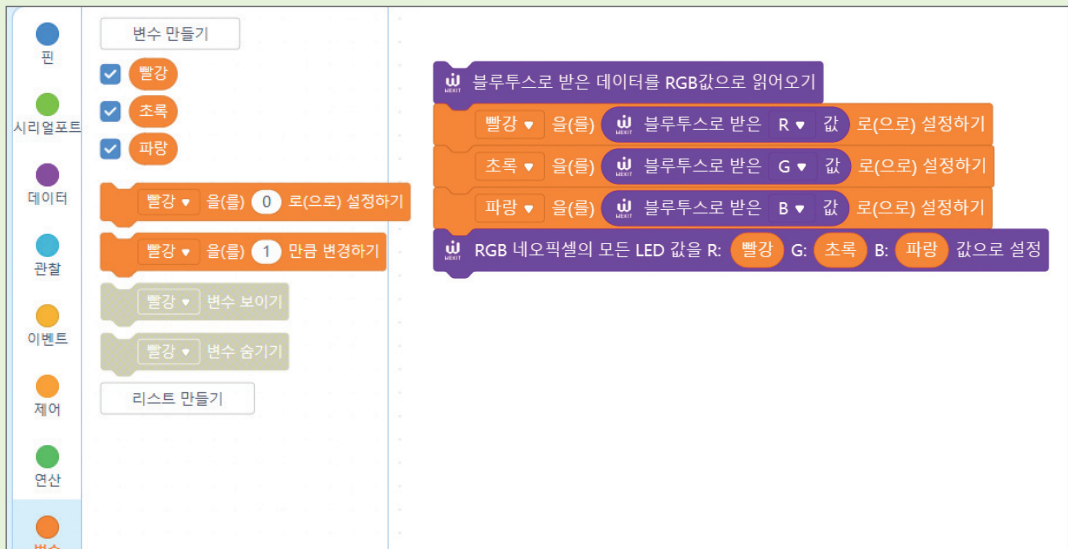
여기서 잠깐! [변수] 카테고리에 있는 블록은 어떻게 사용하나요?



mBlock 프로그램의 변수 카테고리에 들어가서 **변수만들기** 버튼을 클릭해 주세요.
새로운 변수를 만들 수 있는 창이 생기면 새로운 변수 이름에 '빨강'을 입력해 주세요.



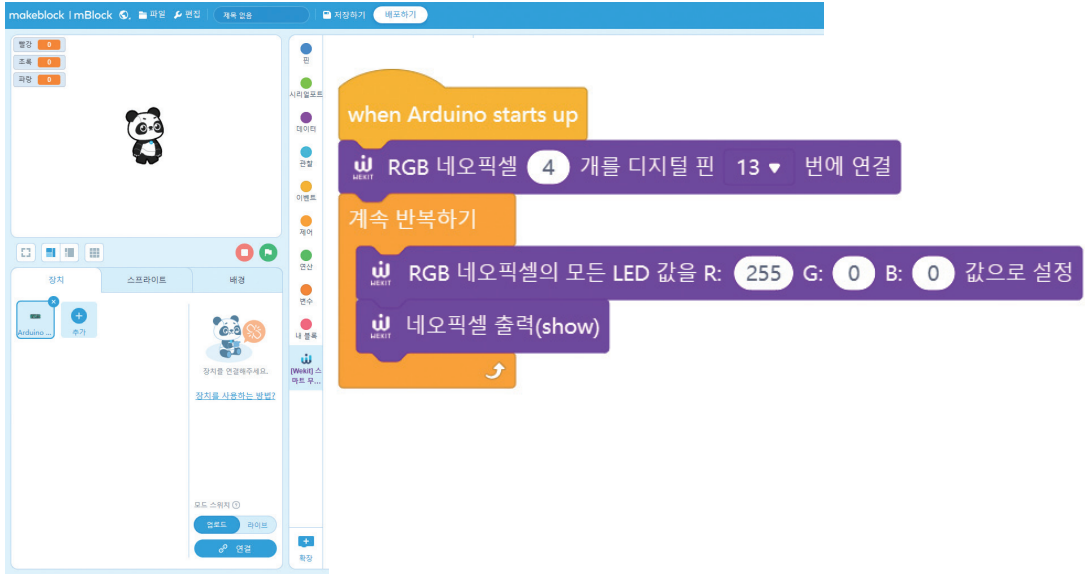
그리고 나면 **빨강** 이라는 변수 블록이 만들어질 거예요, 초록, 파랑도 똑같이 새로운 변수 이름으로 만들어주세요.



만들어진 해당 블록을 작업 영역으로 끌어다 놓은 후, **스마트 무드등** 카테고리에서 **블루투스로 받은 R 값** 블록을 작업영역 변수 블록값 위로 끌어다 놓으면 스마트 무드등+변수 블록이 완성돼요!

