

2020년도 로봇창의교육사업 창의교실



치즈스틱으로 배우는 로봇 프로그래밍

총괄

한국로봇산업진흥원 글로벌성장팀 김규환 팀장, 동효경 책임

도움을 주신 분들

- **대구컴퓨팅교사연구회**
 - 김진수 교사(대구유가초등학교)
 - 이정서 교사(대구화남초등학교)
 - 장준혁 교사(대구서동초등학교)
 - 전용욱 교사(대구한실초등학교)
 - 정현재 교사(대구다사초등학교)
- (주)로보메이션

일 러 두 기

1. 본 교재는 지능형 로봇보급 및 확산사업 內 ‘로봇창의교육사업’의 결과물입니다.
2. 본 교재는 2020년 로봇창의교육사업 창의교실 교육용 로봇 업체로 선정된 (주)로보메이션의 교구인 ‘치즈스틱 스타터키트’를 활용한 수업 운영을 위하여 만든 교재입니다.
3. 본 교재는 창의교실 운영학교 및 ‘치즈스틱 스타터키트’ 활용 교재가 필요한 학교를 대상으로 만든 교재이며, 활용 시 원본데이터의 출처를 밝혀주시기 바랍니다.
4. 본 교재는 한국로봇산업진흥원이 기획하고, 대구컴퓨팅교사연구회 및 (주)로보메이션의 연구로 만들어졌습니다.
5. 본 교재와 관련된 문의는 한국로봇산업진흥원 글로벌성장팀 (053-210-9673)으로 연락주시기 바랍니다.

CONTENTS

- | | |
|---|-------------------------------------|
|  치즈 스틱 | 1. 피아노 연주하기 1 |
|  치즈 스틱 | 2. O,X 퀴즈 버튼 만들기 11 |
|  치즈 스틱 | 3. 나만의 애니메이션 만들기 22 |
|  치즈 스틱 | 4. 엔트리봇 조종기 만들기 34 |
|  치즈 스틱 | 5. 방향조절 바람개비 만들기 45 |
|  치즈 스틱 | 6. 감정표현 프로그램 만들기 55 |
|  치즈 스틱 | 7. 공 튀기기 게임 만들기 65 |
|  치즈 스틱 | 8. 밝기 조절 가로등 만들기 75 |
|  치즈 스틱 | 9. 춤추는 종이컵 인형 만들기 85 |
|  햄스터 | 10. 햄스터 로봇 알아보기 93 |
|  치즈 스틱 햄스터 | 11. 햄스터 조종기 만들기 102 |
|  치즈 스틱 햄스터 | 12. 지뢰탐지 로봇 만들기 110 |
|  부록 | 인형 꾸미기(9. 춤추는 종이컵 인형 만들기) 119 |

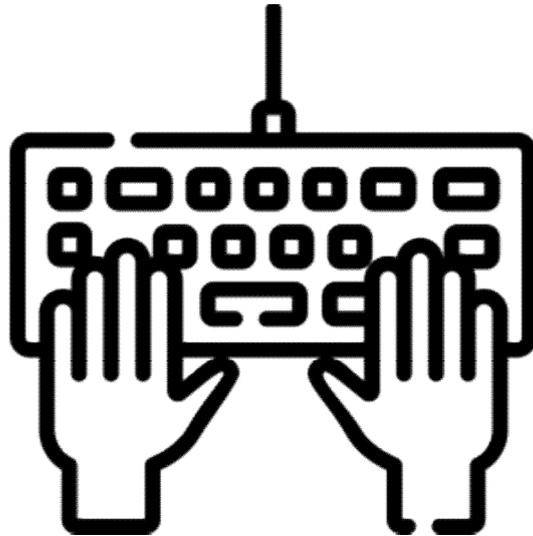
Basic Programing

1 피아노 연주하기

키보드의 자판을 사용하여 치즈스틱으로 피아노 연주를 할 수 있어요.



키보드의 자판을 사용하여 치즈스틱으로 피아노 연주를 할 수 있어요.
치즈스틱의 [소리 블록]에 대하여 알아보시다.



준비를 해봅시다.



컴퓨터 또는 노트북



USB 동글



치즈스틱 본체



사용할 블록을 알아보시다.

 시작하기 버튼을 클릭했을 때

시작하기 버튼을 클릭하면
아래에 연결된 블록들을
실행합니다.

 q 키를 눌렀을 때

지정된 키를 누르면 아래에
연결된 블록들을 실행합니다.

치즈 스틱 0 : 도 4 음을 0.5 박자 연주하기 

지정된 소리를
입력한 박자만큼만
연주합니다.

치즈 스틱 0 : 0.25 박자 쉬기 

입력한 박자만큼 쉽니다.

치즈 스틱 0 : 연주 속도를 20 BPM만큼 바꾸기 

지정한 속도만큼 연주 속도를
바꾸어 줍니다.

치즈 스틱 0 : 연주 속도를 60 BPM으로 정하기 

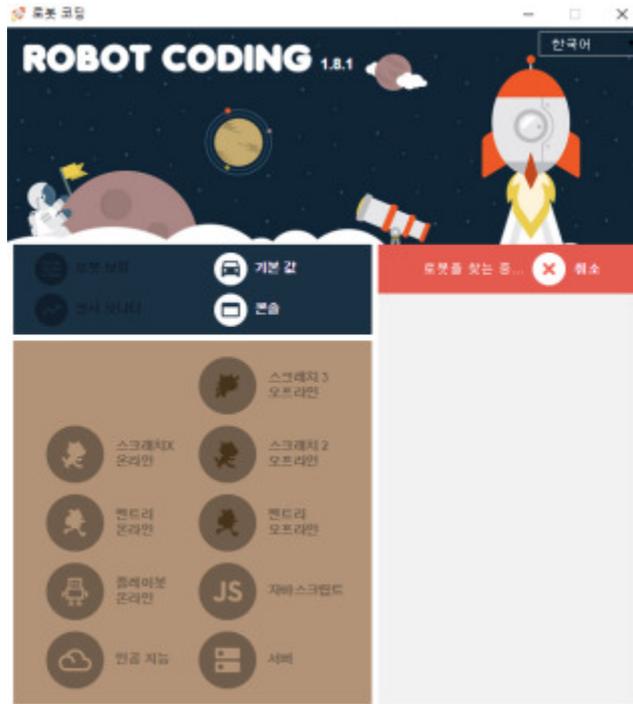
지정한 속도로 연주 속도를
정합니다.



치즈스틱 프로그램을 설치해 봅시다.



컴퓨터에 '로봇 코딩 소프트웨어'를 설치하고 프로그램을 실행합니다.



'로봇 코딩 소프트웨어'는 <http://robomation.net> 또는 <http://hamster.school/ko/> 사이트에서 다운받을 수 있습니다.



햄스터 로봇



터틀 로봇



'로봇 코딩 소프트웨어'를 사용하면 치즈스틱 뿐만 아니라 햄스터 로봇, 터틀 로봇을 제어할 수 있습니다.



치즈스틱을 컴퓨터와 연결해 봅시다.



컴퓨터에 'USB 동글'을 연결합니다. 그리고 치즈스틱을 USB 동글 가까이(약 20cm 이내) 가져가서 전원을 ON으로 켵니다.



치즈스틱에서 뽁! 소리가 나고 치즈스틱과 USB 동글의 블루투스 표시등이 파란색으로 계속 켜져 있거나 빠르게 깜빡이면 바르게 연결되었습니다.



■ 블루투스 연결 표시등 (*)

표시등 동작	표시등의 의미
파란색으로 천천히 깜박임	블루투스 연결을 기다리는 중
파란색으로 계속 켜져 있음	블루투스가 연결된 상태
파란색으로 빠르게 깜박임	데이터를 받고 있는 상태



USB 동글은 가장 최근에 페어링 된 치즈스틱 하나만 기억합니다. 따라서 페어링은 한 번만 하면 되고, 그 이후부터는 치즈스틱을 USB 동글 가까이 가져가서 페어링 할 필요는 없습니다.



로봇코딩으로 엔트리를 실행하여 프로그래밍을 해봅시다.



치즈스틱이 컴퓨터에 바르게 연결되었으면 ‘엔트리 오프라인’을 선택합니다.



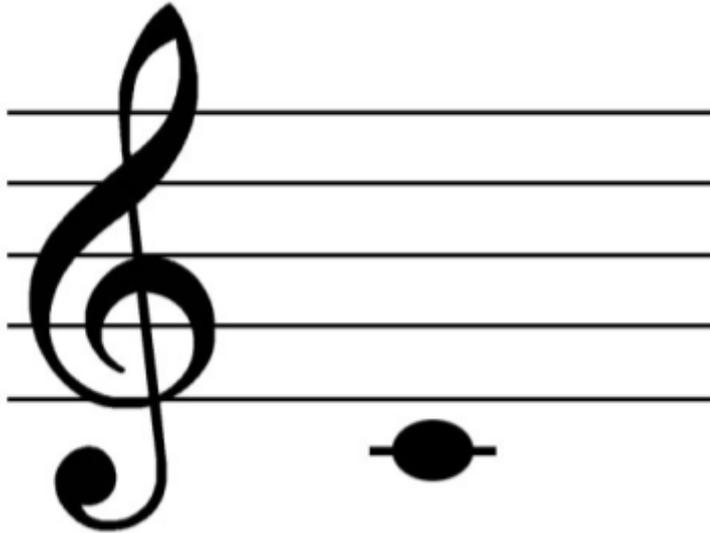
단일 로봇을 선택한 후에, ‘새 파일’을 선택합니다.



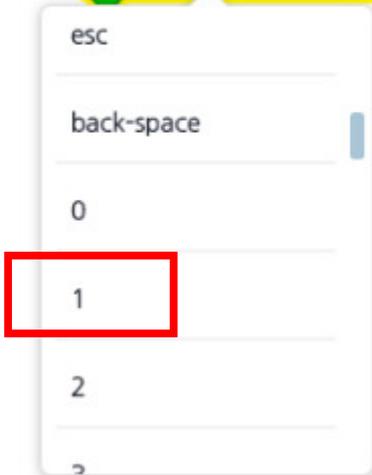
현재 치즈스틱은 엔트리 v2.0.17과 호환되며 계속 업데이트 될 예정입니다.



키보드 '숫자 1'을 선택했을 때, 치즈스틱에서 게임음 '도' 소리가 나도록 해봅시다.



카테고리에서 [q키를 눌렀을 때] 블록을 가져옵니다.

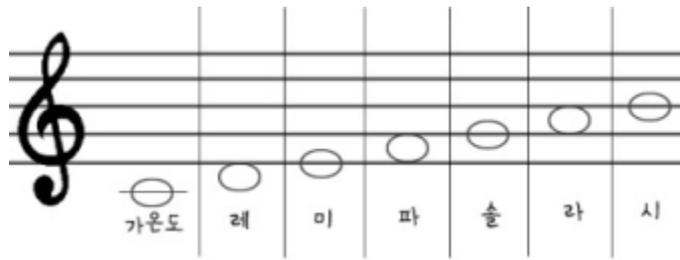


'q'를 선택하여 '숫자 1'로 변경합니다.



카테고리에서 [도 4음을 0.5 박자 연주하기] 블록을 연결합니다.

[연습과제 1] 키보드의 숫자 키 '1~7'으로 '도레미파솔라시' 연주하기



[연습과제 2] '높은 도' 소리가 나도록 프로그래밍하기



확인하기

[연습과제 1]

1▼ 키를 눌렀을 때
치즈 스틱 0 : 도▼ 4▼ 음을 0.5 박자 연주하기

5▼ 키를 눌렀을 때
치즈 스틱 0 : 솔▼ 4▼ 음을 0.5 박자 연주하기

2▼ 키를 눌렀을 때
치즈 스틱 0 : 레▼ 4▼ 음을 0.5 박자 연주하기

6▼ 키를 눌렀을 때
치즈 스틱 0 : 라▼ 4▼ 음을 0.5 박자 연주하기

3▼ 키를 눌렀을 때
치즈 스틱 0 : 미▼ 4▼ 음을 0.5 박자 연주하기

7▼ 키를 눌렀을 때
치즈 스틱 0 : 시▼ 4▼ 음을 0.5 박자 연주하기

4▼ 키를 눌렀을 때
치즈 스틱 0 : 파▼ 4▼ 음을 0.5 박자 연주하기

[연습과제 2]

8▼ 키를 눌렀을 때
치즈 스틱 0 : 도▼ 5▼ 음을 0.5 박자 연주하기



프로그램을 실행했을 때 게이름이 자동으로 연주되도록 해봅시다.

시작하기 버튼을 클릭했을 때

치즈 스틱 0 : 도▼ 4▼ 음을 0.5 박자 연주하기

치즈 스틱 0 : 레▼ 4▼ 음을 0.5 박자 연주하기

치즈 스틱 0 : 미▼ 4▼ 음을 0.5 박자 연주하기

치즈 스틱 0 : 파▼ 4▼ 음을 0.5 박자 연주하기

치즈 스틱 0 : 솔▼ 4▼ 음을 0.5 박자 연주하기

치즈 스틱 0 : 라▼ 4▼ 음을 0.5 박자 연주하기

치즈 스틱 0 : 시▼ 4▼ 음을 0.5 박자 연주하기

치즈 스틱 0 : 도▼ 5▼ 음을 0.5 박자 연주하기



자주 사용하는 음표와 쉼표의 길이를 박자로 나타내면 다음과 같습니다.

구분	음표 또는 쉼표	박자
	8분 음표	0.5박자
	4분 음표	1박자
	점4분 음표	1.5박자
	2분 음표	2박자
	8분 쉼표	0.5박자
	4분 쉼표	1박자



학습문제로 공부한 내용을 정리해 봅시다.

작은 별

윤석중요곡
모짜르트곡

반 짝 반 짝 작 은 별 아 름 답 게
비 치 네 동 쪽 하 늘 에 서 도
서 쪽 하 늘 에 서 도 반 짝 반 짝
작 은 별 아 름 답 게 비 치 네

[문제1] 아래와 같이 명령어를 작성하여 키보드로 직접 연주해 봅시다.

1▼ 키를 눌렀을 때
치즈 스틱 0 : 도▼ 4▼ 음을 0.5 박자 연주하기

5▼ 키를 눌렀을 때
치즈 스틱 0 : 솔▼ 4▼ 음을 0.5 박자 연주하기

2▼ 키를 눌렀을 때
치즈 스틱 0 : 레▼ 4▼ 음을 0.5 박자 연주하기

6▼ 키를 눌렀을 때
치즈 스틱 0 : 라▼ 4▼ 음을 0.5 박자 연주하기

3▼ 키를 눌렀을 때
치즈 스틱 0 : 미▼ 4▼ 음을 0.5 박자 연주하기

7▼ 키를 눌렀을 때
치즈 스틱 0 : 시▼ 4▼ 음을 0.5 박자 연주하기

4▼ 키를 눌렀을 때
치즈 스틱 0 : 파▼ 4▼ 음을 0.5 박자 연주하기

8▼ 키를 눌렀을 때
치즈 스틱 0 : 도▼ 5▼ 음을 0.5 박자 연주하기



학습문제로 공부한 내용을 정리해 봅시다.

똑같아요

조금 빠르게

윤석홍
외국곡

무엇이 무엇이 똑같은가
무엇이 무엇이 똑같은가

젓가락 두 짝이 똑같아요
젓가락 네 짝이 똑같아요

[문제2] 아래와 같이 명령어를 작성하여 자동으로 연주가 되도록 해봅시다.

▶ 시작하기 버튼을 클릭했을 때

- 치즈 스틱 0 : 연주 속도를 120 BPM으로 정하기
- 치즈 스틱 0 : 도 4 음을 1 박자 연주하기
- 치즈 스틱 0 : 미 4 음을 1 박자 연주하기
- 치즈 스틱 0 : 솔 4 음을 1 박자 연주하기
- 치즈 스틱 0 : 도 4 음을 1 박자 연주하기
- 치즈 스틱 0 : 미 4 음을 1 박자 연주하기
- 치즈 스틱 0 : 솔 4 음을 1 박자 연주하기
- 치즈 스틱 0 : 라 4 음을 1 박자 연주하기
- 치즈 스틱 0 : 라 4 음을 1 박자 연주하기
- 치즈 스틱 0 : 솔 4 음을 2 박자 연주하기

[반복하기] 블록을 활용할 수도 있어요.

▶ 시작하기 버튼을 클릭했을 때

- 치즈 스틱 0 : 연주 속도를 120 BPM으로 정하기
- 2 번 반복하기
- 치즈 스틱 0 : 도 4 음을 1 박자 연주하기
- 치즈 스틱 0 : 미 4 음을 1 박자 연주하기
- 치즈 스틱 0 : 솔 4 음을 1 박자 연주하기
- 치즈 스틱 0 : 라 4 음을 1 박자 연주하기
- 치즈 스틱 0 : 라 4 음을 1 박자 연주하기
- 치즈 스틱 0 : 라 4 음을 1 박자 연주하기
- 치즈 스틱 0 : 솔 4 음을 1 박자 연주하기

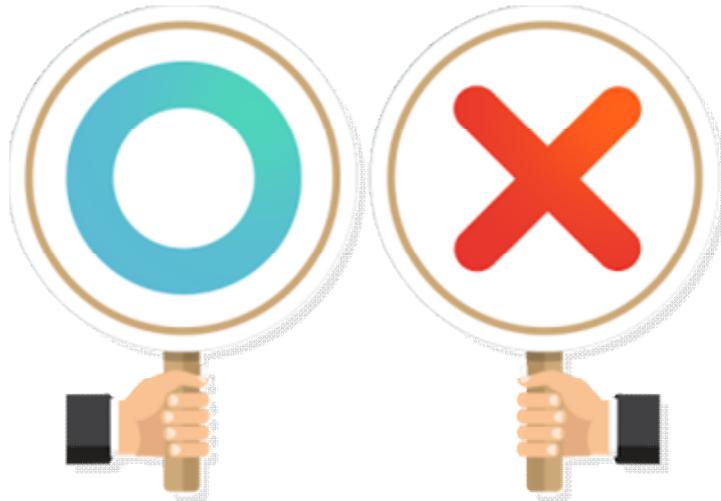
Basic Programing

2 O,X 퀴즈 버튼 만들기

5x5 LED 매트릭스를 사용하여 O, X 퀴즈 버튼을 만들 수 있어요.



5x5 LED 매트릭스의 A, B버튼을 사용하여 O, X 퀴즈 버튼을 만들 수 있어요. 5x5 LED 매트릭스에 대하여 알아보시다.



준비를 해봅시다.



컴퓨터 또는
노트북



USB 동글



치즈스틱 본체



5x5 LED
매트릭스



사용할 블록을 알아보시다.

시작하기 버튼을 클릭했을 때

시작하기 버튼을 클릭하면 아래에 연결된 블록들을 실행합니다.

계속 반복하기

감싸고 있는 블록들을 계속해서 반복 실행합니다.

만일 참 (이)라면

만일 판단이 참이면, 감싸고 있는 블록들을 실행합니다.

참 그리고 참

두 판단이 모두 참인 경우 '참'으로 판단합니다.

치즈 스틱 : HAT-010 5x5 매트릭스 시작하기

5x5 LED 매트릭스를 시작할 때 사용합니다.

치즈 스틱 : HAT-010 버튼 A 를 클릭했는가 ?

5x5 LED 매트릭스의 버튼을 클릭할 때 사용합니다.

치즈 스틱 : HAT-010 그림 을(를) 빨간색 사각형 (으)로 정하기

5x5 LED 매트릭스에 여러 가지 그림을 표시합니다.

치즈 스틱 : HAT-010 그림 지우기

5x5 LED 매트릭스의 그림을 지웁니다.

