



SCRATCH



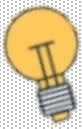


Module **5** : 거리센서

HUMMINGBIRD 로터리센서, 빛 센서, 거리센서

LED와 모터들은 출력장치입니다. 스크래치의 스크립트는 출력 장치들이 작동을 하도록 명령을 내립니다.

반면에, **센서들은 입력장치입니다.** 센서들의 정보가 스크래치로 전달이 됩니다. 스크래치는 정보를 이용해 판단을 내리고 LED와 모터를 제어합니다.



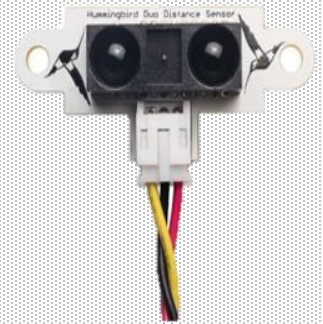
로터리센서



빛 센서



거리센서



허밍버드 키트의 센서들은 모두 같은 방식으로 작동합니다.

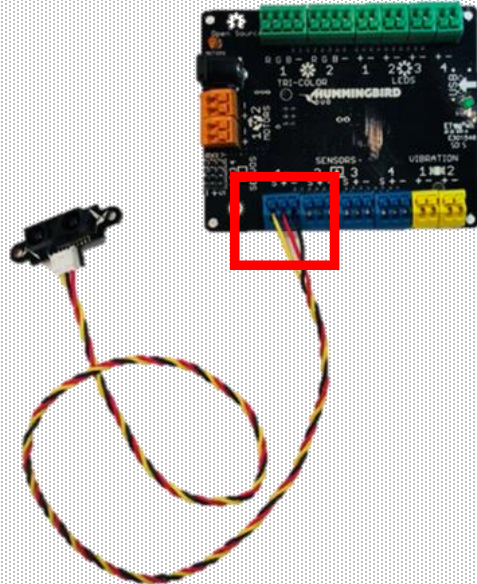
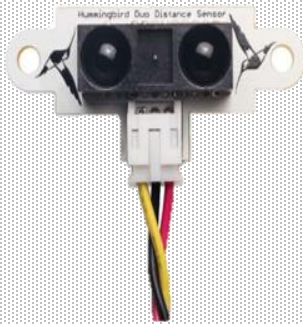
“**SENSORS**”부분의 1~4 포트에 연결합니다.

센서의 빨간색 선은 ‘+’터미널, 검은색 선은 ‘-’터미널,
노란색 선은 ‘S’터미널에 연결합니다.

거리센서



거리센서



HB 거리 센서 포트 1

거리센서는 가장 가까운 물체까지의 거리를 측정합니다.

허밍버드 컨트롤러에 거리센서를 연결해주세요.

허밍버드 거리센서블록을 스크립트로 드래그해 오세요. 드래그해 오셨나요?

그럼, 허밍버드 컨트롤러에 연결된 거리센서의 포트번호와 거리센서 블록의 포트번호가 맞는지 확인해 주세요.

거리센서는 물체까지의 거리를 cm 단위로 측정합니다.

거리센서의 값은 약 8cm 에서 100cm 사이입니다. 거리센서에 물체가 매우 가깝거나 너무 멀리 떨어진 경우에는 거리를 측정할 수 없습니다.

거리센서는 **20cm ~60cm의 범위에서 가장 잘 작동합니다.**

블록을 클릭하면 거리센서의 현재 값을 알 수 있습니다.
현재, 거리센서는 포트1번에 연결되어 있습니다.

스크래치 x 허밍버드 키트

거리센서의 값의 변화를 보려면 변수를 만드는 것이 좋은 방법입니다.
변수이름은 변화하는 값을 대표하는 이름을 의미합니다.
변수이름을 '거리' 로 정해봅시다.
'거리'변수는 거리센서의 변화는 값을 저장하고 있으며,
무대에서 그 값을 읽을 수 있습니다.



변수 만들기

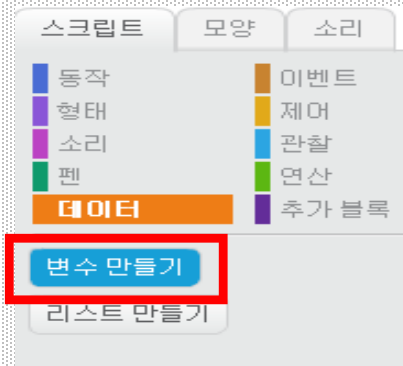
1

먼저, 데이터 메뉴를
눌러주세요.