② 휴대폰 APP없이 스마트 무드등 색상변경

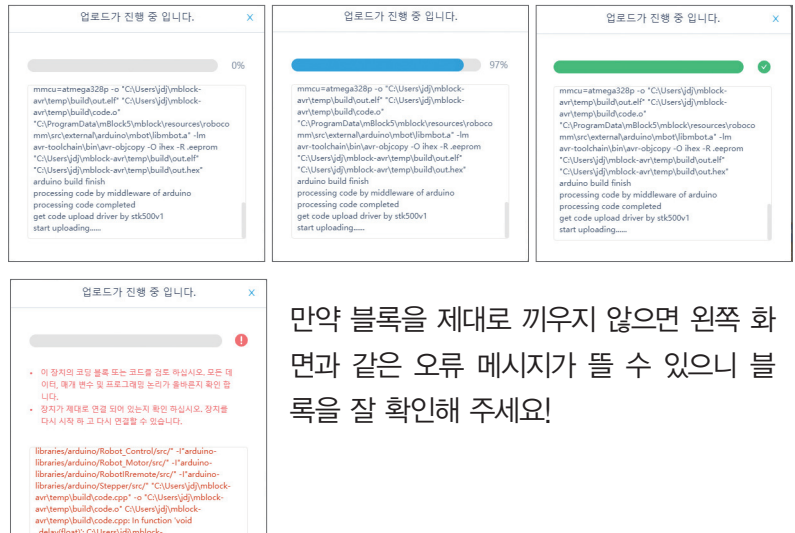
만약 블루투스를 통해 제어하지 않고 아두이노 코딩으로 제어하고 싶다면, 아래에 있는 블록코딩을 사용해도 돼요. 간단하지만 원하는 색상을 네오픽셀로 출력할 수 있는 코딩이에요. LED 값은 0에서 255까지의 값으로 설정이 가능하고, 컴퓨터로 연결된 상태에서만 다른 색상으로 변경이 가능해요. 블록들을 드래그해 순서대로 끌어와주세요.



6. mBlock 블록코딩 업로드



아두이노가 mBlock과 연결된 상태라면 왼쪽 하단에 있는 장치 영역에서 업로드 버튼이 활성화되어 있어요. 버튼을 누르면, 위와 같이 아두이노에 코딩이 업로드돼요.



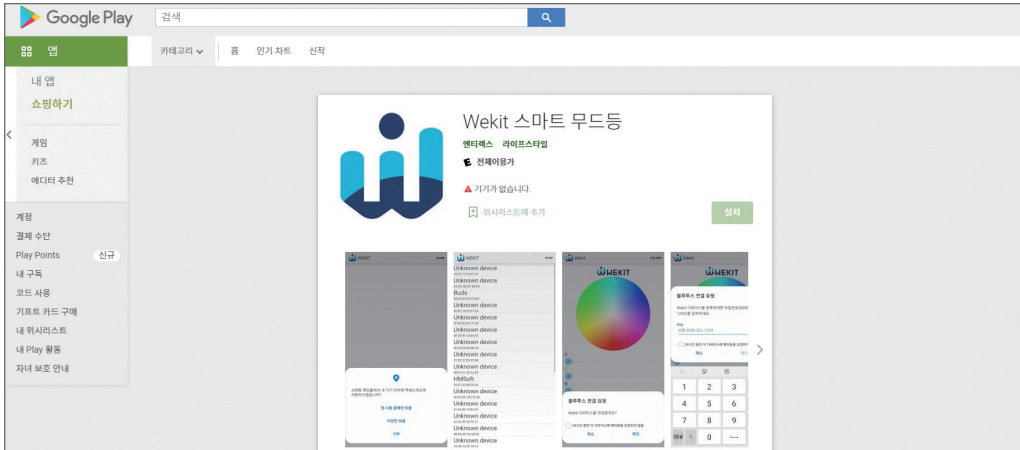
만약 블록을 제대로 끼우지 않으면 왼쪽 화면과 같은 오류 메시지가 뜰 수 있으니 블록을 잘 확인해 주세요!



APP 연결하기

무드등 색상을 바꿔주는 APP을 확인해보아요!



1. 안드로이드폰 버전

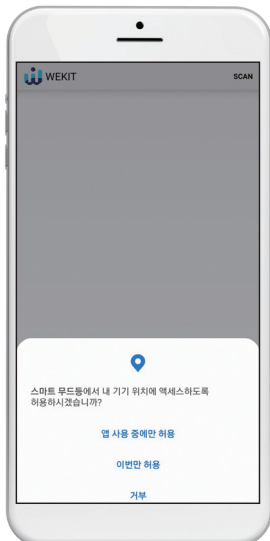


안드로이드 스마트폰에서 아래 링크를 통해 다운로드하거나, <https://play.google.com/store/apps/details?id=com.ledcontrol.wekit1> 구글 플레이 스토어에서 Wekit 스마트 무드등을 검색하시면 해당 앱을 설치할 수 있어요.