치즈 스틱 : 소리 번 재생하기

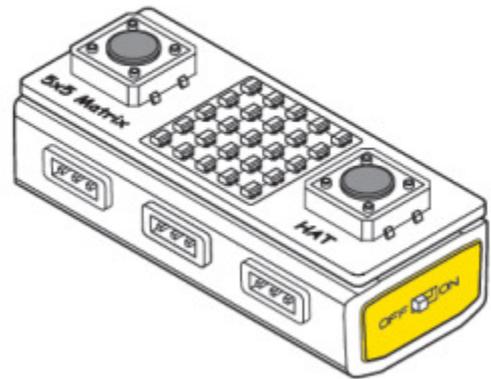
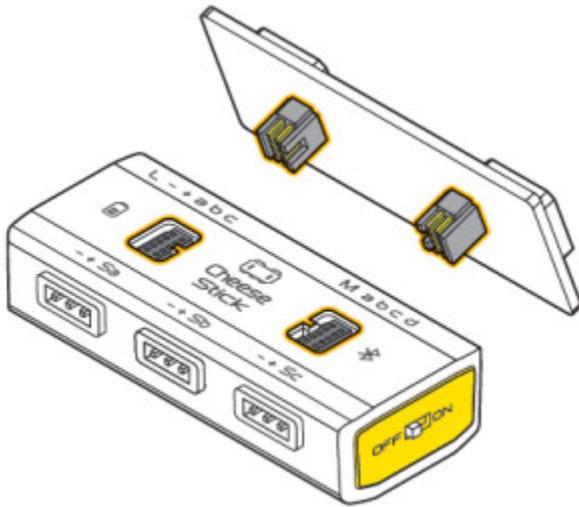
지정한 소리를 입력한 횟수만큼 재생합니다.



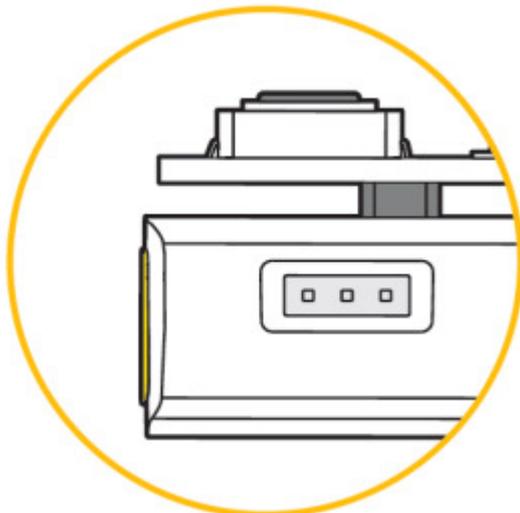
5×5 LED 매트릭스를 연결해 봅시다.



5×5 LED 매트릭스의 연결 단자를 치즈스틱의 HAT 포트에 맞춰 꽂아줍니다.



5×5 LED 매트릭스는 치즈스틱 본체에 꽂는 방향의 구분 없이 사용이 가능합니다.



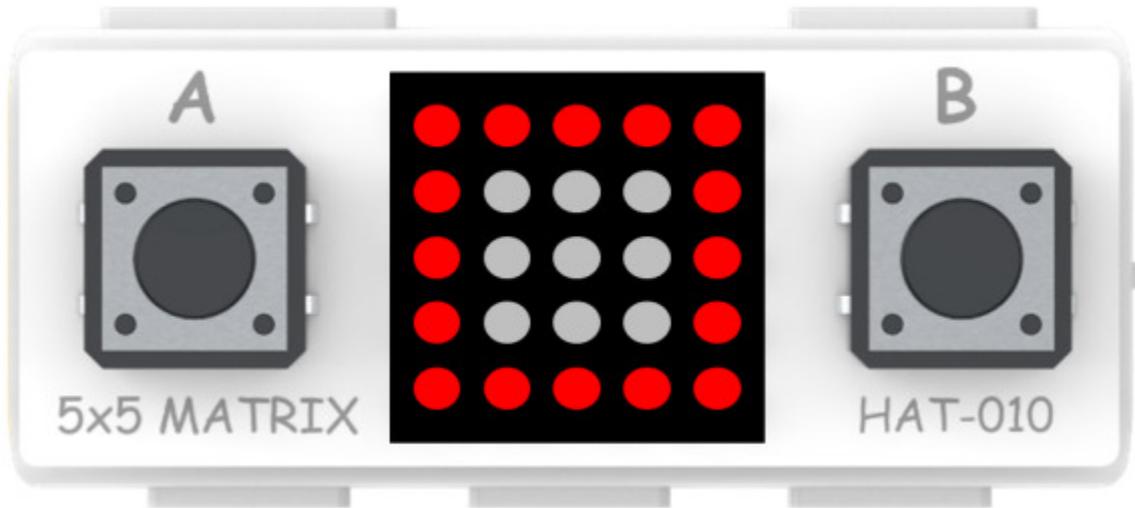
< 측면 연결 모습 >



5×5 LED 매트릭스를 HAT 포트에 꽂았을 때, 치즈스틱 본체와 부품 간에 조금의 간격이 있는 것이 정상입니다.



5x5 LED 매트릭스에 '사각형'을 표현해 봅시다.



▶ 시작하기 버튼을 클릭했을 때



카테고리에서 [시작하기 버튼을 클릭했을 때] 블록을 가져옵니다.

▶ 시작하기 버튼을 클릭했을 때
치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기



카테고리에서 [HAT-010 5x5 매트릭스 시작하기] 블록을 연결합니다.

▶ 시작하기 버튼을 클릭했을 때
치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
치즈 스틱 0 : HAT-010 그림 1 을(를) 빨간색 ▾ 사각형 ▾ (으)로 정하기



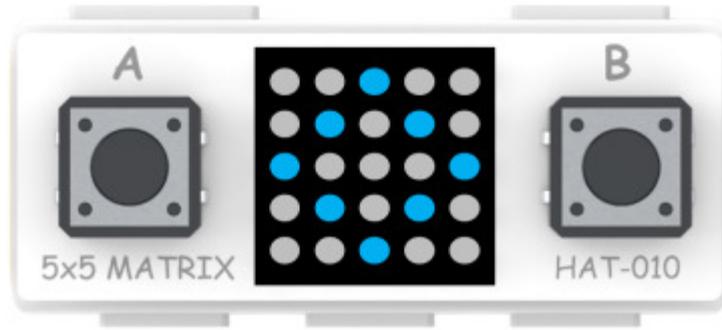
카테고리에서 'HAT-010 그림 1을(를) 빨간색 사각형으로 정하기' 블록을 연결합니다.



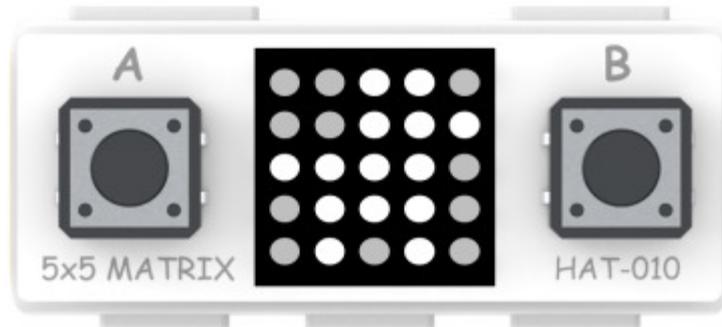
5x5 LED 매트릭스를 사용하기 위해서는

반드시 치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기 블록을 사용해야 합니다.

[연습과제 1] 5x5 LED 매트릭스에 '파란색 다이아몬드' 표현하기



[연습과제 2] 5x5 LED 매트릭스에 '하얀색 강아지' 표현하기



확인하기

[연습과제 1]

시작하기 버튼을 클릭했을 때

치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기

치즈 스틱 0 : HAT-010 그림 1 울(를) 파란색 ▶ 다이아몬드 ▶ (으)로 정하기

[연습과제 2]

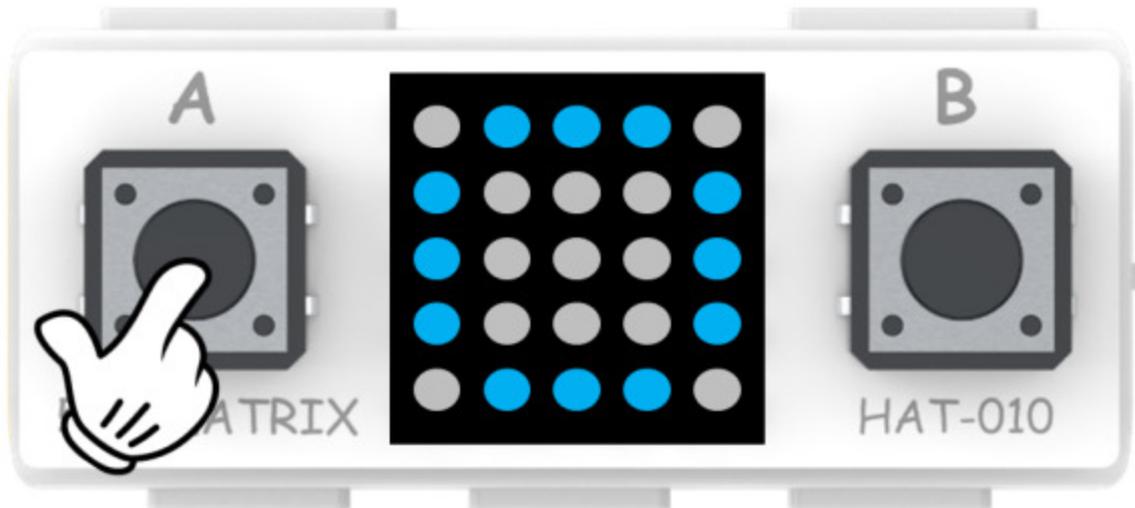
시작하기 버튼을 클릭했을 때

치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기

치즈 스틱 0 : HAT-010 그림 1 울(를) 하얀색 ▶ 강아지 ▶ (으)로 정하기



5x5 LED 매트릭스의 'A버튼'을 클릭하면 파란색으로 '0'가 표시되도록 해봅시다.



```

▶ 시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  계속 반복하기
    만일 치즈 스틱 0 : HAT-010 버튼 A 를 클릭했는가 ? (이)라면
      치즈 스틱 0 : HAT-010 그림 1 을(를) 파란색 원 (으)로 정하기
  
```



버튼을 클릭했을 때 5x5 LED 매트릭스가 켜지도록 하기 위해서

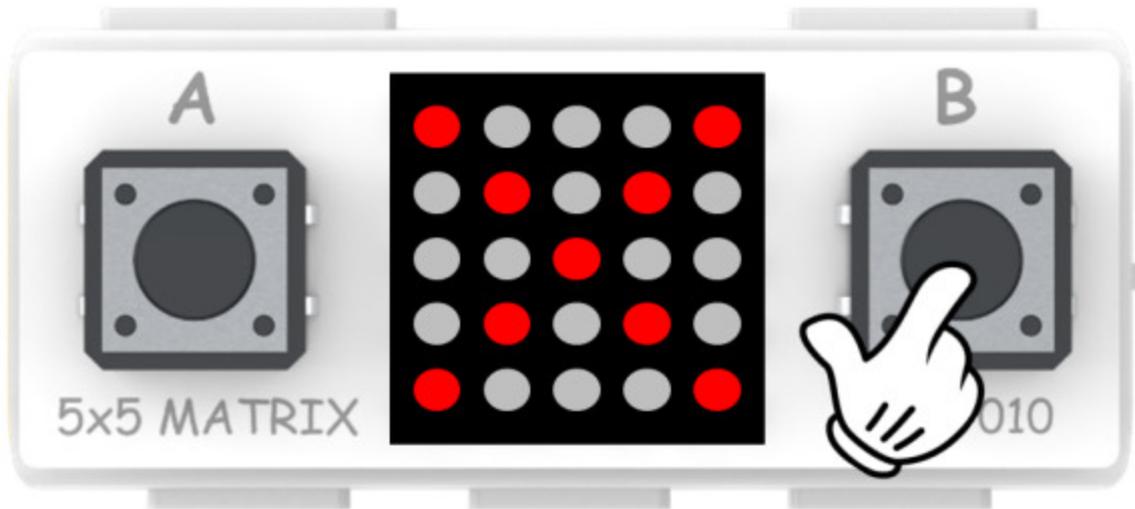
```

만일 참 (이)라면
  과 치즈 스틱 0 : HAT-010 버튼 A 를 클릭했는가 ? 블록을
  
```

사용합니다.

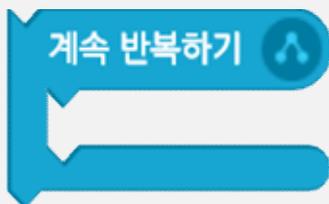


5x5 LED 매트릭스의 'B버튼'을 클릭하면 빨간색으로 'X'가 표시되도록 해봅시다.

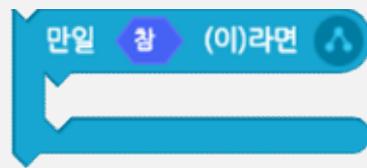


```

▶ 시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  계속 반복하기
    만일 치즈 스틱 0 : HAT-010 버튼 A 를 클릭했는가 ? (이)라면
      치즈 스틱 0 : HAT-010 그림 1 을(를) 파란색 원 (으)로 정하기
    만일 치즈 스틱 0 : HAT-010 버튼 B 를 클릭했는가 ? (이)라면
      치즈 스틱 0 : HAT-010 그림 1 을(를) 빨간색 X (으)로 정하기
  
```



블록에

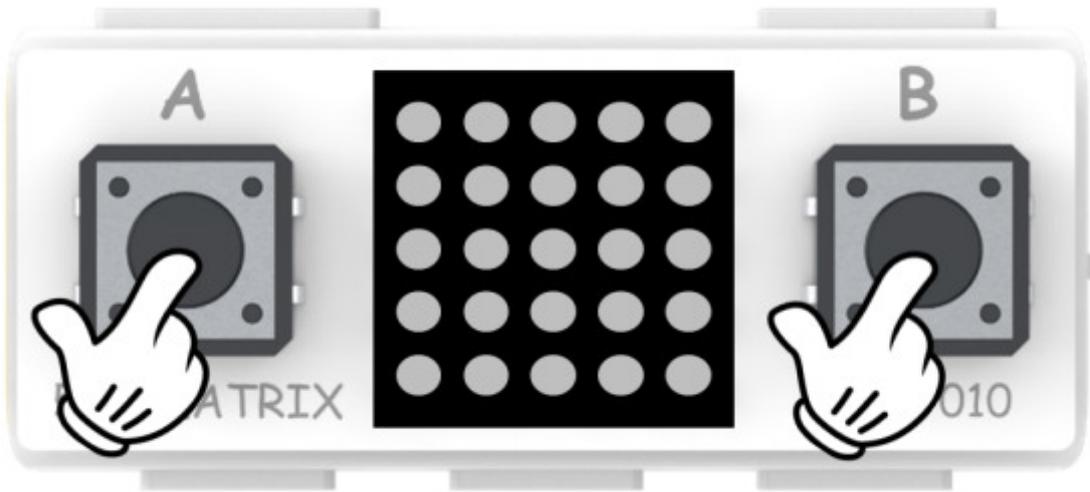


블록을

1개 더 추가하여 프로그래밍 합니다.



5x5 LED 매트릭스의 A버튼과 B버튼을 동시에 클릭하면 O 또는 X 표시가 사라지도록 해봅시다.



```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시각하기
  계속 반복하기
    만일 치즈 스틱 0 : HAT-010 버튼 A 를 클릭했는가 ? (이)라면
      치즈 스틱 0 : HAT-010 그림 1 을(를) 파란색 원 (으)로 정하기
    만일 치즈 스틱 0 : HAT-010 버튼 B 를 클릭했는가 ? (이)라면
      치즈 스틱 0 : HAT-010 그림 1 을(를) 빨간색 X (으)로 정하기
    만일 치즈 스틱 0 : HAT-010 버튼 A 를 클릭했는가 ? 그리고 치즈 스틱 0 : HAT-010 버튼 B 를 클릭했는가 ? (이)라면
      치즈 스틱 0 : HAT-010 그림 1 지우기
  
```



A버튼과 B버튼을 동시에 선택했을 때를 구현하기 위해서



판단

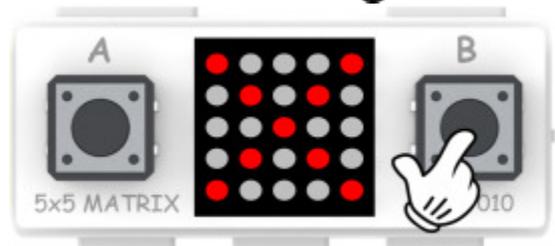
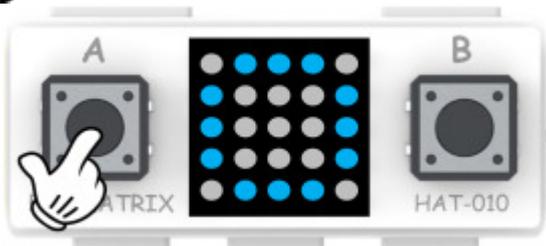
카테고리에 있는



블록을 사용합니다.



5x5 LED 매트릭스의 버튼을 클릭했을 때, O, X 표시와 함께 ‘특정 소리’가 재생되도록 표현해 봅시다.



```

▶ 시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  계속 반복하기
    만일 치즈 스틱 0 : HAT-010 버튼 A 를 클릭했는가 ? (이)라면
      치즈 스틱 0 : 삐 소리 1 번 재생하기
      치즈 스틱 0 : HAT-010 그림 1 을(를) 파란색 원 (으)로 정하기
    만일 치즈 스틱 0 : HAT-010 버튼 B 를 클릭했는가 ? (이)라면
      치즈 스틱 0 : 삐 소리 1 번 재생하기
      치즈 스틱 0 : HAT-010 그림 1 을(를) 빨간색 X (으)로 정하기
    만일 치즈 스틱 0 : HAT-010 버튼 A 를 클릭했는가 ? 그리고 치즈 스틱 0 : HAT-010 버튼 B 를 클릭했는가 ? (이)라면
      치즈 스틱 0 : 잘 했어요 소리 1 번 재생하기
      치즈 스틱 0 : HAT-010 그림 1 지우기
  
```



버튼을 클릭했을 때 O, X 표시와 함께 동시에 특정 소리를 재생하기

위해서



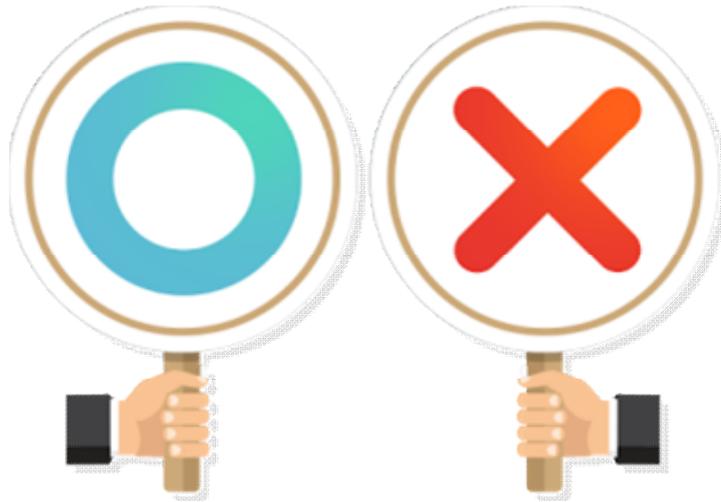
카테고리에 있는



블록을 사용합니다.



학습문제로 공부한 내용을 정리해 봅시다.



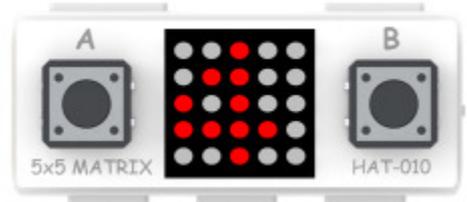
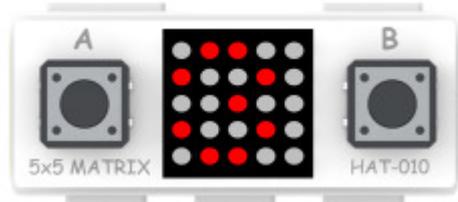
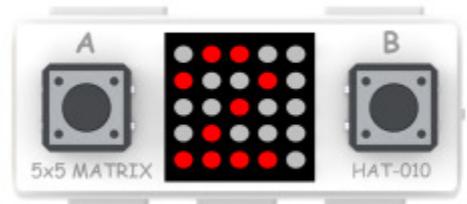
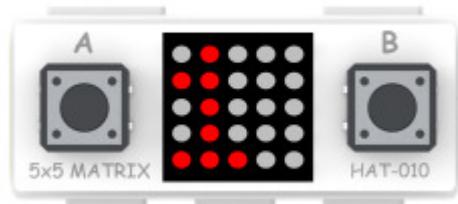
[문제1] 치즈스틱 버튼을 사용하여 친구들과 함께 O, X 퀴즈 대결을 해봅시다.