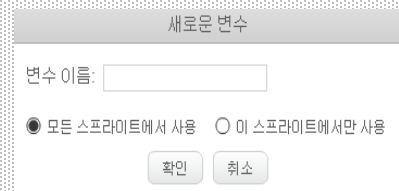
2

다음에 변수 만들기를
클릭해 줍니다.



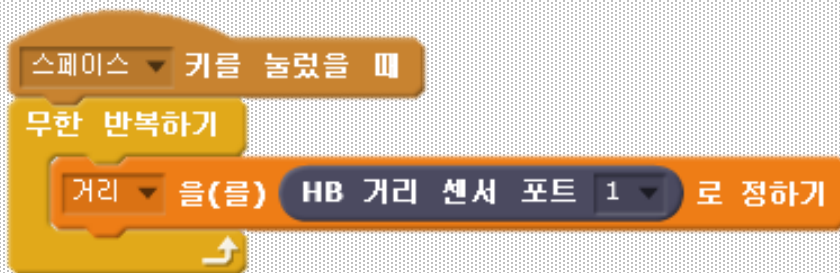
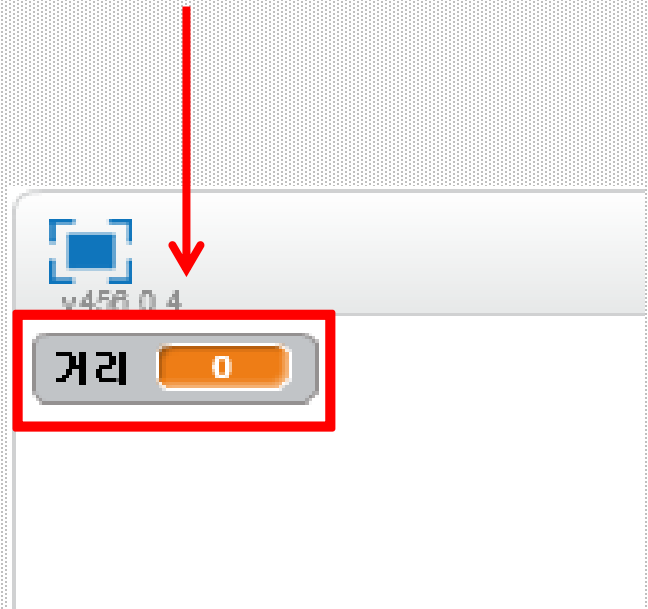
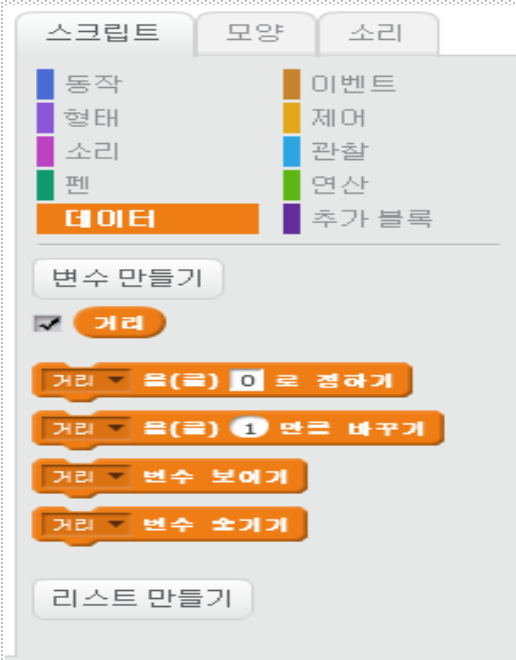
3

새로운 변수 창이 나타납니다.
변수 이름을 '거리' 로 지정하고 확
인을 눌러주세요.



스크래치 x 허밍버드 키트

변수를 생성하면 블록이 생성되는 것을 확인 할 수 있습니다. 여기에는 변수의 값을 변화를 확인 할 수 있는 블록이 포함되어 있습니다. 또한 변수의 값은 무대 왼쪽 상단 모서리에 표시 됩니다.