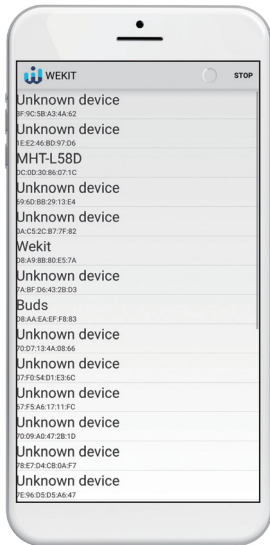
주의사항

앱을 실행하기 전에 안드로이드 폰에서  블루투스와  위치 기능을 켜야 오류 없이 연결할 수 있어요.

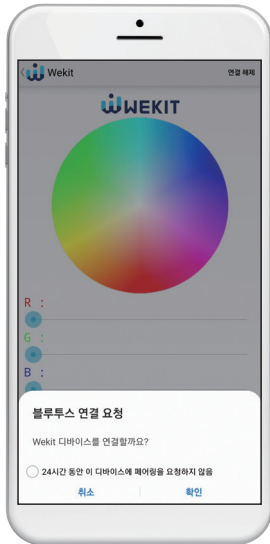


앱 설치가 완료된 후 실행하면 나오는 첫 번째 화면이에요.

기기 위치 권한은 블루투스 위치를 찾기 위함으로 꼭 '앱 사용 중에만 허용'을 클릭해 주세요.



다음 화면은 출력되는 블루투스 리스트 목록이에요.
블록코딩을 통해 블루투스 모듈의 이름을 바꿨다면 'Wekit'을 선택,
블록코딩으로 블루투스 모듈의 이름을 바꾸지 않았다면 기본 이름인
'HMSoft'를 선택해 주세요.



확인 버튼을 눌러 블루투스 연결을 해주고,
다음 화면으로 넘어가 주세요.



블루투스 모듈의 기본 비밀번호는 '000000'이에요.



원형 색상표에서 원하는 색상을 클릭! 하거나 R, G, B 값을 지정해서 자유롭게 색상을 조절해보세요!

※ 원형 색상표는 특정 색상을 터치해야지만 해당 색상이 노출돼요. (드래그로 색상 변경 불가)

밑에 있는 RED, GREEN, BLUE 버튼은 각각 빨강, 초록, 파란색이 출력되도록 나오는 버튼이고 OFF 버튼은 색상을 꺼주는 버튼이에요. 만약 블루투스 통신 리스트가 제대로 출력되지 않을 경우 아래 사항을 검토해 주세요.



1. 위치 기능 및 블루투스 기능 활성화
2. 설정 - 애플리케이션 - 스마트 무드등 - 권한 - 위치 - 앱 사용 중에만 허용으로 변경

2. 아이폰 버전

Wekit에서 지원하는 APP은 안드로이드 폰에서만 사용이 가능해서 아이폰을 사용하시는 분들은 다른 APP을 사용해야 해요.

물론, 이 APP도 다양하게 색상 변경이 가능하며 연결하는 방법은 디바이스마트 블로그 글을 통해 소개하도록 할게요. 아래 링크를 검색하시거나, QR코드를 스캔하시면 바로 확인할 수 있어요.

https://blog.naver.com/no1_devicemart/222256082118

무드등 키트 본체 조립전! 위와 같은 모든 기능들이 앱과 연동하여 작동하는지 확인해 주시고 다음 단계로 넘어가 주세요!