< O, X 퀴즈로 알아보는 '생활 속 거리두기 수칙' >

1. 열이 나거나 기침, 가래, 근육통 등 호흡기 증상이 있으면 외출한다. (X)
- 집에 머물며 3~4일간 쉰다.
2. 사람과 사람 사이에는 두 팔 간격으로 충분한 간격을 둔다. (O)
- 코로나 19는 주로 침방울을 통해 전파되므로 사람과 사람사이 거리를 2M 이상 두는 경우 위험이 줄어든다.
3. 손을 자주 꼼꼼히 씻고, 기침 할 때 옷소매로 가린다. (O)
- 오염된 손을 거쳐 바이러스가 몸에 들어오는 것을 막고, 기침예절을 지켜 침방울을 통한 전파도 최소화한다.
4. 환기와 소독은 자주하지 않아도 된다. (X)
- 매일 2번 이상 환기하고, 주기적으로 소독한다.
5. 거리는 멀어져도 마음은 가까이 한다. (O)
- 코로나 19는 나 뿐 아니라 우리 모두의 노력이 있어야 극복할 수 있다. 서로를 배려하고 위로하며 함께 노력하는 사회를 만들어야 한다.



학습문제로 공부한 내용을 정리해 봅시다.



[문제2] 보기와 같이 명령어를 작성하여 사지선다형 퀴즈 버튼을 만들어 봅시다.

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  계속 반복하기
    만일 1 > 키가 눌러져 있는가? (이)라면
      치즈 스틱 0 : 삐 소리 1 번 재생하기
      치즈 스틱 0 : HAT-010 그림 1 을(를) 빨간색 글자 1 (으)로 정하기
    만일 2 > 키가 눌러져 있는가? (이)라면
      치즈 스틱 0 : 삐 소리 1 번 재생하기
      치즈 스틱 0 : HAT-010 그림 1 을(를) 빨간색 글자 2 (으)로 정하기
    만일 3 > 키가 눌러져 있는가? (이)라면
      치즈 스틱 0 : 삐 소리 1 번 재생하기
      치즈 스틱 0 : HAT-010 그림 1 을(를) 빨간색 글자 3 (으)로 정하기
    만일 4 > 키가 눌러져 있는가? (이)라면
      치즈 스틱 0 : 삐 소리 1 번 재생하기
      치즈 스틱 0 : HAT-010 그림 1 을(를) 빨간색 글자 4 (으)로 정하기
  만일 스페이스 > 키가 눌러져 있는가? (이)라면
    치즈 스틱 0 : 잘 했어요 소리 1 번 재생하기
    치즈 스틱 0 : HAT-010 그림 1 지우기
  
```

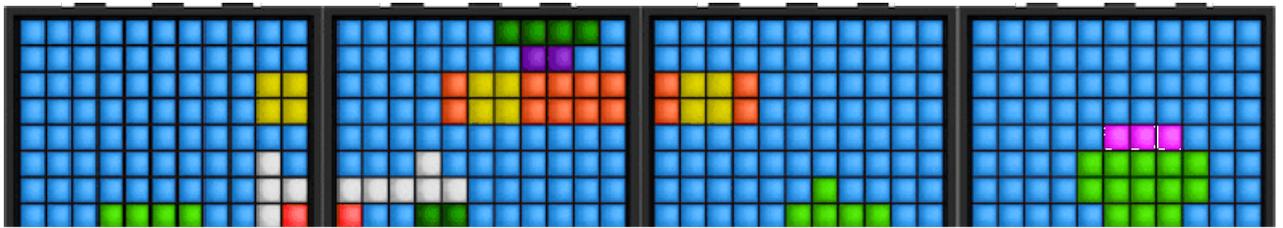
Basic Programing

3 나만의 애니메이션 만들기

5x5 LED 매트릭스를 사용하여 나만의 간단한 애니메이션 효과를 만들 수 있어요.



5x5 LED 매트릭스를 사용하여 나만의 간단한 애니메이션 효과를 만들 수 있어요. 25개의 LED를 프로그래밍 하는 방법에 대하여 알아보시다.



준비를 해봅시다.



컴퓨터 또는
노트북



USB 동글



치즈스틱 본체



5x5 LED
매트릭스



사용할 블록을 알아보시다.

시작하기 버튼을 클릭했을 때

시작하기 버튼을 클릭하면 아래에 연결된 블록들을 실행합니다.

계속 반복하기

감싸고 있는 블록들을 계속해서 반복 실행합니다.

2 초 기다리기

설정된 시간만큼 기다린 후 다음 블록을 실행합니다.

HAT-010 5x5 매트릭스 시작하기

5x5 LED 매트릭스를 시작할 때 사용합니다.

치즈 스틱 0 : HAT-010 배경 x: 0 y: 0 빨간색 ▾ 으로 켜기

가로 X축과 세로 Y축에 입력한 위치의 LED를 켵니다.

치즈 스틱 0 : HAT-010 배경 ▾ x: 0 y: 0 만큼 이동하기

배경을 지정한 위치만큼 이동합니다.

치즈 스틱 0 : HAT-010 배경 빨간색 ▾ 사각형 ▾ 음(음) x: 0 y: 0 에 그리기

지정한 배경을 입력한 위치에 나타냅니다.

치즈 스틱 0 : HAT-010 배경 빨간색 ▾ 사각형 ▾ 음(음) x: 0 y: 0 에 그리기

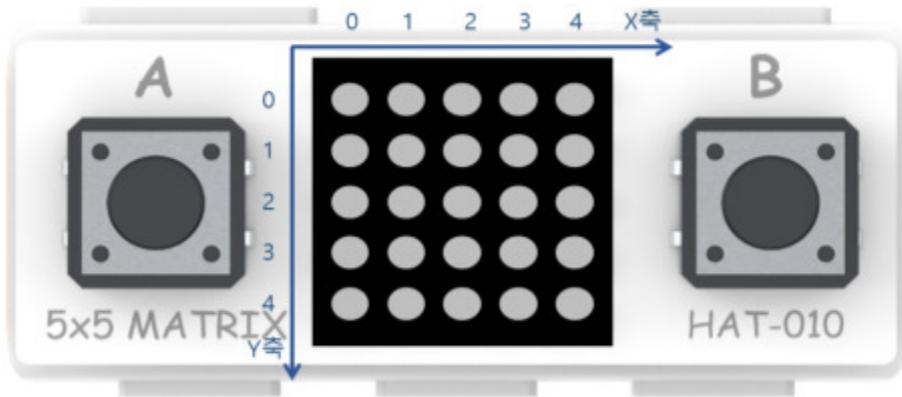
5x5 LED 매트릭스의 밝기를 지정한 비율만큼 정합니다.



5x5 LED 매트릭스의 LED를 각각 프로그래밍 해봅시다.



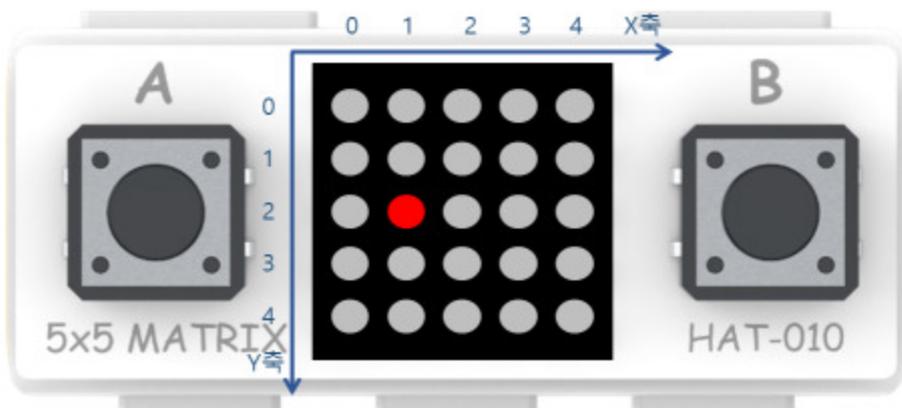
5x5 LED 매트릭스에는 총 25개의 LED가 있습니다.



25개의 LED는 가로 X축과 세로 Y축의 값을 사용하여 각각 색깔과 밝기를 제어할 수 있습니다.

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  치즈 스틱 0 : HAT-010 배경 x: 0 y: 0 빨간색 으로 켜기
  
```

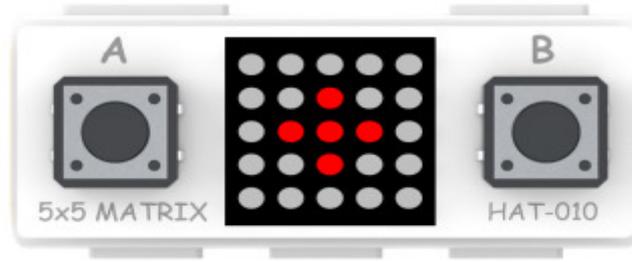


치즈 스틱 0 : HAT-010 배경 x: 0 y: 0 빨간색 으로 켜기

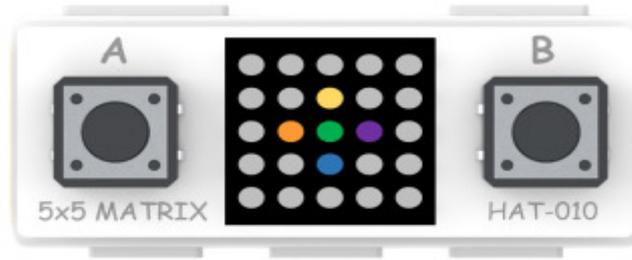
블록은

25개의 LED 가운데 특정 LED를 지정한 색상으로 켤 수 있습니다.

[연습과제 1] 5x5 LED 매트릭스에 빨간색으로 아래 그림 표현하기



[연습과제 2] 5x5 LED 매트릭스에 여러 가지 색으로 아래 그림 표현하기



확인하기

[연습과제 1]

▶ 시작하기 버튼을 클릭했을 때

- 치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
- 치즈 스틱 0 : HAT-010 배경 x: 1 y: 2 빨간색 ▾ 으로 켜기
- 치즈 스틱 0 : HAT-010 배경 x: 2 y: 1 빨간색 ▾ 으로 켜기
- 치즈 스틱 0 : HAT-010 배경 x: 2 y: 2 빨간색 ▾ 으로 켜기
- 치즈 스틱 0 : HAT-010 배경 x: 2 y: 3 빨간색 ▾ 으로 켜기
- 치즈 스틱 0 : HAT-010 배경 x: 3 y: 2 빨간색 ▾ 으로 켜기

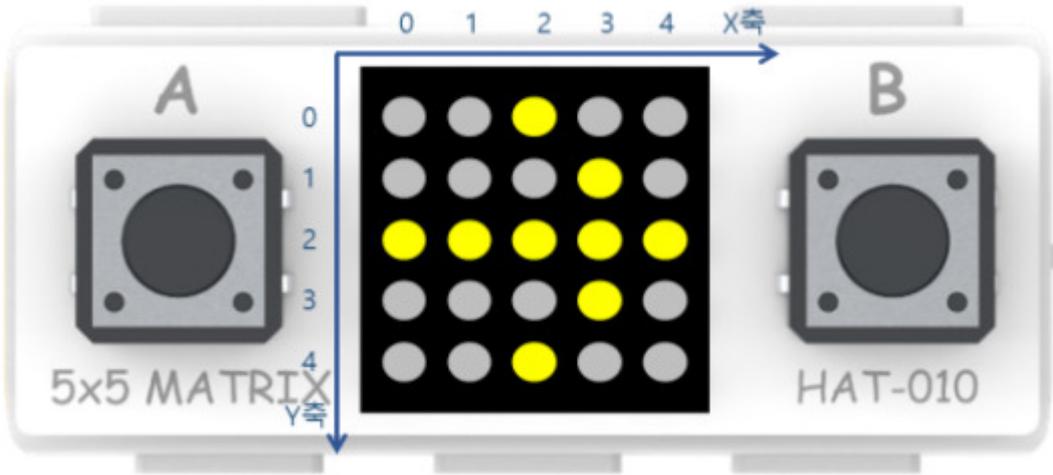
[연습과제 2]

▶ 시작하기 버튼을 클릭했을 때

- 치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
- 치즈 스틱 0 : HAT-010 배경 x: 1 y: 2 주황색 ▾ 으로 켜기
- 치즈 스틱 0 : HAT-010 배경 x: 2 y: 1 노란색 ▾ 으로 켜기
- 치즈 스틱 0 : HAT-010 배경 x: 2 y: 2 초록색 ▾ 으로 켜기
- 치즈 스틱 0 : HAT-010 배경 x: 2 y: 3 파란색 ▾ 으로 켜기
- 치즈 스틱 0 : HAT-010 배경 x: 3 y: 2 보라색 ▾ 으로 켜기



5x5 LED 매트릭스에 '오른쪽 화살표'를 표현해 봅시다.



▶ 시작하기 버튼을 클릭했을 때

- 치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
- 치즈 스틱 0 : HAT-010 밝기를 50 %로 정하기
- 치즈 스틱 0 : HAT-010 배경 x: 0 y: 2 노란색 ▾ 으로 켜기
- 치즈 스틱 0 : HAT-010 배경 x: 1 y: 2 노란색 ▾ 으로 켜기
- 치즈 스틱 0 : HAT-010 배경 x: 2 y: 0 노란색 ▾ 으로 켜기
- 치즈 스틱 0 : HAT-010 배경 x: 2 y: 2 노란색 ▾ 으로 켜기
- 치즈 스틱 0 : HAT-010 배경 x: 2 y: 4 노란색 ▾ 으로 켜기
- 치즈 스틱 0 : HAT-010 배경 x: 3 y: 1 노란색 ▾ 으로 켜기
- 치즈 스틱 0 : HAT-010 배경 x: 3 y: 2 노란색 ▾ 으로 켜기
- 치즈 스틱 0 : HAT-010 배경 x: 3 y: 3 노란색 ▾ 으로 켜기
- 치즈 스틱 0 : HAT-010 배경 x: 4 y: 2 노란색 ▾ 으로 켜기



LED의 기본 밝기는 20%입니다.

```
치즈 스틱 0 : HAT-010 밝기를 20 %로 정하기
```

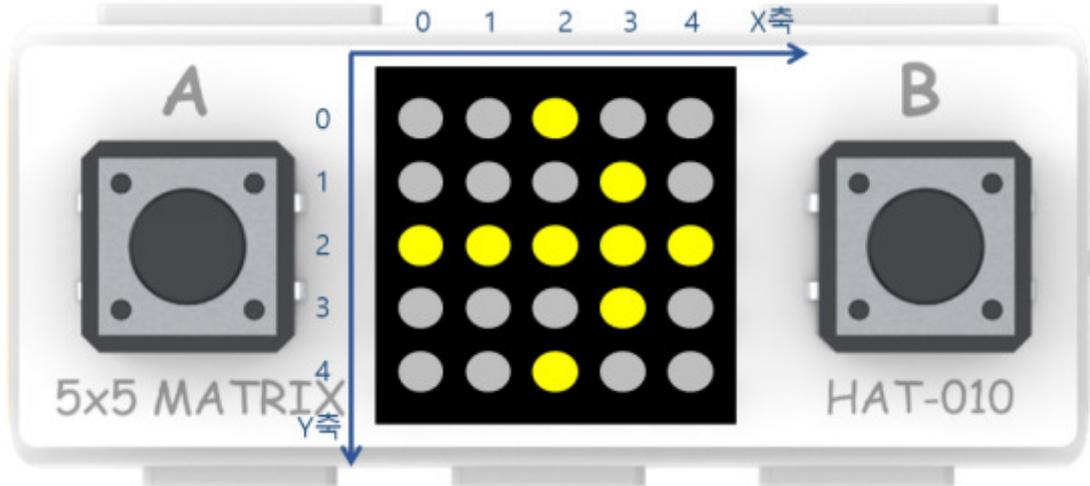
블록을 사용하면

LED의 밝기를 조절할 수 있습니다.

단, LED 빛의 세기를 강하게 하여 직접 바라볼 시 눈에 영향을 줄 수 있으므로 반드시 빛의 세기를 약하게 사용해야 합니다.



다른 방법으로 5x5 LED 매트릭스에 '오른쪽 화살표'를 표현해 봅시다.



▶ 시작하기 버튼을 클릭했을 때

- 치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
- 치즈 스틱 0 : HAT-010 배경 노란색 패턴 00100 을(를) x: 0 y: 0 에 그리기
- 치즈 스틱 0 : HAT-010 배경 노란색 패턴 00010 을(를) x: 0 y: 1 에 그리기
- 치즈 스틱 0 : HAT-010 배경 노란색 패턴 11111 을(를) x: 0 y: 2 에 그리기
- 치즈 스틱 0 : HAT-010 배경 노란색 패턴 00010 을(를) x: 0 y: 3 에 그리기
- 치즈 스틱 0 : HAT-010 배경 노란색 패턴 00100 을(를) x: 0 y: 4 에 그리기

💡 패턴 블록은 0과 1로 이루어진 컴퓨터언어 이진수를 사용하여 LED를 켜거나 끌 수 있습니다.

가로 X축과 세로 Y축의 위치에 5자리의 이진수를 사용하여 배경을 그립니다.

- 치즈 스틱 0 : HAT-010 배경 빨간색 패턴 10010 을(를) x: 0 y: 0 에 그리기



또 다른 방법으로 5x5 LED 매트릭스에 '오른쪽 화살표'를 표현해 봅시다.



▶ 시작하기 버튼을 클릭했을 때

```

치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
치즈 스틱 0 : HAT-010 배경 노란색 오른쪽 화살표 음(를) x: 0 y: 0 에 그리기
  
```



배경 블록은 그림 블록과 다르게 여러 장의 배경(레이어)을 순서대로 겹쳐서 하나의 이미지로 표현할 수 있습니다.