‘거리’ 변수의 값을 거리센서의 값과 같게 설정하려고 합니다. 이렇게 하기 위해서 우리는 정하기 블록을 사용 할 수 있습니다.

정하기 블록 안에 거리센서의 블록을 넣어줍니다.

이 블록을 무한반복 블록으로 감싸주면 ‘거리’ 변수에 거리센서의 변화는 값을 지속적으로 확인 할 수 있습니다.

스크래치 x 허밍버드 키트

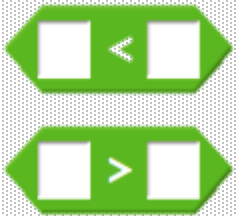
Exercise 5.1

앞의 블록들을 실행시키고 센서 앞에 물체를 앞뒤로 움직여보며, 값의 변화를 살펴봅시다.

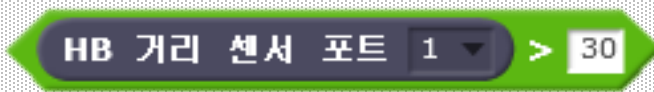
스크래치 x 허밍버드 키트



거리센서를 사용하여 로봇이 결정하게 할 수 있습니다.
스크래치의 **만약 - (이)라면 아니면 블록**은 판단 블록입니다.
이 블록은 **제어 메뉴**에서 찾을 수 있습니다. 눈치 채셨나요?

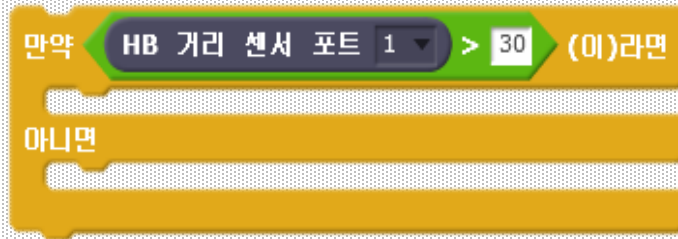


스크래치의 **만약 - (이)라면 블록**과 **만약 - (이)라면 아니면 블록**에는 다른 스크래치 블록을 사용하는 공백이 있습니다. 이 공백은 논리 블록이 필요합니다. **논리 블록**은 '참' 과 '거짓'을 판단 할 수 있습니다.



허밍버드 **거리센서 블록**과 **연산 메뉴**의 논리 블록을 이용하여 논리블록을 만들 수 있습니다. 이 블록은 **만약 ~ (이)라면 아니면 블록**의 공백과 모양이 같습니다.

스크래치 x 허밍버드 키트



이 블록은 거리센서의 값이 30보다 큰 경우에 '참' 이고 그렇지 않은 경우에 '거짓' 입니다.

'30' 이라는 값은 논리 블록이 결정을 내리는데 사용하는 임계값 이라고 합니다.

위의 블록에서 거리센서의 임계값은 30cm 입니다. 거리센서 앞에 물체를 두고 논리 블록을 클릭해 보세요. '참' 인가요? '거짓' 인가요?

만약 과 아니면 블록 사이에
삼색 LED 블록을 추가해 결과를 확인해 봅시다.

스크래치 x 허밍버드 키트



예를 들어 위 프로그램을 확인해보세요. 스페이스 바를 눌렀을 때, 물체와 센서까지의 거리가 30cm이상 이면 ‘참’의 값으로 삼색 LED가 빨간색으로 켜고, ‘거짓’의 값이면 아래 부분에 있는 블록인 삼색LED의 초록색의 빛으로 바뀝니다. 색의 변화를 확인해 보세요!

위의 블록들은 스페이스 바를 누를 때마다 결정을 내립니다. 이 결정을 반복 할 수도 있습니다. 어떤 반복 블록을 사용하면 지속적으로 센서의 값에 따라 삼색 LED의 색이 변할까요?

Exercise 5.2

어떤 물체가 거리센서에 가까워질 때 단색 LED가 깜빡거리며 거리에 따라 서보 모터의 각도가 변하도록 프로그래밍 해봅시다.

단, 거리 센서 근처에 아무런 물체도 존재하지 않으면 단색 LED가 꺼지고, 서보 모터의 움직임도 멈춰야 합니다.