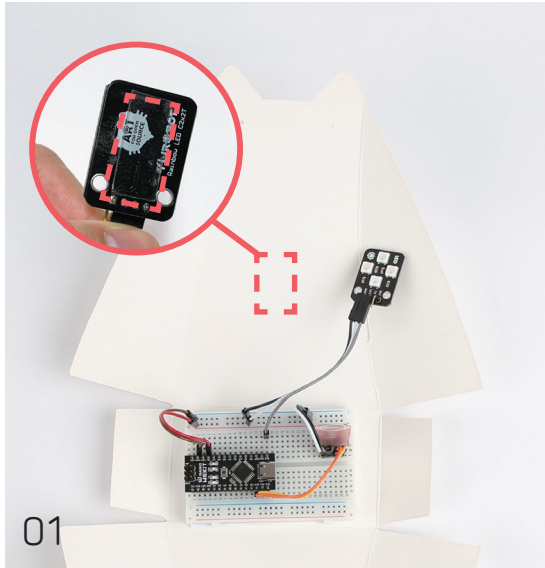


아이폰 버전 확인하러가기

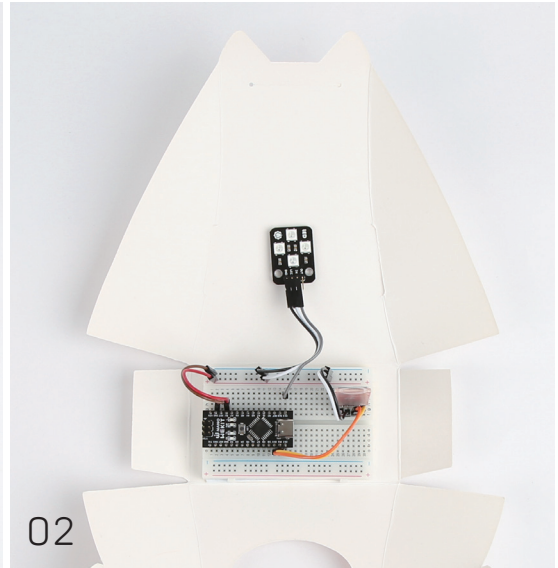


스마트 무드등 완성하기

무드등 외관을 조립해보아요!

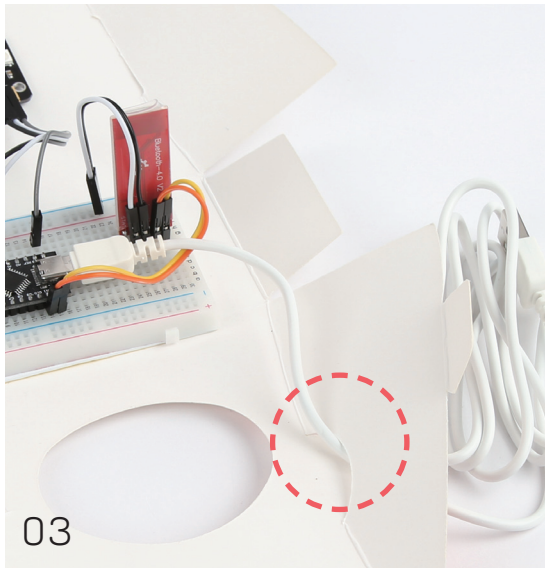


01

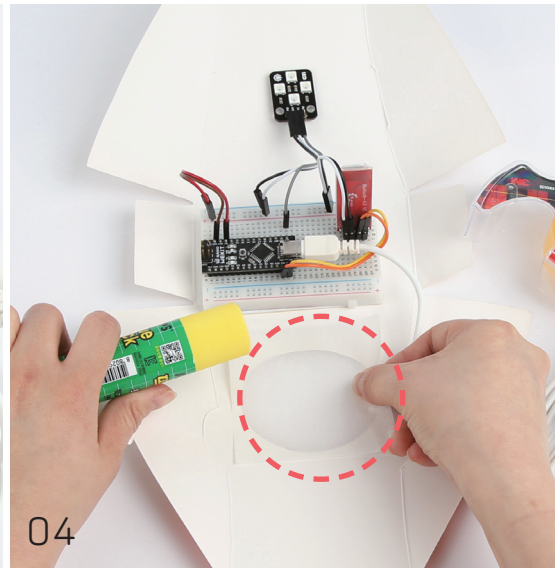


02

회로도에 따라 만든 무드등 본체의 동작 확인이 모두 끝났으면 무드등 외관을 조립해보아요! 무드등 외관을 본체 아래에 사진과 같이 두고, 네오픽셀 모듈 뒷면에 동봉된 양면테이프를 부착 후 첫 번째 사진 속 빨간 라인 안에 붙여주세요.



03



04

다음은 아두이노 나노에 케이블을 연결한 후 외관 앞면 고양이 앞발 쪽으로 케이블 선을 빼주세요. 은은한 빛 번짐을 위해 반투명지를 풀로 원형 구멍 위에 부착해 주세요.

※ 풀로 부착 후 투명 테이프로 한 번 더 고정시키면 단단히 유지가 돼요!



05



06

무드등 완성 후 외관의 틈이 벌어지는 것을 방지하기 위해 외관 뒷면에 양면테이프를 좌, 우에 부착하고, 외관 앞면 머리 부분을 뒷면 홈에 사진처럼 넣어주세요.



07



양쪽 옆면을 접어서 외관 뒷면 홈에 끼우면서 기존에 좌, 우에 부착한 양면테이프로 잘 고정하면 완성!



The background of the image is a teal color with a white circuit board pattern. The pattern consists of numerous thin white lines that form a complex network of paths, resembling a printed circuit board (PCB) layout. These lines are interconnected at various points, creating a dense and intricate web of connections across the entire frame.

Let's play **WEKIT**