<그림 블록>

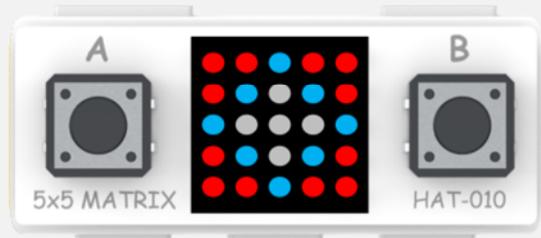
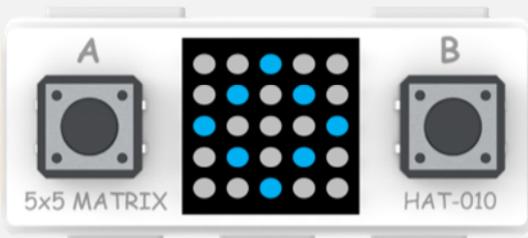
<배경 블록>

```

▶ 시작하기 버튼을 클릭했을 때
치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
치즈 스틱 0 : HAT-010 그림 1 음(를) 빨간색 사각형 (으)로 정하기
치즈 스틱 0 : HAT-010 그림 1 음(를) 파란색 다이에본드 (으)로 정하기
  
```

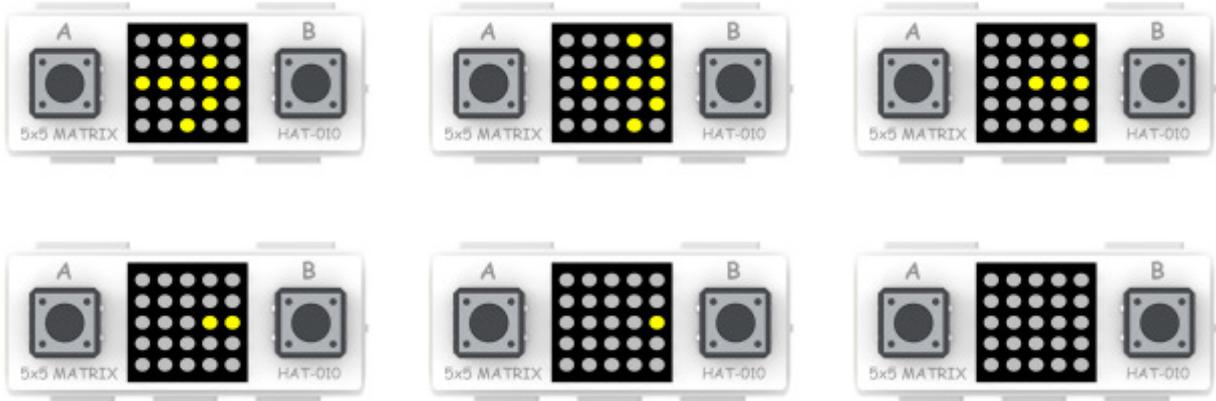
```

▶ 시작하기 버튼을 클릭했을 때
치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
치즈 스틱 0 : HAT-010 배경 빨간색 사각형 음(를) x: 0 y: 0 에 그리기
치즈 스틱 0 : HAT-010 배경 파란색 다이에본드 음(를) x: 0 y: 0 에 그리기
  
```





5x5 LED 매트릭스에 '움직이는 화살표'를 만들어 LED 비상구를 표현해 봅시다.



```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  계속 반복하기
    치즈 스틱 0 : HAT-010 배경 노란색 오른쪽 화살표 음(를) x: 0 y: 0 에 그리기
    6 번 반복하기
      0.2 초 기다리기
      치즈 스틱 0 : HAT-010 배경 x: 1 y: 0 만큼 이동하기
  
```



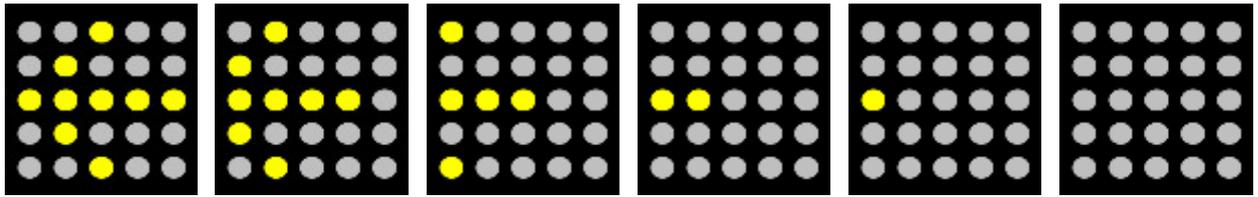
화살표가 좌, 우 또는 상, 하로 움직이는 효과를 나타내기 위해서

```

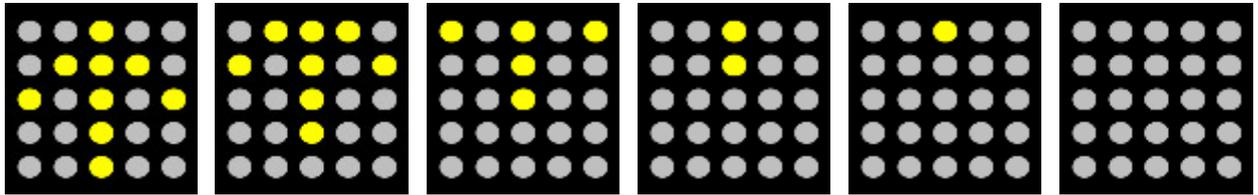
10 번 반복하기
  와
  2 초 기다리기
  와
  치즈 스틱 0 : HAT-010 배경 x: 0 y: 0 만큼 이동하기
  
```

블록을 사용합니다.

[연습과제 1] '왼쪽으로' 움직이는 화살표 애니메이션 만들기



[연습과제 2] '위로' 움직이는 화살표 애니메이션 만들기



확인하기

[연습과제 1]

```

[시작하기 버튼을 클릭했을 때]
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  계속 반복하기
    치즈 스틱 0 : HAT-010 배경 노란색 오른쪽 화살표 을(를) x: 0 y: 0 에 그리기
    6 번 반복하기
      0.2 초 기다리기
      치즈 스틱 0 : HAT-010 배경 x: -1 y: 0 만큼 이동하기
  
```

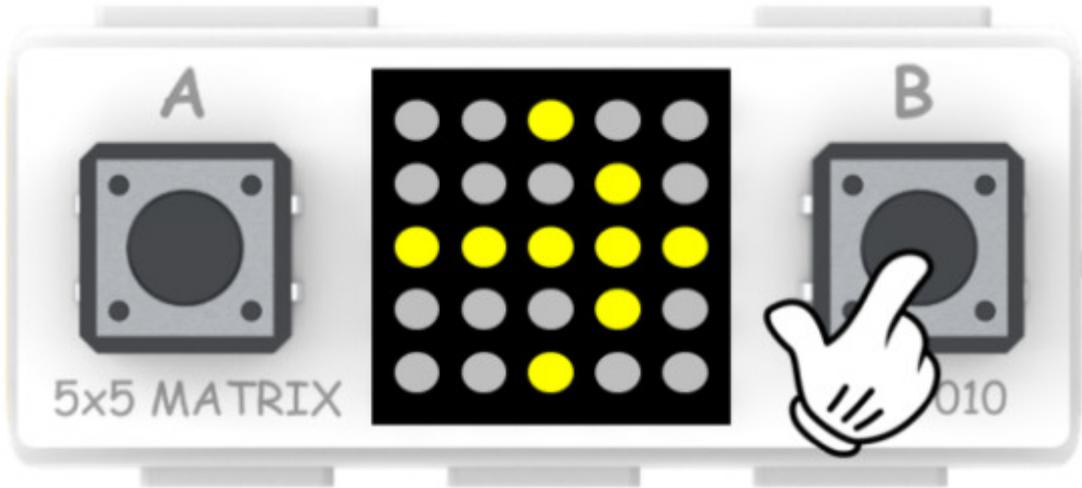
[연습과제 2]

```

[시작하기 버튼을 클릭했을 때]
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  계속 반복하기
    치즈 스틱 0 : HAT-010 배경 노란색 오른쪽 화살표 을(를) x: 0 y: 0 에 그리기
    6 번 반복하기
      0.2 초 기다리기
      치즈 스틱 0 : HAT-010 배경 x: 0 y: -1 만큼 이동하기
  
```



5x5 LED 매트릭스의 A, B버튼을 클릭하면 화살표가 왼쪽 또는 오른쪽으로 움직이는 애니메이션 효과를 만들어 봅시다.



```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  계속 반복하기
    만일 치즈 스틱 0 : HAT-010 버튼 A 를 클릭했는가 ? (이)라면
      치즈 스틱 0 : HAT-010 배경 노란색 왼쪽 화살표 을(를) x: 0 y: 0 에 그리기
      6 번 반복하기
        0.2 초 기다리기
        치즈 스틱 0 : HAT-010 배경 x: -1 y: 0 만큼 이동하기
        치즈 스틱 0 : HAT-010 배경 지우기
    만일 치즈 스틱 0 : HAT-010 버튼 B 를 클릭했는가 ? (이)라면
      치즈 스틱 0 : HAT-010 배경 노란색 오른쪽 화살표 을(를) x: 0 y: 0 에 그리기
      6 번 반복하기
        0.2 초 기다리기
        치즈 스틱 0 : HAT-010 배경 x: 1 y: 0 만큼 이동하기
        치즈 스틱 0 : HAT-010 배경 지우기
  
```



버튼을 클릭하여 화살표가 표시되고 난 후에, `치즈 스틱 0 : HAT-010 배경 지우기` 를 사용하여 처음에 그려진 배경을 지워야 다음 화살표를 표시할 수 있습니다.



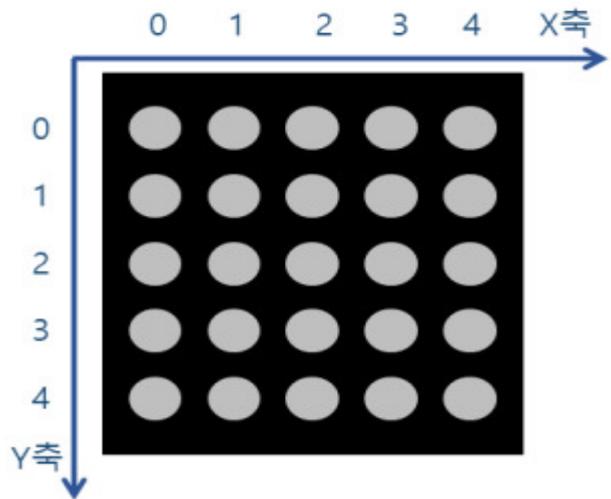
학습문제로 공부한 내용을 정리해 봅시다.

[문제1] 보기와 같이 명령어를 작성했을 때 LED의 모습을 표시해 봅시다.

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  치즈 스틱 0 : HAT-010 배경 x: 0 y: 3 빨간색으로 켜기
  치즈 스틱 0 : HAT-010 배경 x: 1 y: 2 빨간색으로 켜기
  치즈 스틱 0 : HAT-010 배경 x: 1 y: 3 빨간색으로 켜기
  치즈 스틱 0 : HAT-010 배경 x: 2 y: 1 빨간색으로 켜기
  치즈 스틱 0 : HAT-010 배경 x: 2 y: 3 빨간색으로 켜기
  치즈 스틱 0 : HAT-010 배경 x: 3 y: 2 빨간색으로 켜기
  치즈 스틱 0 : HAT-010 배경 x: 3 y: 3 빨간색으로 켜기
  치즈 스틱 0 : HAT-010 배경 x: 4 y: 3 빨간색으로 켜기

```

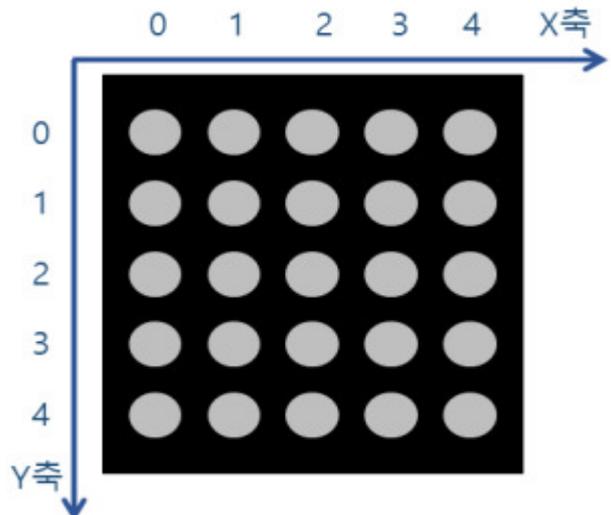


[문제2] 보기와 같이 명령어를 작성했을 때 LED의 모습을 표시해 봅시다.

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  치즈 스틱 0 : HAT-010 배경 빨간색 사각형 음(홀) x: 1 y: 1 에 그리기

```



[문제3] 보기와 같이 명령어를 작성하여 '테트리스 게임 댄스'를 만들어 봅시다.



▶ 시작하기 버튼을 클릭했을 때

치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기

계속 반복하기

- 치즈 스틱 0 : HAT-010 배경 자주색 ▼ 패턴 00101 울(름) x: 0 y: 0 에 그리기
- 치즈 스틱 0 : HAT-010 배경 자주색 ▼ 패턴 11111 울(름) x: 0 y: 1 에 그리기
- 치즈 스틱 0 : HAT-010 배경 자주색 ▼ 패턴 10100 울(름) x: 0 y: 2 에 그리기
- 치즈 스틱 0 : HAT-010 배경 자주색 ▼ 패턴 01110 울(름) x: 0 y: 3 에 그리기
- 치즈 스틱 0 : HAT-010 배경 자주색 ▼ 패턴 01010 울(름) x: 0 y: 4 에 그리기

0.5 초 기다리기

- 치즈 스틱 0 : HAT-010 배경 자주색 ▼ 패턴 10100 울(름) x: 0 y: 0 에 그리기
- 치즈 스틱 0 : HAT-010 배경 자주색 ▼ 패턴 11111 울(름) x: 0 y: 1 에 그리기
- 치즈 스틱 0 : HAT-010 배경 자주색 ▼ 패턴 00101 울(름) x: 0 y: 2 에 그리기
- 치즈 스틱 0 : HAT-010 배경 자주색 ▼ 패턴 01110 울(름) x: 0 y: 3 에 그리기
- 치즈 스틱 0 : HAT-010 배경 자주색 ▼ 패턴 01010 울(름) x: 0 y: 4 에 그리기

0.5 초 기다리기

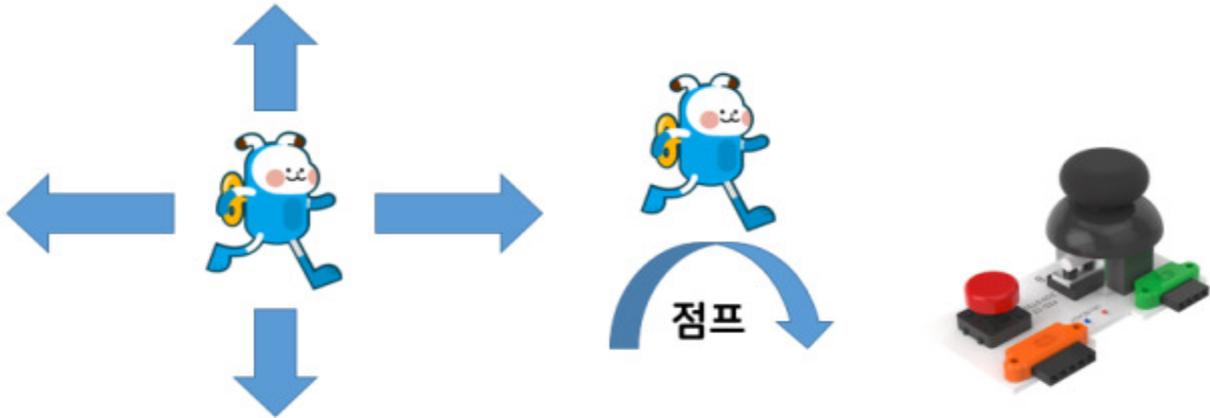
Basic Programing

4 엔트리봇 조종기 만들기

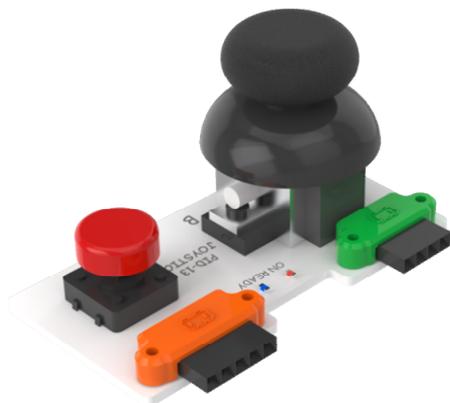
조이스틱을 사용해 엔트리 오브젝트의 동작을 조종할 수 있어요.



조이스틱을 사용하면 엔트리 오브젝트의 다양한 움직임을 컨트롤 할 수 있어요. 조이스틱에 대하여 알아보시다.



준비를 해봅시다.



조이스틱



치즈스틱 본체



사용할 블록을 알아보시다.

치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기

치즈 스틱 0 : PID x1 치즈 스틱 0 : PID y1

치즈 스틱 0 : PID x1 의 범위 0 ~ 255 음(음) 0 ~ 100 소수점 없음 으로 경하기

치즈 스틱 0 : PID x1 의 범위 0 ~ 127 ~ 255 음(음) -100 ~ 0 ~ 100 소수점 없음 으로 경하기

10 > 10

참 또는 참

치즈 스틱 0 : PID 버튼 1 음(음) 클릭했는가 ?

2 초 동안 x: 10 y: 10 만큼 움직이기

x 좌표를 10 만큼 바꾸기

만일 참 (이)라면
아니면

조이스틱을 사용할 때 필요합니다.

조이스틱의 좌우, 상하값을 설정할 때 필요합니다.

조이스틱의 측정값을 0~100의 값으로 설정합니다.

조이스틱의 측정값을 세 단계의 값으로 설정합니다.

값의 크기 비교를 통해 명령을 설정합니다.

하나 또는 둘 모두 참인 경우 '참'으로 판단합니다.

버튼을 눌렀을 때 명령을 실행합니다.

주어진 시간 동안 오브젝트를 입력한 좌표값 만큼 이동합니다.

X 좌표값을 입력한 만큼 변경합니다.

주어진 조건을 만족했을 때와 만족하지 않았을 때 서로 다른 명령을 실행합니다.



조이스틱을 연결하고 값을 확인하는 방법을 알아보시다.



본체와 조이스틱의 방향을 맞추어 연결합니다.



조이스틱은 조이스틱과 버튼으로 구성되어 있으며 엔트리 오브젝트의 동작, 모양, 움직임 등을 제어할 수 있습니다.



X 좌표값이 점점 감소하여 0이 됩니다.



X 좌표값이 점점 증가하여 255가 됩니다.
(최대값은 기기에 따라 조금씩 다를 수 있습니다.)



조이스틱을 사용해 엔트리봇 오브젝트가 앞으로 움직이도록 해봅시다.

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  계속 반복하기
    만일 치즈 스틱 0 : PID x1 > 130 (이)라면
      2 초 동안 x: 50 y: 0 만큼 움직이기
  
```



앞으로 가기



조이스틱을 사용해 엔트리봇 오브젝트가 뒤로 움직이도록 해봅시다.

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  계속 반복하기
    만일 치즈 스틱 0 : PID x1 < 120 (이)라면
      2 초 동안 x: -50 y: 0 만큼 움직이기
  
```



뒤로 가기



조이스틱을 사용할 때는 반드시

```

치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  
```

블록을 사용해야 합니다.

[연습과제1]



위로 가기

[연습과제2]



아래로 가기

확인하기

[연습과제1]

```

▶ 시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  계속 반복하기
  만일 치즈 스틱 0 : PID y1 > 130 (아)라면
    2 초 동안 x: 0 y: 50 만큼 움직이기
  
```

[연습과제2]

```

▶ 시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  계속 반복하기
  만일 치즈 스틱 0 : PID y1 < 120 (아)라면
    2 초 동안 x: 0 y: -50 만큼 움직이기
  
```



오브젝트를 움직일 때 두 가지 블록의 쓰임새에 유의하여 블록을 사용해야 합니다.

```

2 초 동안 x: 10 y: 10 만큼 움직이기
  
```

조건이 맞을 경우 주어진 시간 동안 X 좌표값, Y 좌표값 만큼 이동합니다.

```

2 초 동안 x: 10 y: 10 위치로 이동하기
  
```

조건이 맞을 경우 주어진 시간 동안 X 좌표의 위치, Y 좌표의 위치로 이동합니다.



조이스틱을 사용해 엔트리봇을 앞뒤로 움직이도록 해봅시다.

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  치즈 스틱 0 : PID x1 의 범위 0 ~ 255 을(를) 0 ~ 100 소수점 없음 으로 정하기
  계속 반복하기
    만일 치즈 스틱 0 : PID x1 ≥ 50 (이)라면
      x 좌표를 10 만큼 바꾸기
    아니면
      x 좌표를 -10 만큼 바꾸기
  
```



조이스틱을 사용해 엔트리봇 오브젝트를 앞뒤로 움직일 때 화면 끝에 닿으면 처음 위치로 오도록 해봅시다.

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  치즈 스틱 0 : PID x1 의 범위 0 ~ 255 을(를) 0 ~ 100 소수점 없음 으로 정하기
  계속 반복하기
    만일 치즈 스틱 0 : PID x1 ≥ 50 (이)라면
      x 좌표를 10 만큼 바꾸기
    아니면
      x 좌표를 -10 만큼 바꾸기
    만일 벽 에 닿았는가? (이)라면
      x: 0 y: 0 위치로 이동하기
  
```



조이스틱을 사용할 때는 반드시

```

치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  
```

블록을 사용해야 합니다.



조이스틱을 사용해 엔트리봇 오브젝트를 앞뒤 3단계로 움직이도록 해봅시다.

```

▶ 시작하기 버튼을 클릭했을 때
치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
치즈 스틱 0 : PID x1 의 범위 0 ~ 127 ~ 255 을(를) -100 ~ 0 ~ 100 소수점 없음 으로 정하기
계속 반복하기
만일 <-10 < 치즈 스틱 0 : PID x1 > 그리고 < 치즈 스틱 0 : PID x1 > < 10 (이)라면
  x 좌표를 0 만큼 바꾸기
아니면
  만일 < 치즈 스틱 0 : PID x1 > > 10 (이)라면
    x 좌표를 10 만큼 바꾸기
  아니면
    x 좌표를 -10 만큼 바꾸기
  
```



조이스틱을 사용해 엔트리봇 오브젝트를 위아래 3단계로 움직이도록 해봅시다.

```

▶ 시작하기 버튼을 클릭했을 때
치즈 스틱 0 : PID y1 의 범위 0 ~ 127 ~ 255 을(를) -100 ~ 0 ~ 100 소수점 없음 으로 정하기
계속 반복하기
만일 <-50 < 치즈 스틱 0 : PID y1 > 그리고 < 치즈 스틱 0 : PID y1 > < 10 (이)라면
  y 좌표를 0 만큼 바꾸기
아니면
  만일 < 치즈 스틱 0 : PID y1 > > 50 (이)라면
    y 좌표를 10 만큼 바꾸기
  아니면
    y 좌표를 -10 만큼 바꾸기
  
```





조이스틱의 버튼을 사용해 엔트리봇 오브젝트의 모양이 변하도록 해봅시다.

▶ 시작하기 버튼을 클릭했을 때

치즈 스틱 0 : PID-13 조이스틱과 버튼 ▶ 시작하기

계속 반복하기

만일 치즈 스틱 0 : PID 버튼 2 ▶ 올(를) 클릭했는가 ▶ ? (이)라면

다음 ▶ 모양으로 바꾸기



조이스틱의 버튼을 사용해 엔트리봇 오브젝트가 점프하도록 해봅시다.

▶ 시작하기 버튼을 클릭했을 때

치즈 스틱 0 : PID-13 조이스틱과 버튼 ▶ 시작하기

계속 반복하기

만일 치즈 스틱 0 : PID 버튼 2 ▶ 올(를) 클릭했는가 ▶ ? (이)라면

1 초 동안 x: 30 y: 50 만큼 움직이기

1 초 동안 x: 30 y: -50 만큼 움직이기





조이스틱과 버튼을 함께 사용해서 엔트리봇 오브젝트가 움직이도록 해봅시다.

시작하기 버튼을 클릭했을 때

치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기

치즈 스틱 0 : PID x1 의 범위 0 ~ 127 ~ 255 을(를) -100 ~ 0 ~ 100 소수점 없음 으로 정하기

계속 반복하기

만일 치즈 스틱 0 : PID x1 < -10 그리고 치즈 스틱 0 : PID x1 < 10 (이)라면

x 좌표를 0 만큼 바꾸기

아니면

만일 치즈 스틱 0 : PID x1 > 10 (이)라면

x 좌표를 10 만큼 바꾸기

아니면

x 좌표를 -10 만큼 바꾸기

시작하기 버튼을 클릭했을 때

치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기

계속 반복하기

만일 치즈 스틱 0 : PID 버튼 2 을(를) 클릭했는가 ? (이)라면

1 초 동안 x: 30 y: 50 만큼 움직이기

1 초 동안 x: 30 y: -50 만큼 움직이기



두 개의 조건을 동시에 실행하기 위해서는 블록을 동시에 실행하려는 각각의 명령 블록에 연결해 주어야 합니다.

시작하기 버튼을 클릭했을 때



학습문제로 공부한 내용을 정리해 봅시다.

[문제1] 보기와 같이 명령어를 작성했을 때 엔트리봇 오브젝트의 움직임을 작성해 봅시다.

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID y1 의 범위 0 ~ 127 ~ 255 을(를) -100 ~ 0 ~ 100 소수점 없음 으로 정하기
  계속 반복하기
    만일 <-50 < 치즈 스틱 0 : PID x1 > 그리고 < 치즈 스틱 0 : PID x1 > < 10 (이)라면
      x 좌표를 10 만큼 바꾸기
    아니면
      만일 < 치즈 스틱 0 : PID x1 > > 50 (이)라면
        x 좌표를 10 만큼 바꾸기
      아니면
        x 좌표를 -10 만큼 바꾸기
  
```

[문제2] 보기와 같이 명령어를 작성했을 때 엔트리봇 오브젝트의 움직임을 작성해 봅시다.

```

시작하기 버튼을 클릭했을 때
  PID-13 조이스틱과 버튼 시작하기
  계속 반복하기
    만일 버튼 2 을(를) 클릭했는가? (이)라면
      2 초 동안 x: 10 y: 10 위치로 이동하기
  
```

[문제3] 엔트리봇 오브젝트가 앞뒤로 움직이면서 버튼을 누르면 모양이 바뀌도록 하려면 어떤 명령 블록모음들을 동시에 사용해야 하는지 표시해 봅시다.

시작하기 버튼을 클릭했을 때

치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기

치즈 스틱 0 : PID x1 의 범위 0 ~ 127 ~ 255 을(를) -100 ~ 0 ~ 100 소수점 없음 으로 경하기

계속 반복하기

만일 치즈 스틱 0 : PID x1 > 그리고 치즈 스틱 0 : PID x1 < 10 (이)라면

x 좌표를 0 만큼 바꾸기

아니면

만일 치즈 스틱 0 : PID x1 > 10 (이)라면

x 좌표를 10 만큼 바꾸기

아니면

x 좌표를 -10 만큼 바꾸기

()

시작하기 버튼을 클릭했을 때

치즈 스틱 0 : PID y1 의 범위 0 ~ 127 ~ 255 을(를) -100 ~ 0 ~ 100 소수점 없음 으로 경하기

계속 반복하기

만일 치즈 스틱 0 : PID y1 < 그리고 치즈 스틱 0 : PID y1 < 10 (이)라면

y 좌표를 0 만큼 바꾸기

아니면

만일 치즈 스틱 0 : PID y1 > 50 (이)라면

y 좌표를 10 만큼 바꾸기

아니면

y 좌표를 -10 만큼 바꾸기

()

시작하기 버튼을 클릭했을 때

치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기

계속 반복하기

만일 치즈 스틱 0 : PID 버튼 2 을(를) 클릭했는가 ? (이)라면

1 초 동안 x: 30 y: 50 만큼 움직이기

1 초 동안 x: 30 y: -50 만큼 움직이기

()

시작하기 버튼을 클릭했을 때

치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기

계속 반복하기

만일 치즈 스틱 0 : PID 버튼 2 을(를) 클릭했는가 ? (이)라면

다음 모양으로 바꾸기

()

시작하기 버튼을 클릭했을 때

치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기

치즈 스틱 0 : PID x1 의 범위 0 ~ 255 을(를) 0 ~ 100 소수점 없음 으로 경하기

계속 반복하기

만일 치즈 스틱 0 : PID x1 >= 50 (이)라면

x 좌표를 10 만큼 바꾸기

아니면

x 좌표를 -10 만큼 바꾸기

만일 벡 에 닿았는가? (이)라면

x: 0 y: 0 위치로 이동하기

()

시작하기 버튼을 클릭했을 때

치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기

치즈 스틱 0 : PID x1 의 범위 0 ~ 255 을(를) 0 ~ 100 소수점 없음 으로 경하기

계속 반복하기

만일 치즈 스틱 0 : PID x1 >= 50 (이)라면

x 좌표를 10 만큼 바꾸기

아니면

x 좌표를 -10 만큼 바꾸기

()

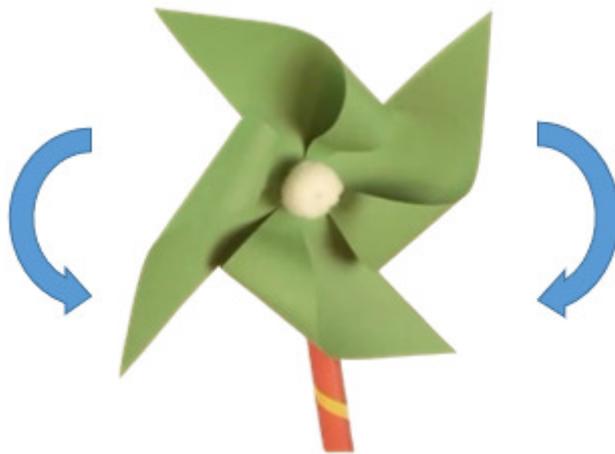
Basic Programing

5 방향조절 바람개비 만들기

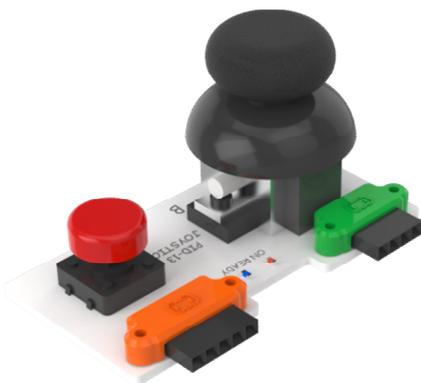
조이스틱과 서보모터, 색종이를 사용하여 바람개비를 조종할 수 있어요.



조이스틱과 서보모터를 사용하여 바람개비의 방향을 조종할 수 있어요.
색종이로 바람개비를 만들어 서보모터에 연결하고 조이스틱으로 조종해 봅시다.



준비를 해봅시다.



조이스틱



서보모터



치즈스틱 본체



사용할 블록을 알아봅시다.

PID-13 조이스틱과 버튼 ▶ 시작하기

x1 ▼ y1 ▼

치즈 스틱 0 : PID x1 ▼ 의 범위 0 ~ 255 윤(輪) 0 ~ 100 소수점 없음 ▼ 으로 정하기

x1 ▼ 의 범위 0 ~ 127 ~ 255 윤(輪) -100 ~ 0 ~ 100 (으)로 정하기

10 > 10

2 초 기다리기

치즈 스틱 0 : 서보 모터 Sa ▼ 의 각도를 0 도로 정하기

치즈 스틱 0 : 서보 모터 Sa ▼ 의 각도를 10 도만큼 바꾸기

치즈 스틱 0 : 서보 모터 Sa ▼ 전원 끄기

만일 참 (이)라면
아니면

조이스틱을 사용할 때 필요합니다.

조이스틱의 좌우, 상하값을 설정할 때 필요합니다.

조이스틱의 측정값을 0~100의 값으로 설정합니다.

조이스틱의 측정값을 세 단계의 값으로 설정합니다.

값의 크기 비교를 통해 명령을 설정합니다.

설정된 시간만큼 기다린 후 다음 블록을 실행합니다.

서보모터의 각도를 설정합니다.

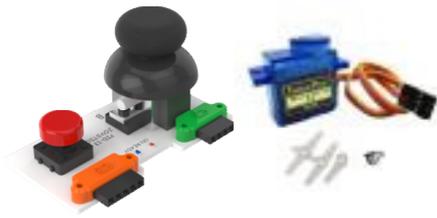
설정된 시간동안 서보모터의 각도를 변화시킵니다.

앞에서 실행중인 서보모터의 명령을 종료합니다.

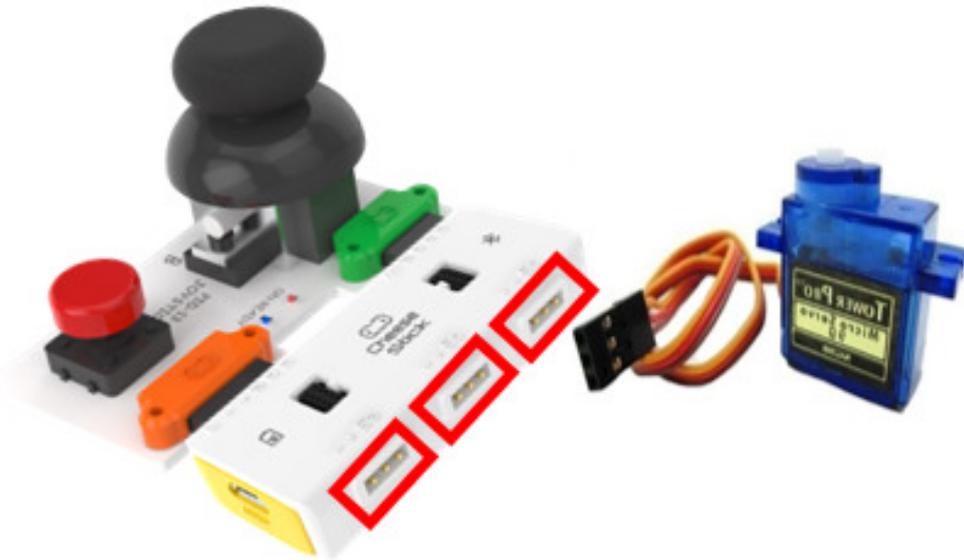
주어진 조건을 만족했을 때와 만족하지 않았을 때 서로 다른 명령을 실행합니다.



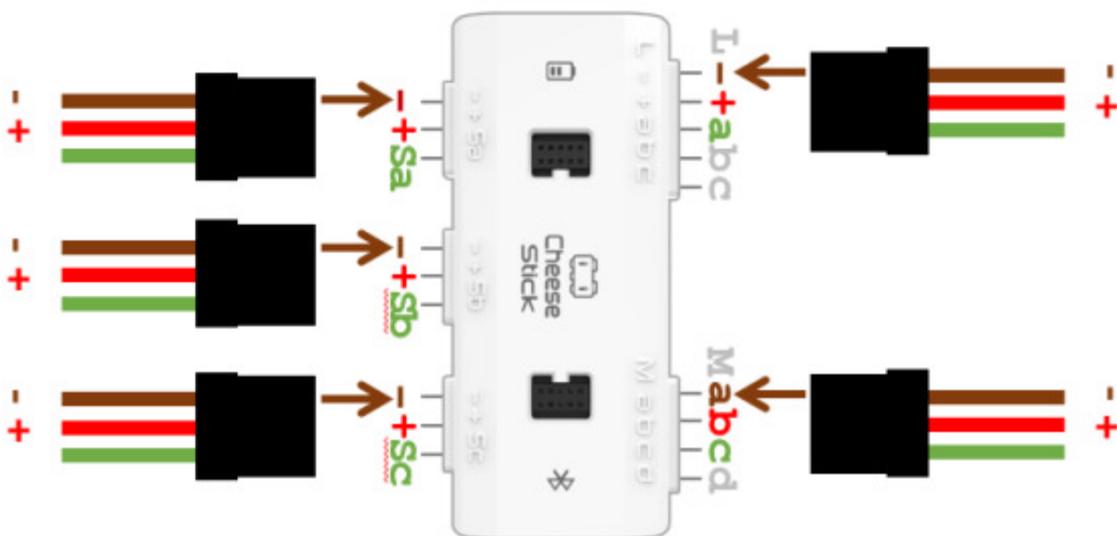
치즈스틱 본체에 조이스틱과 서보모터를 연결해 봅시다.



치즈스틱 본체에 조이스틱과 서보모터를 연결합니다. 이 때 조이스틱은 L포트와 M포트에 맞춰서 연결하고, 서보모터는 Sa, Sb, Sc 포트 중 하나를 선택하여 연결합니다.



서보모터는 기본 5개 연결이 가능합니다. 그리고 확장 키트 활용 시 최대 8개까지 치즈스틱 본체와 연결할 수 있습니다.



서보모터 연결 시 갈색 선이 '-' 쪽으로 연결될 수 있도록 합니다.

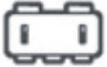


서보모터가 5번 동안 움직이도록 해봅시다.

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : 서보 모터 Sc▼ 의 각도를 0 도로 정하기
  1 초 기다리기
  5 번 반복하기
    치즈 스틱 0 : 서보 모터 Sc▼ 의 각도를 10 도만큼 바꾸기
    1 초 기다리기
  치즈 스틱 0 : 서보 모터 Sc▼ 전원 끄기

```



서보모터가 계속 움직이도록 해봅시다.

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : 서보 모터 Sc▼ 의 각도를 0 도로 정하기
  1 초 기다리기
  계속 반복하기
    치즈 스틱 0 : 서보 모터 Sc▼ 의 각도를 10 도만큼 바꾸기
    1 초 기다리기

```



서보모터는 0~180° 까지만 움직일 수 있습니다.



서보모터가 더 빠르게 움직이도록 해봅시다.

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : 서보 모터 Sc▼ 의 각도를 0 도로 정하기
  1 초 기다리기
  계속 반복하기
    치즈 스틱 0 : 서보 모터 Sc▼ 의 각도를 20 도만큼 바꾸기
    0.5 초 기다리기

```



서보모터의 속력을 빠르게 하려면 각도의 크기를 늘리거나 각도 변화 후 대기 시간을 짧게 설정하면 됩니다.



조이스틱을 사용해 서보모터가 한 방향으로 움직이도록 해봅시다.

▶ 시작하기 버튼을 클릭했을 때

```

    치즈 스틱 0 : PID-13 조이스틱과 버튼 ▶ 시작하기
    치즈 스틱 0 : PID x1 ▼ 의 범위 0 ~ 255 을(를) 0 ~ 100 소수점 없음 ▼ 으로 정하기
    계속 반복하기
    만일 < 치즈 스틱 0 : PID x1 ▼ > 60 (이)라면
    치즈 스틱 0 : 서보 모터 Sc ▼ 의 각도를 10 도만큼 바꾸기
    0.5 초 기다리기
  
```

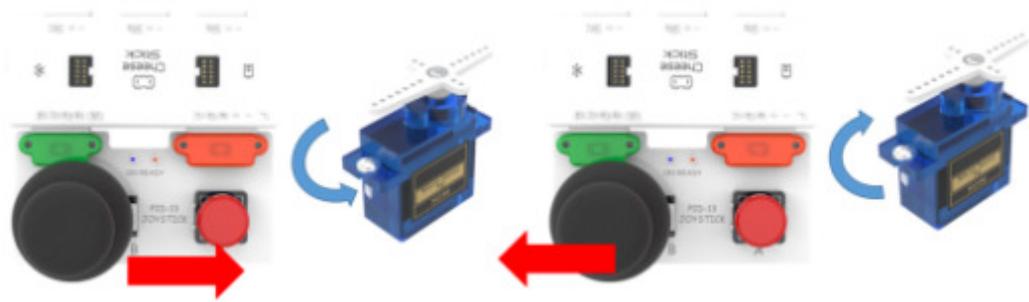


조이스틱을 사용해 서보모터가 양 방향으로 움직이도록 해봅시다.

▶ 시작하기 버튼을 클릭했을 때

```

    치즈 스틱 0 : PID-13 조이스틱과 버튼 ▶ 시작하기
    치즈 스틱 0 : PID x1 ▼ 의 범위 0 ~ 255 을(를) 0 ~ 100 소수점 없음 ▼ 으로 정하기
    계속 반복하기
    만일 < 치즈 스틱 0 : PID x1 ▼ > 60 (이)라면
    치즈 스틱 0 : 서보 모터 Sc ▼ 의 각도를 10 도만큼 바꾸기
    0.5 초 기다리기
    아니면
    치즈 스틱 0 : 서보 모터 Sc ▼ 의 각도를 -10 도만큼 바꾸기
    0.5 초 기다리기
  
```





조이스틱 사용 없이 서보모터가 양 방향으로 움직이도록 해봅시다.

▶ 시작하기 버튼을 클릭했을 때

계속 반복하기

치즈 스틱 0 : 서보 모터 Sc 의 각도를 0 도로 정하기

0.5 초 기다리기

치즈 스틱 0 : 서보 모터 Sc 의 각도를 10 도만큼 바꾸기

0.5 초 기다리기



조이스틱 버튼을 사용해 서보모터가 움직이도록 해봅시다.

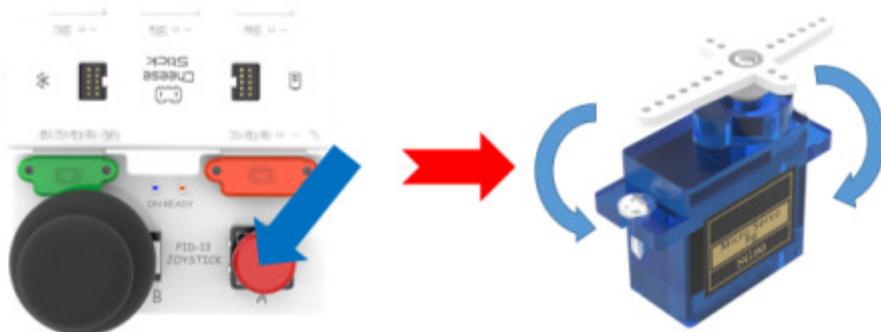
▶ 시작하기 버튼을 클릭했을 때

치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기

계속 반복하기

만일 치즈 스틱 0 : PID 버튼 2 을(를) 클릭했는가 ? (이)라면

치즈 스틱 0 : 서보 모터 Sc 의 각도를 10 도만큼 바꾸기





조이스틱 버튼을 눌렀을 때 움직이던 서보모터가 정지하도록 해봅시다.

```

시작하기 버튼을 클릭했을 때
  PID-13 조이스틱과 버튼 ▾ 시작하기
  계속 반복하기
    서보 모터 Sc ▾ 의 각도를 0 도로 정하기
    0.5 초 기다리기
    서보 모터 Sc ▾ 의 각도를 180 도만큼 바꾸기
    0.5 초 기다리기
  
```

```

시작하기 버튼을 클릭했을 때
  PID-13 조이스틱과 버튼 ▾ 시작하기
  계속 반복하기
    만일 버튼 2 ▾ 을(를) 클릭했는가? (이)라면
      서보 모터 Sc ▾ 전원 끄기
    
```



두 가지 명령을 동시에 수행할 때는

```

시작하기 버튼을 클릭했을 때
  
```

블록을 각각의 처음에 연결해 주면 실행이 가능합니다.



조이스틱을 사용하여 서보모터의 움직임 방향을 조정하고 버튼을 사용해서 서보모터의 움직임이 정지하도록 하여 바람개비에 연결해 조종해 봅시다.

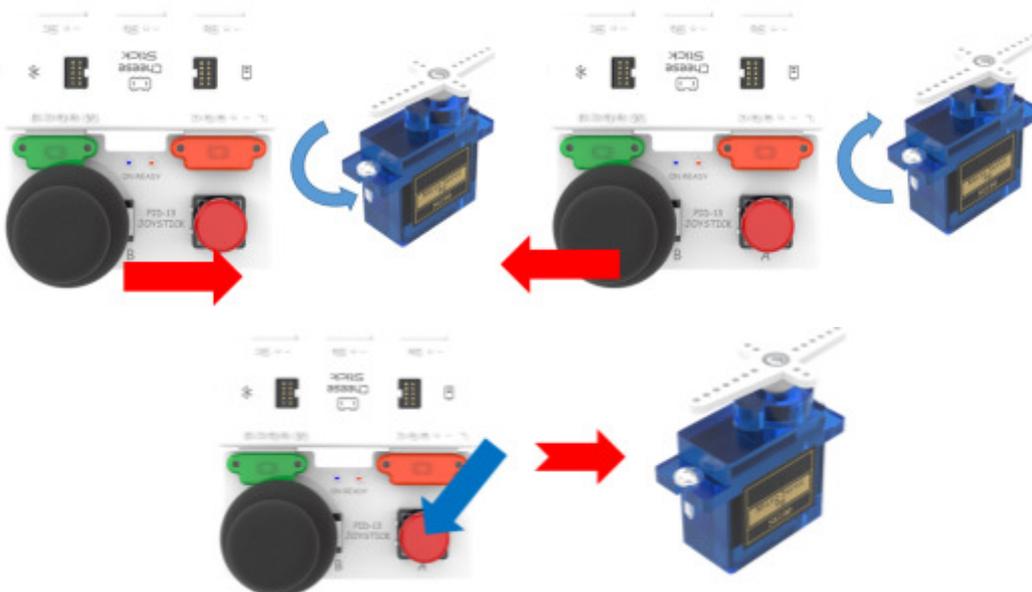
```

시작하기 버튼을 클릭했을 때
  PID-13 조이스틱과 버튼 시작하기
  x1의 범위 0 ~ 255 을(를) 0 ~ 100 (으)로 정하기
  서보 모터 Sc의 각도를 0 도로 정하기
  계속 반복하기
    만일 x1 > 60 (이)라면
      서보 모터 Sc의 각도를 10 도만큼 바꾸기
      0.5 초 기다리기
    아니면
      서보 모터 Sc의 각도를 -10 도만큼 바꾸기
      0.5 초 기다리기
  
```



```

시작하기 버튼을 클릭했을 때
  PID-13 조이스틱과 버튼 시작하기
  계속 반복하기
    만일 버튼 2 을(를) 클릭했는가? (이)라면
      서보 모터 Sc 전원 끄기
  
```





학습문제로 공부한 내용을 정리해 봅시다.

[문제1] 아래와 같이 명령어를 작성했을 때 서보모터가 어떻게 동작할지 작성해 봅시다.

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  치즈 스틱 0 : PID x1 의 범위 0 ~ 255 을(를) 0 ~ 100 소수점 없음 으로 정하기
  계속 반복하기
    만일 치즈 스틱 0 : PID x1 > 60 (이)라면
      치즈 스틱 0 : 서보 모터 Sc 의 각도를 10 도만큼 바꾸기
      0.5 초 기다리기
    아니면
      치즈 스틱 0 : 서보 모터 Sc 의 각도를 -10 도만큼 바꾸기
      0.5 초 기다리기
  
```

[문제2] 아래와 같이 명령어를 작성했을 때 서보모터가 어떻게 동작할지 작성해 봅시다.

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  치즈 스틱 0 : PID x1 의 범위 0 ~ 255 을(를) 0 ~ 100 소수점 없음 으로 정하기
  치즈 스틱 0 : 서보 모터 Sc 의 각도를 0 도로 정하기
  계속 반복하기
    만일 치즈 스틱 0 : PID x1 > 60 (이)라면
      치즈 스틱 0 : 서보 모터 Sc 의 각도를 40 도만큼 바꾸기
      0.5 초 기다리기
  
```

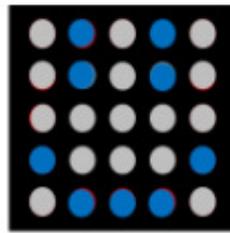
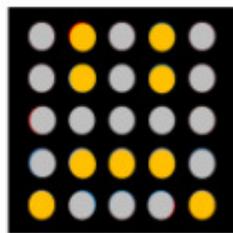
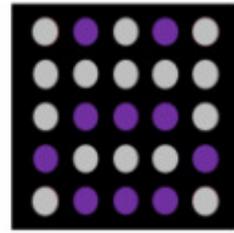
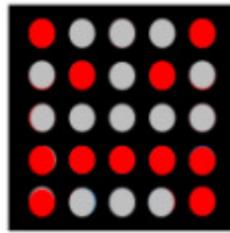
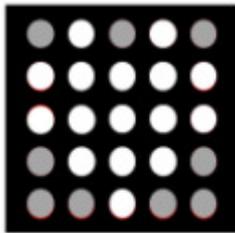

Basic Programming

6 감정표현 프로그램 만들기

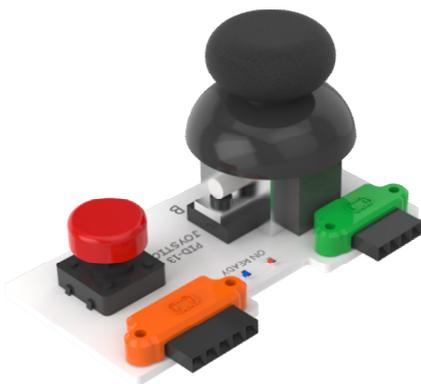
조이스틱을 사용하여 5x5 LED 매트릭스에 다양한 모양을 나타낼 수 있어요.



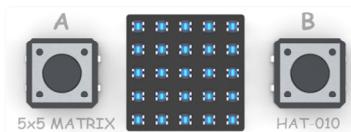
조이스틱을 사용하여 5x5 LED 매트릭스를 조종할 수 있어요. 5x5 LED 매트릭스에서 다양한 색이 출력되도록 조이스틱으로 조종해 봅시다.



준비를 해봅시다.



조이스틱



5x5 LED 매트릭스



치즈스틱 본체



사용할 블록을 알아봅시다.

치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기

치즈 스틱 0 : HAT-010 배경 x: 0 y: 0 빨간색 으로 켜기

치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기

치즈 스틱 0 : HAT-010 배경 빨간색 사각형 을(를) x: 0 y: 0 에 그리기

치즈 스틱 0 : HAT-010 배경 x: 0 y: 0 빨간색 으로 켜기

치즈 스틱 0 : PID x1 치즈 스틱 0 : PID y1

10 > 10

치즈 스틱 0 : PID x1 의 범위 0 ~ 255 을(를) 0 ~ 100 소수점 없음 으로 정하기

x1 의 범위 0 ~ 127 ~ 255 을(를) -100 ~ 0 ~ 100 (으)로 정하기

참 또는 참

버튼 1 을(를) 클릭했는가?

만일 참 (이)라면
아니면

조이스틱을 사용할 때 필요합니다.

5x5 LED 매트릭스의 LED를 제어합니다.

5x5 LED 매트릭스를 사용할 때 필요합니다.

5x5 LED 매트릭스에 여러 가지 그림을 표시합니다.

5x5 LED 매트릭스의 배경/모두를 지웁니다.

조이스틱의 좌우, 상하값을 설정할 때 필요합니다.

값의 크기 비교를 통해 명령을 설정합니다.

조이스틱의 측정값을 0~100의 값으로 설정합니다.

조이스틱의 측정값을 세 단계의 값으로 설정합니다.

하나 또는 둘 모두 참인 경우 '참'으로 판단합니다.

버튼을 눌렀을 때 명령을 실행합니다.

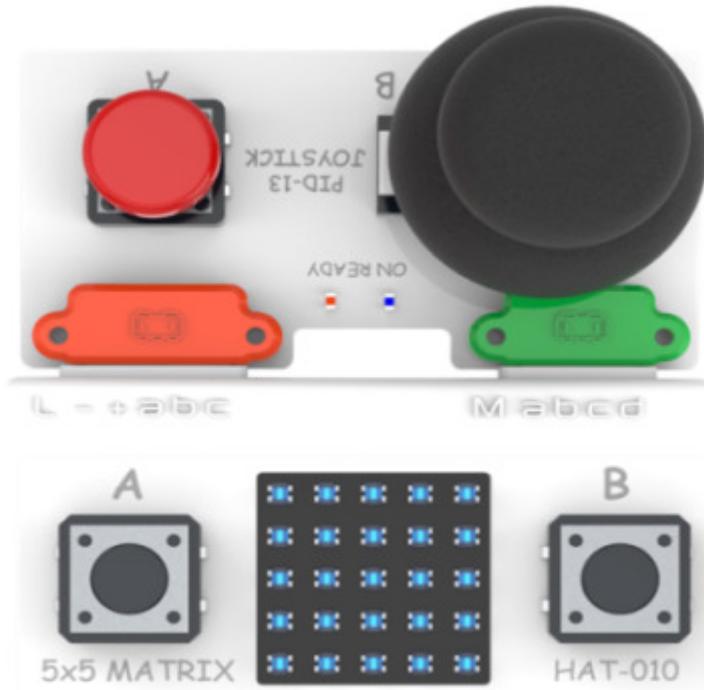
주어진 조건을 만족했을 때와 만족하지 않았을 때 서로 다른 명령을 실행합니다.



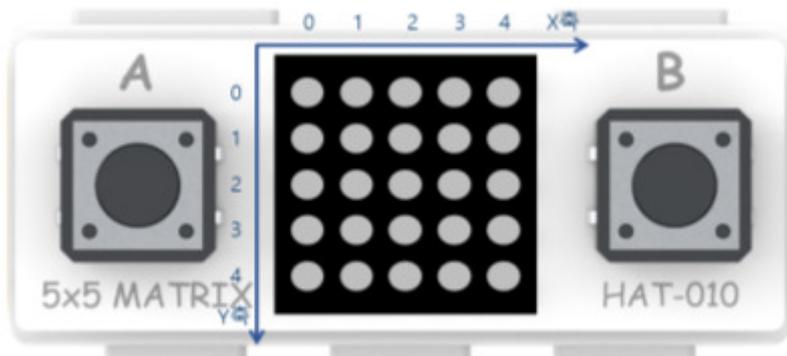
치즈스틱 본체에 조이스틱과 5x5 LED 매트릭스를 연결해 봅시다.



치즈스틱 본체에 조이스틱과 RGB LED를 연결합니다. 이 때 조이스틱은 L포트와 M포트에 맞춰서 연결하고, 5x5 LED 매트릭스는 치즈스틱 본체 위에 연결합니다.



5x5 LED 매트릭스를 연결할 때 치즈스틱 본체에 끝까지 들어갈 때까지 밀어 넣어야 합니다.



5x5 LED 매트릭스는 총 25개의 LED의 색상, 밝기 등을 제어할 수 있습니다.



5x5 LED 매트릭스의 LED에 다양한 도형이 나타나도록 해봅시다.

```

시작하기 버튼을 클릭했을 때
치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
치즈 스틱 0 : HAT-010 배경 빨간색 ▾ 삼각형 ▾ 울(를) x: 0 y: 0 에 그리기
  
```

```

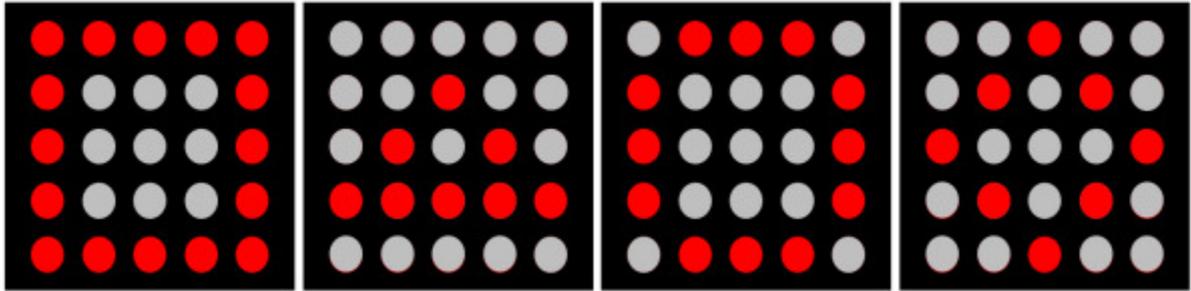
시작하기 버튼을 클릭했을 때
치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
치즈 스틱 0 : HAT-010 배경 빨간색 ▾ 사각형 ▾ 울(를) x: 0 y: 0 에 그리기
  
```

```

시작하기 버튼을 클릭했을 때
치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
치즈 스틱 0 : HAT-010 배경 빨간색 ▾ 원 ▾ 울(를) x: 0 y: 0 에 그리기
  
```

```

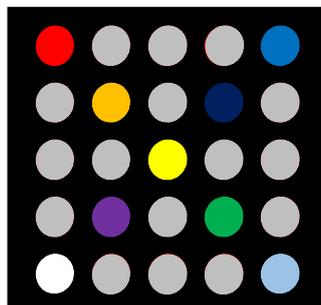
시작하기 버튼을 클릭했을 때
치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
치즈 스틱 0 : HAT-010 배경 빨간색 ▾ 다이아몬드 ▾ 울(를) x: 0 y: 0 에 그리기
  
```



5x5 LED 매트릭스에 여러 가지 색의 LED가 나타나도록 해봅시다.

```

시작하기 버튼을 클릭했을 때
치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
치즈 스틱 0 : HAT-010 배경 x: 0 y: 0 빨간색 ▾ 으로 켜기
치즈 스틱 0 : HAT-010 배경 x: 1 y: 1 주황색 ▾ 으로 켜기
치즈 스틱 0 : HAT-010 배경 x: 2 y: 2 노란색 ▾ 으로 켜기
치즈 스틱 0 : HAT-010 배경 x: 3 y: 3 초록색 ▾ 으로 켜기
치즈 스틱 0 : HAT-010 배경 x: 4 y: 4 하늘색 ▾ 으로 켜기
치즈 스틱 0 : HAT-010 배경 x: 4 y: 0 파란색 ▾ 으로 켜기
치즈 스틱 0 : HAT-010 배경 x: 3 y: 1 보라색 ▾ 으로 켜기
치즈 스틱 0 : HAT-010 배경 x: 1 y: 3 자주색 ▾ 으로 켜기
치즈 스틱 0 : HAT-010 배경 x: 0 y: 4 하얀색 ▾ 으로 켜기
  
```

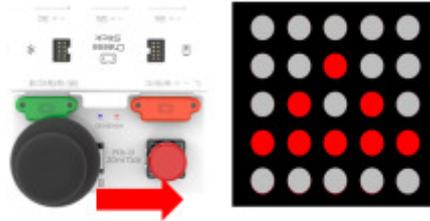




조이스틱을 사용해 5x5 LED 매트릭스에 삼각형이 나타나도록 해봅시다.

```

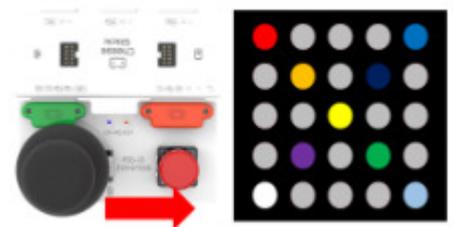
시작하기 버튼을 클릭했을 때
치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
치즈 스틱 0 : PID x1 의 범위 0 ~ 255 을(를) 0 ~ 100 소수점 없음 으로 정하기
계속 반복하기
만일 치즈 스틱 0 : PID x1 > 60 (이)라면
치즈 스틱 0 : HAT-010 배경 빨간색 삼각형 을(를) x: 0 y: 0 에 그리기
  
```



조이스틱을 사용해 5x5 LED 매트릭스에 여러 가지 색의 LED가 나타나도록 해봅시다.

```

시작하기 버튼을 클릭했을 때
치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
치즈 스틱 0 : PID x1 의 범위 0 ~ 255 을(를) 0 ~ 100 소수점 없음 으로 정하기
계속 반복하기
만일 치즈 스틱 0 : PID x1 > 60 (이)라면
치즈 스틱 0 : HAT-010 배경 x: 0 y: 0 빨간색 으로 켜기
치즈 스틱 0 : HAT-010 배경 x: 1 y: 1 주황색 으로 켜기
치즈 스틱 0 : HAT-010 배경 x: 2 y: 2 노란색 으로 켜기
치즈 스틱 0 : HAT-010 배경 x: 3 y: 3 초록색 으로 켜기
치즈 스틱 0 : HAT-010 배경 x: 4 y: 4 하늘색 으로 켜기
치즈 스틱 0 : HAT-010 배경 x: 4 y: 0 파란색 으로 켜기
치즈 스틱 0 : HAT-010 배경 x: 3 y: 1 보라색 으로 켜기
치즈 스틱 0 : HAT-010 배경 x: 1 y: 3 자주색 으로 켜기
치즈 스틱 0 : HAT-010 배경 x: 0 y: 4 하얀색 으로 켜기
  
```



조이스틱을 사용해서 5x5 LED 매트릭스에 명령을 주기 위해서는

```
치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
```

```
치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
```

이 두 블록을 모두 사용하여야 합니다.

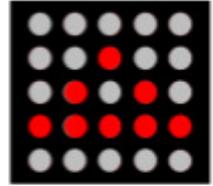


조이스틱의 버튼과 5x5 LED 매트릭스의 버튼을 사용해서 다양한 LED의 모양이 나타나도록 해봅시다.

▶ 시작하기 버튼을 클릭했을 때

```

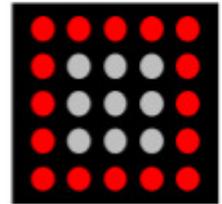
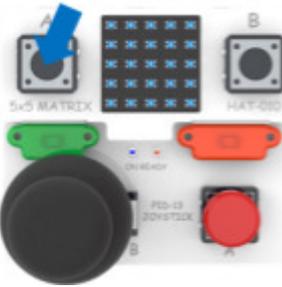
치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
계속 반복하기
만일 치즈 스틱 0 : PID 버튼 2 울(음) 클릭했는가? (이)라면
치즈 스틱 0 : HAT-010 배경 빨간색 삼각형 울(음) x: 0 y: 0 에 그리기
  
```



▶ 시작하기 버튼을 클릭했을 때

```

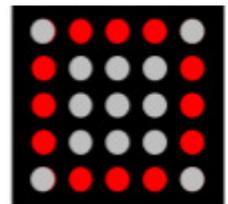
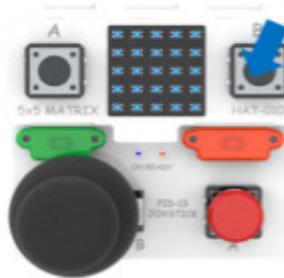
치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
계속 반복하기
만일 치즈 스틱 0 : HAT-010 버튼 A 울(음) 클릭했는가? (이)라면
치즈 스틱 0 : HAT-010 배경 빨간색 사각형 울(음) x: 0 y: 0 에 그리기
  
```



▶ 시작하기 버튼을 클릭했을 때

```

치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
계속 반복하기
만일 치즈 스틱 0 : HAT-010 버튼 B 울(음) 클릭했는가? (이)라면
치즈 스틱 0 : HAT-010 배경 빨간색 원 울(음) x: 0 y: 0 에 그리기
  
```

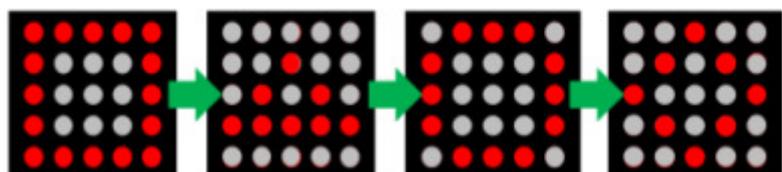
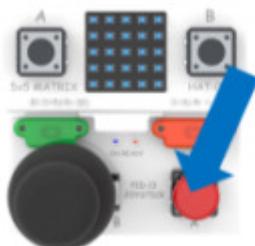


조이스틱 버튼을 사용해 5x5 LED 매트릭스가 다양한 모양으로 순서대로 변하도록 해봅시다.

▶ 시작하기 버튼을 클릭했을 때

```

치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
계속 반복하기
만일 치즈 스틱 0 : PID 버튼 2 울(음) 클릭했는가? (이)라면
치즈 스틱 0 : HAT-010 배경 빨간색 사각형 울(음) x: 0 y: 0 에 그리기
1 초 기다리기
치즈 스틱 0 : HAT-010 배경 빨간색 삼각형 울(음) x: 0 y: 0 에 그리기
1 초 기다리기
치즈 스틱 0 : HAT-010 배경 빨간색 다이아몬드 울(음) x: 0 y: 0 에 그리기
1 초 기다리기
치즈 스틱 0 : HAT-010 배경 빨간색 원 울(음) x: 0 y: 0 에 그리기
1 초 기다리기
  
```





조이스틱으로 5x5 LED 매트릭스에 3가지 감정을 나타내는 프로그램을 만들어 봅시다.

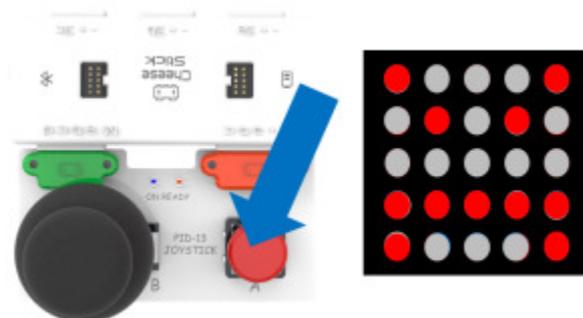
```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  치즈 스틱 0 : PID x1 의 범위 0 ~ 255 을(를) 0 ~ 100 소수점 없음 으로 정하기
  계속 반복하기
    만일 < 치즈 스틱 0 : PID x1 > > 60 (이)라면
      치즈 스틱 0 : HAT-010 배경 파란색 > 좋음 > 을(를) x: 0 y: 0 에 그리기
      1 초 기다리기
    아니면
      만일 < 치즈 스틱 0 : PID x1 > < 40 (이)라면
        치즈 스틱 0 : HAT-010 배경 주황색 > 좋음 > 을(를) x: 0 y: 0 에 그리기
        1 초 기다리기
      아니면
        치즈 스틱 0 : HAT-010 배경 > 지우기
  
```



```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  만일 < 치즈 스틱 0 : PID 버튼 2 > 을(를) 클릭했는가 > ? (이)라면
    치즈 스틱 0 : HAT-010 배경 빨간색 > 화남 > 을(를) x: 0 y: 0 에 그리기
    1 초 기다리기
  치즈 스틱 0 : HAT-010 배경 > 지우기
  
```

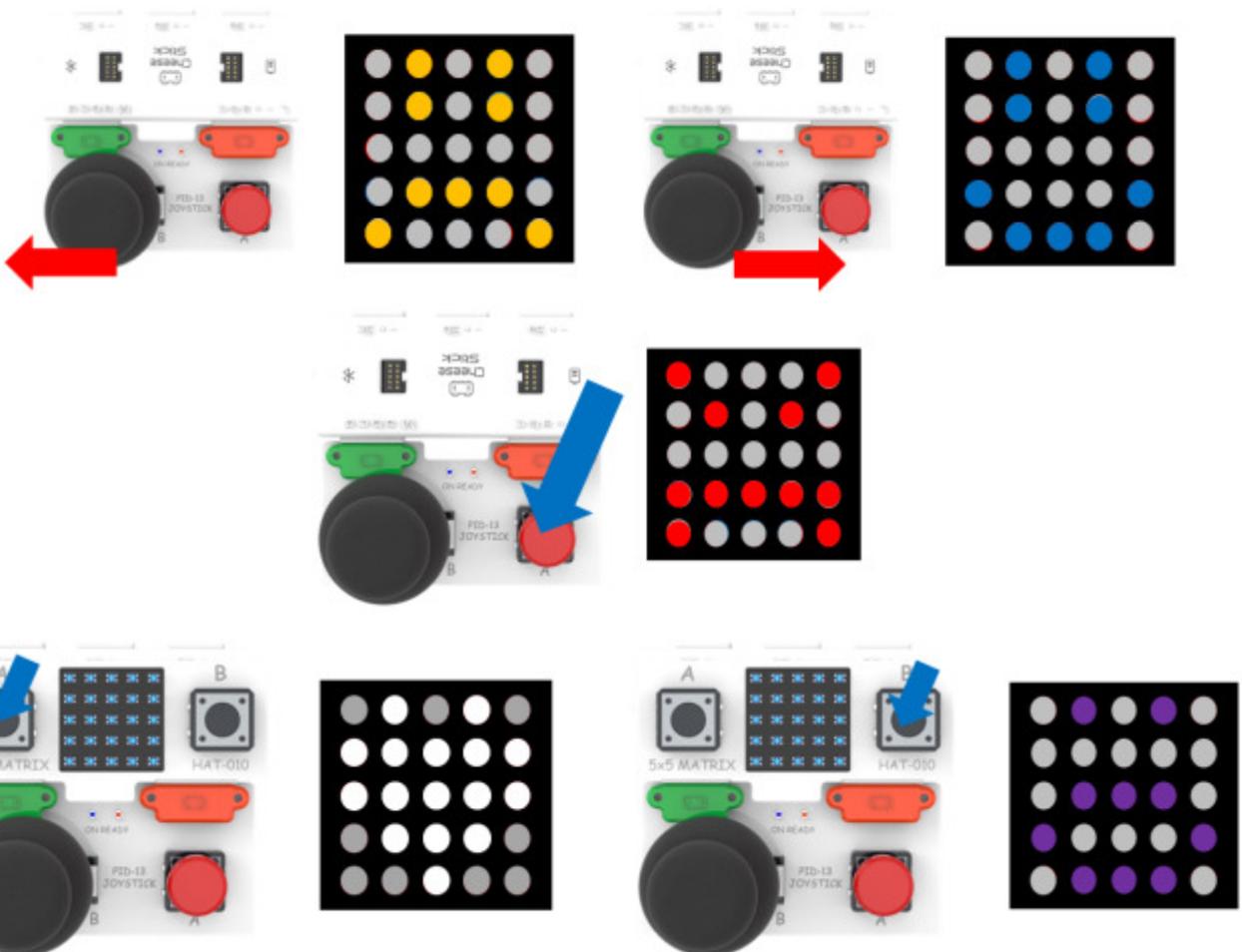




조이스틱과 5x5 LED 매트릭스의 버튼을 활용해 5가지 감정을 나타내는 프로그램을 만들기 위해 추가 명령어를 만들어 봅시다.

시작하기 버튼을 클릭했을 때

- 치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
- 치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
- 계속 반복하기
 - 만일 치즈 스틱 0 : HAT-010 버튼 A 를 클릭했는가? (이)라면
 - 치즈 스틱 0 : HAT-010 배경 하얀색 하트 을(를) x: 0 y: 0 에 그리기
 - 1 초 기다리기
 - 치즈 스틱 0 : HAT-010 배경 지우기
 - 만일 치즈 스틱 0 : HAT-010 버튼 B 를 클릭했는가? (이)라면
 - 치즈 스틱 0 : HAT-010 배경 보라색 입 열기 을(를) x: 0 y: 0 에 그리기
 - 1 초 기다리기
 - 치즈 스틱 0 : HAT-010 배경 지우기



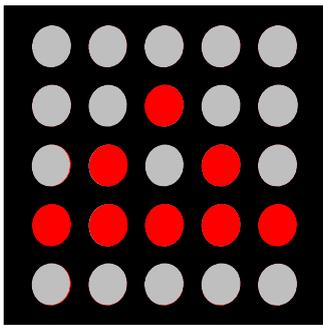


학습문제로 공부한 내용을 정리해 봅시다.

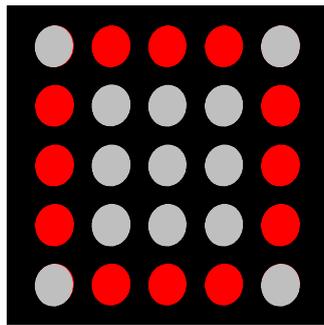
[문제1] 아래와 같이 명령어를 작성했을 때 어떤 모양이 5x5 LED 매트릭스에 나타나는지 확인해 봅시다.

```

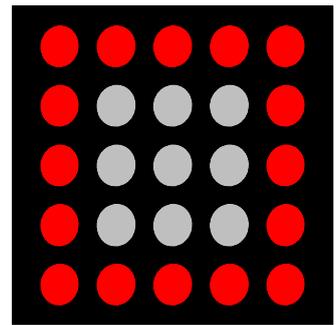
시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  치즈 스틱 0 : PID x1 의 범위 0 ~ 255 을(를) 0 ~ 100 소수점 없음 으로 정하기
  계속 반복하기
    만일 치즈 스틱 0 : PID x1 > 60 (이)라면
      치즈 스틱 0 : HAT-010 배경 빨간색 삼각형 을(를) x: 0 y: 0 에 그리기
  
```



()



()

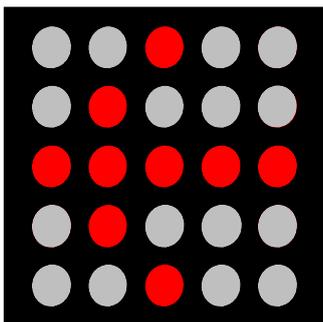


()

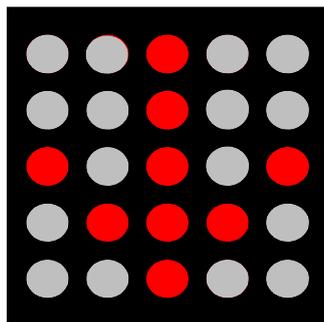
[문제2] 아래와 같이 명령어를 작성했을 때 버튼을 누르면 어떤 모양이 5x5 LED 매트릭스에 나타나는지 확인해 봅시다.

```

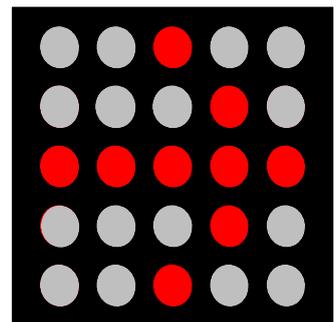
시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  계속 반복하기
    만일 치즈 스틱 0 : PID 버튼 2 을(를) 클릭했는가 ? (이)라면
      치즈 스틱 0 : HAT-010 배경 빨간색 왼쪽 화살표 을(를) x: 0 y: 0 에 그리기
  
```



()



()



()

[문제3] 조이스틱과 5x5 LED 매트릭스의 버튼을 활용해 5가지 감정을 나타내는 프로그램을 만들기 위해서 어떤 명령 블록을 동시에 사용해야 하는지 표시해 봅시다.

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  치즈 스틱 0 : PID x1 의 범위 0 ~ 255 을(를) 0 ~ 100 소수점 없음 으로 경하기
  계속 반복하기
    만일 치즈 스틱 0 : PID x1 > 60 (이)라면
      치즈 스틱 0 : HAT-010 배경 파란색 줌 음(를) x: 0 y: 0 에 그리기
      1 초 기다리기
    아니면
      만일 치즈 스틱 0 : PID x1 < 40 (이)라면
        치즈 스틱 0 : HAT-010 배경 주황색 실음 음(를) x: 0 y: 0 에 그리기
        1 초 기다리기
      아니면
        치즈 스틱 0 : HAT-010 배경 지우기
  
```

()

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  치즈 스틱 0 : PID x1 의 범위 0 ~ 255 을(를) 0 ~ 100 소수점 없음 으로 경하기
  계속 반복하기
    만일 치즈 스틱 0 : PID x1 > 60 (이)라면
      치즈 스틱 0 : HAT-010 배경 x: 0 y: 0 빨간색 으로 켜기
      치즈 스틱 0 : HAT-010 배경 x: 1 y: 1 주황색 으로 켜기
      치즈 스틱 0 : HAT-010 배경 x: 2 y: 2 노란색 으로 켜기
      치즈 스틱 0 : HAT-010 배경 x: 3 y: 3 초록색 으로 켜기
      치즈 스틱 0 : HAT-010 배경 x: 4 y: 4 하늘색 으로 켜기
      치즈 스틱 0 : HAT-010 배경 x: 4 y: 0 파란색 으로 켜기
      치즈 스틱 0 : HAT-010 배경 x: 3 y: 1 보라색 으로 켜기
      치즈 스틱 0 : HAT-010 배경 x: 1 y: 3 자주색 으로 켜기
      치즈 스틱 0 : HAT-010 배경 x: 0 y: 4 하얀색 으로 켜기
  
```

()

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  계속 반복하기
    만일 치즈 스틱 0 : HAT-010 버튼 A 를 클릭했는가 ? (이)라면
      치즈 스틱 0 : HAT-010 배경 하얀색 하트 음(를) x: 0 y: 0 에 그리기
      1 초 기다리기
      치즈 스틱 0 : HAT-010 배경 지우기
    만일 치즈 스틱 0 : HAT-010 버튼 B 를 클릭했는가 ? (이)라면
      치즈 스틱 0 : HAT-010 배경 보라색 입 열기 음(를) x: 0 y: 0 에 그리기
      1 초 기다리기
      치즈 스틱 0 : HAT-010 배경 지우기
  
```

()

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  계속 반복하기
    만일 치즈 스틱 0 : PID 버튼 2 을(를) 클릭했는가 ? (이)라면
      치즈 스틱 0 : HAT-010 배경 빨간색 사각형 음(를) x: 0 y: 0 에 그리기
      1 초 기다리기
      치즈 스틱 0 : HAT-010 배경 빨간색 삼각형 음(를) x: 0 y: 0 에 그리기
      1 초 기다리기
      치즈 스틱 0 : HAT-010 배경 빨간색 다이아몬드 음(를) x: 0 y: 0 에 그리기
      1 초 기다리기
      치즈 스틱 0 : HAT-010 배경 빨간색 원 음(를) x: 0 y: 0 에 그리기
      1 초 기다리기
  
```

()

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  만일 치즈 스틱 0 : PID 버튼 2 을(를) 클릭했는가 ? (이)라면
    치즈 스틱 0 : HAT-010 배경 빨간색 화남 음(를) x: 0 y: 0 에 그리기
    1 초 기다리기
    치즈 스틱 0 : HAT-010 배경 지우기
  
```

()

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기
  치즈 스틱 0 : HAT-010 배경 x: 0 y: 0 빨간색 으로 켜기
  치즈 스틱 0 : HAT-010 배경 x: 1 y: 1 주황색 으로 켜기
  치즈 스틱 0 : HAT-010 배경 x: 2 y: 2 노란색 으로 켜기
  치즈 스틱 0 : HAT-010 배경 x: 3 y: 3 초록색 으로 켜기
  치즈 스틱 0 : HAT-010 배경 x: 4 y: 4 하늘색 으로 켜기
  치즈 스틱 0 : HAT-010 배경 x: 4 y: 0 파란색 으로 켜기
  치즈 스틱 0 : HAT-010 배경 x: 3 y: 1 보라색 으로 켜기
  치즈 스틱 0 : HAT-010 배경 x: 1 y: 3 자주색 으로 켜기
  치즈 스틱 0 : HAT-010 배경 x: 0 y: 4 하얀색 으로 켜기
  
```

()

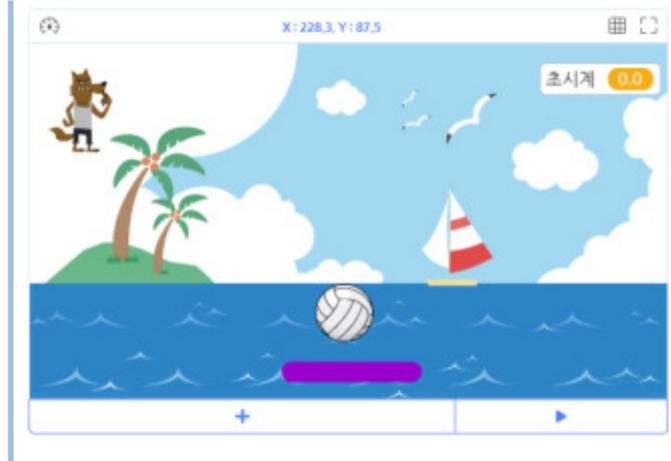
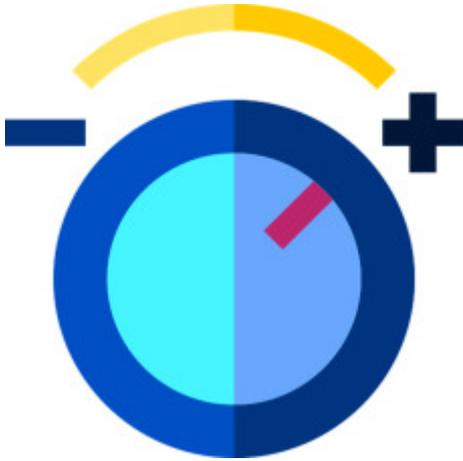
Basic Programing

7 공 튀기기 게임 만들기

치즈스틱과 가변저항을 이용하여 간단한 공 튀기기 게임을 만들 수 있어요.



가변저항(포텐셔미터, Potentiometer)은 센서의 한 종류로서 손잡이(노브)를 돌림으로서 입력값이 변화됩니다. 이번 챕터에서는 가변저항에 대해 알아보고 이를 이용한 간단한 공 튀기기 게임을 만들어 봅시다.



준비를 해봅시다.



가변저항



치즈스틱 본체



사용할 블록을 알아보시다.

치즈 스틱 0 : 포트 Sa 를 메이키 입력으로 경하기

치즈 스틱 0 : 입력 Sa

치즈 스틱 0 : 입력 Sa 의 범위 0 ~ 255 음(음) 0 ~ 100 소수점 없음 으로 경하기

초시계 시작하기

초시계 값

모양 보이기

모양 숨기기

안녕! 음(글) 말하기

자신 의 복제본 만들기

복제본이 처음 생성되었을때

이 복제본 삭제하기

이동 방향으로 10 만큼 움직이기

이동 방향을 90° 만큼 회전하기

x: 0 y: 0 위치로 이동하기

화면 끝에 닿으면 튕기기

0 부터 10 사이의 무작위 수

마우스포인터 에 닿았는가?

10 > 10

포트 Sa, Sb, Sc에 연결되는
입력·출력 장치를 정할 때 필요합니다.

포트 Sa에 연결된 입력·출력
장치의 값이 저장되어 있습니다.

포트 Sa에 연결된 입력·출력
장치의 값을 원하는 범위로 변경합니다.

엔트리 초시계를 시작/종료합니다.

초시계의 현재값이 저장되어 있습니다.

오브젝트를 보이거나 숨깁니다.

오브젝트가 말풍선으로 말을 하게 합니다.

오브젝트의 복사본과 관련된
블록입니다.

오브젝트를 이동방향으로
움직입니다.

오브젝트의 이동방향을
회전시킵니다.

오브젝트를 특정 좌표로
이동시킵니다.

오브젝트가 화면 끝에 닿으면
튕기도록 합니다.

원하는 범위에서 무작위 수를
사용합니다.

벽이나 오브젝트 등에 닿았는지
확인합니다.

값의 크기 비교를 통해 명령을
설정합니다.



가변저항을 치즈스틱 본체와 연결하여 봅시다.



가변저항을 치즈스틱 본체와 연결합니다. Sa, Sb, Sc 포트 중 한 곳에 꽂으면 됩니다. L 포트에 꽂기를 원한다면 가변저항을 뒤집고 포트의 왼쪽으로 붙여서 ‘+a’ 순서대로 꽂아야 합니다.

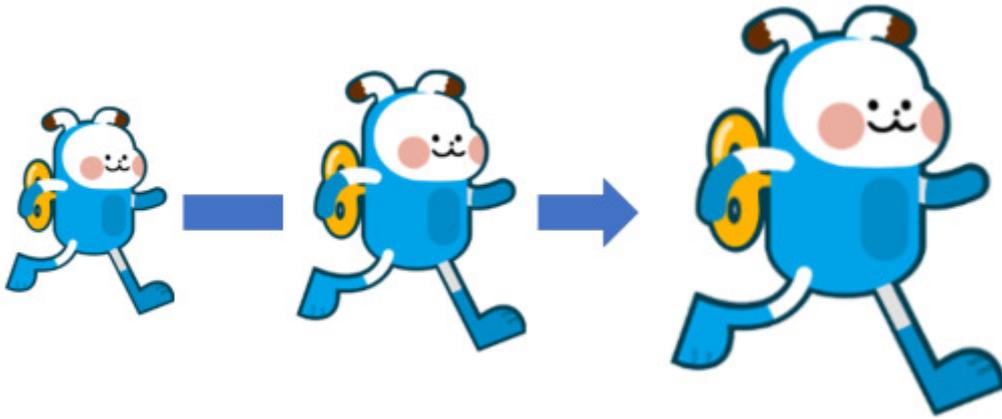


가변저항은 손잡이(노브)를 돌려서 입력값을 바꿀 수 있습니다. 가변저항의 값을 알려주는 엔트리 프로그램을 만들고 실행해 봅시다. 가변저항을 왼쪽, 오른쪽으로 돌려가며 값이 변하는 것을 관찰합니다.



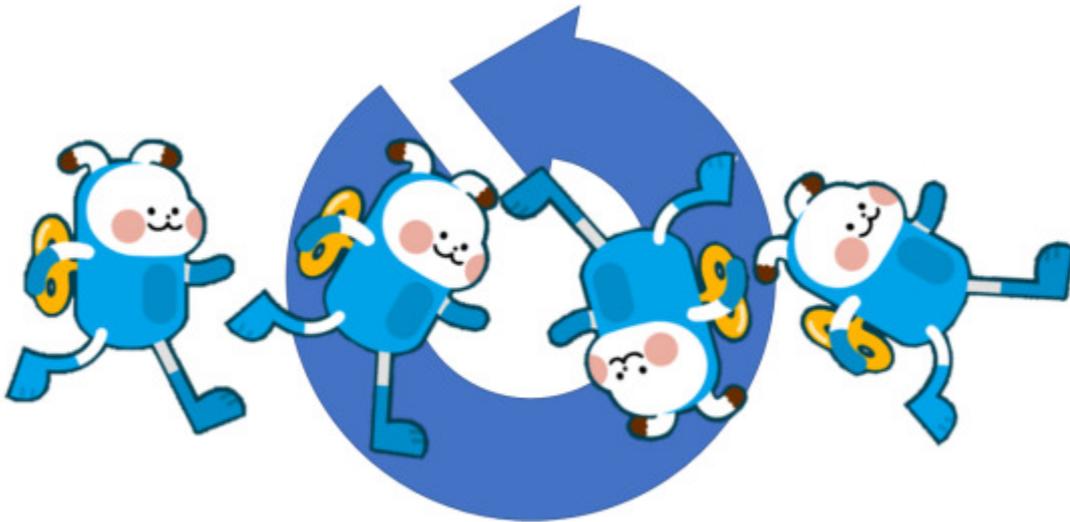
가변저항은 아날로그 입력장치입니다. 따라서 가변저항을 이용할 때는 반드시 **치즈 스틱 0 : 포트 Sa 를 아날로그 입력으로 정하기** 블록을 사용해야 합니다.

[연습과제1]



가변저항을 이용하여 엔트리봇 확대·축소시키기

[연습과제2]



가변저항을 이용하여 엔트리봇 회전시키기

확인하기

[연습과제1]

시작하기 버튼을 클릭했을 때

치즈 스틱 0 : 포트 Sa ▼ 를 아날로그 ▼ 입력으로 정하기

계속 반복하기

크기를 치즈 스틱 0 : 입력 Sa ▼ (으)로 정하기

[연습과제2]

시작하기 버튼을 클릭했을 때

치즈 스틱 0 : 포트 Sa ▼ 를 아날로그 ▼ 입력으로 정하기

계속 반복하기

방향을 치즈 스틱 0 : 입력 Sa ▼ (으)로 정하기



가변저항을 이용한 공 튀기기 게임을 만들어 봅시다. 우선 가변저항의 손잡이를 돌림에 따라 움직이는 막대를 만들어 봅시다. ‘판’ 오브젝트를 추가하고 명령 코드를 만듭니다.



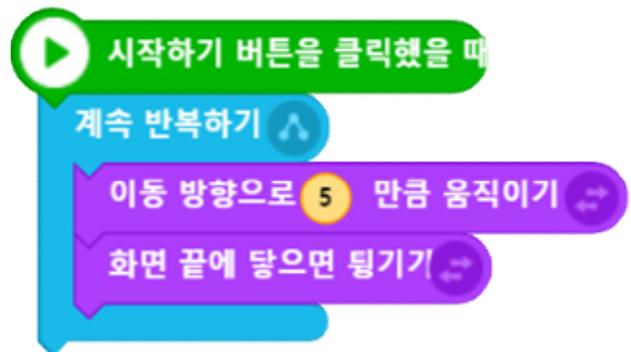
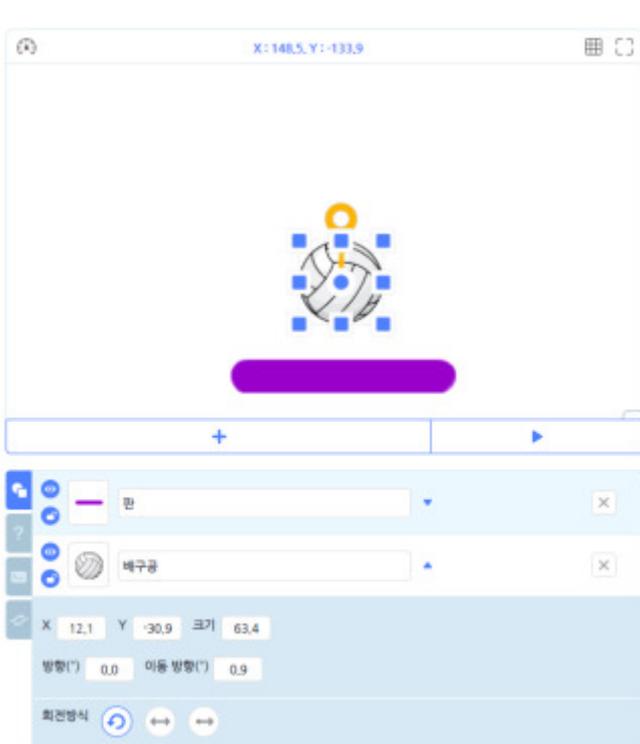
가변저항은 0~1023 사이의 값을 가집니다. 그런데 공을 튀기기 위한 판이 움직이는 X좌표의 범위는 -240~240입니다. 따라서 가변저항의 값을 X좌표의 범위로 바꾸어주는 블록이 필요합니다. 이 때 사용하는 블록이



입니다.



공 오브젝트를 추가하고, 오브젝트의 이동방향에 따라 움직이다가 화면 가장자리에 닿으면 튕기는 공을 만들어 봅시다.





공이 위아래로만 튕긴다면 재미가 없겠죠? 위아래로만 튕기고 있는 공이 '판' 오브젝트와 닿으면 아무 방향으로 튕기도록 만들어 봅시다.

```

시작하기 버튼을 클릭했을 때
  계속 반복하기
    이동 방향으로 5 만큼 움직이기
    화면 끝에 닿으면 튕기기
    만일 판에 닿았는가? (이)라면
      방향을 -50 부터 50 사이의 무작위 수 (으)로 정하기
  
```



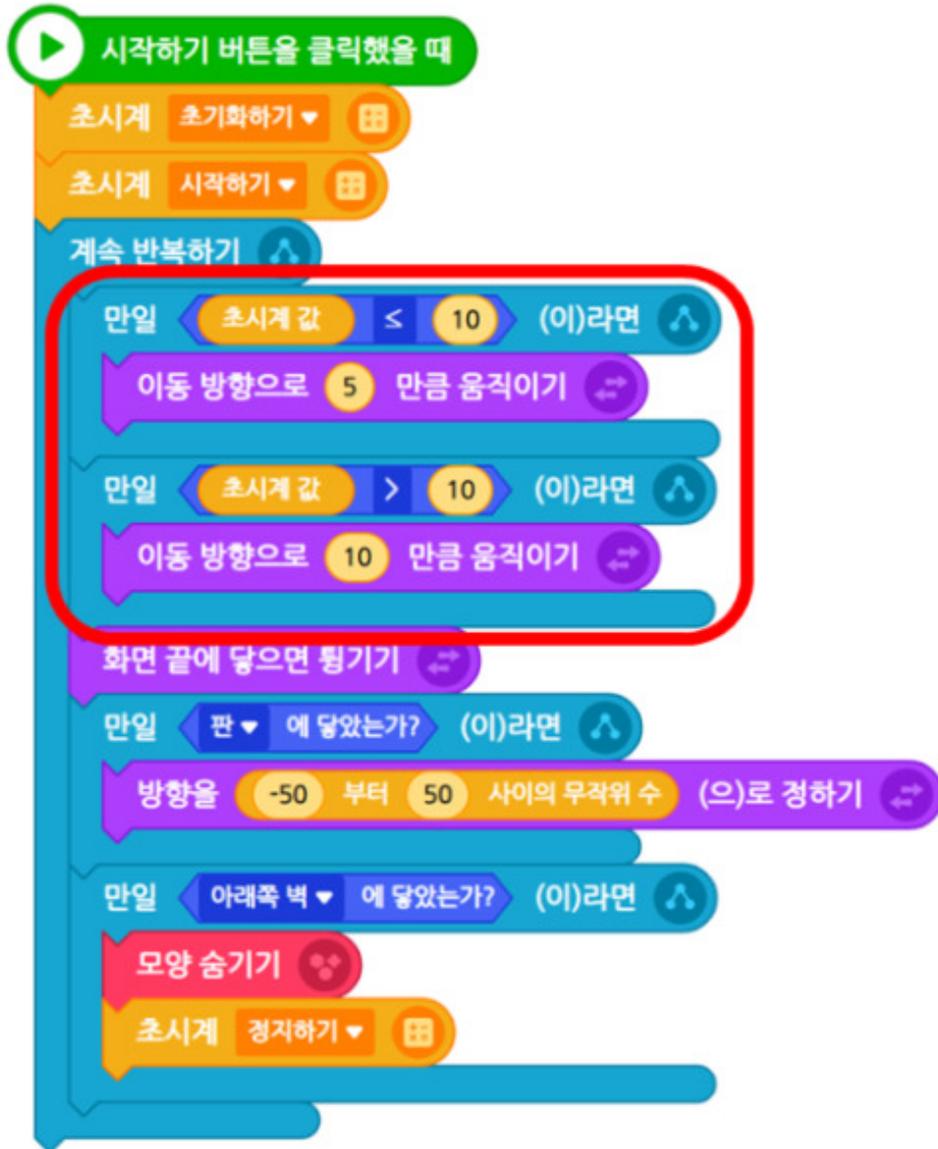
공을 바닥에 떨어트리지 않고 얼마나 오래 튕기는지를 측정해 봅시다. 초시계로 공을 튕긴 시간을 측정하고, 바닥에 닿으면 초시계가 멈추고 공이 사라짐으로써 게임이 끝나도록 만들어 봅시다.

```

시작하기 버튼을 클릭했을 때
  초시계 초기화하기
  초시계 시작하기
  계속 반복하기
    이동 방향으로 5 만큼 움직이기
    화면 끝에 닿으면 튕기기
    만일 판에 닿았는가? (이)라면
      방향을 -50 부터 50 사이의 무작위 수 (으)로 정하기
    만일 아래쪽 벽에 닿았는가? (이)라면
      모양 숨기기
      초시계 정지하기
  
```

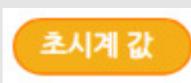


게임이 너무 쉽다면 난이도를 바꾸어 봅시다. 게임이 시작된 후 0~10초 동안은 천천히 공이 움직이고 10초가 지나면 공의 속도를 높여봅시다.



초시계의 값을 이용하여 공의 속도를 바꿀 수 있습니다. 초시계의 값이 10이하일 때 이동 방향으로 5만큼 움직이다가, 10초가 넘으면 10만큼 움직이게 함으로써 공의 속도가 2배로 빨라지게 만들 수 있습니다.

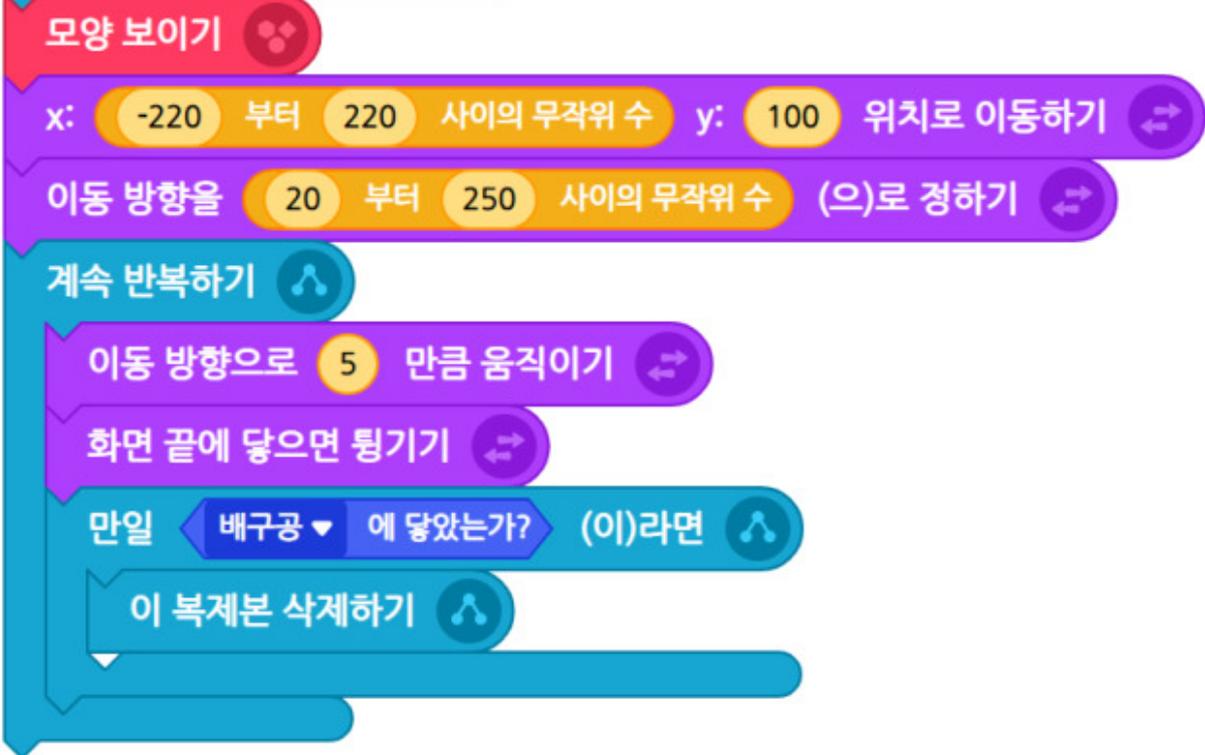
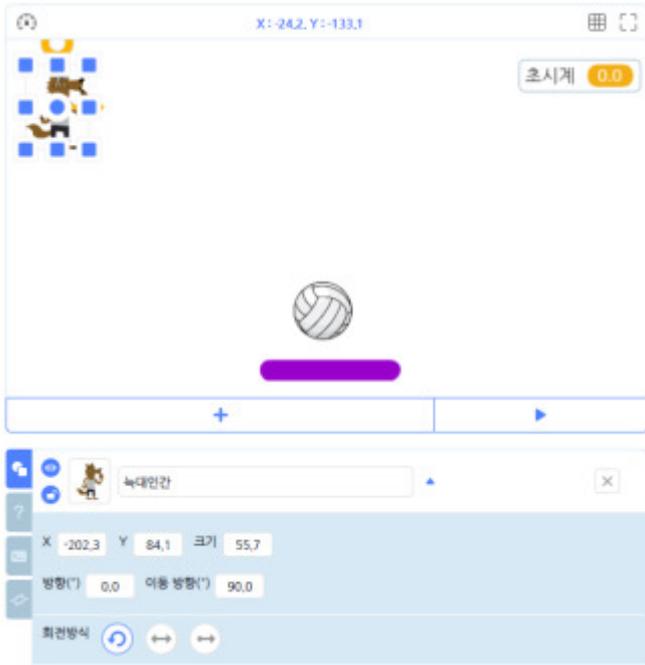
기준이 되는 시간이나 공의 속도를 자유롭게 바꾸어 봅시다.

 블록은  구리미에서 찾을 수 있고,

 블록은  구리미에서 찾을 수 있습니다.



단순히 공만 튀기는 것이 아니라 튀기는 공을 이용하여 몬스터를 잡는다면 더 재미있을 것 같습니다. '늑대인간' 오브젝트를 추가하고, 늑대인간이 공에 맞으면 사라지는 프로그램을 만들어 봅시다.



몬스터를 등장시킬 때는 '복제하기' 기능을 사용하여 몬스터를 나타나게 합니다. 각각의 명령 블록이 의미하는 것이 무엇인지 잘 생각해 보세요!



학습문제로 공부한 내용을 정리해 봅시다.

[문제1] 보기와 같이 가변저항을 연결하고 명령어를 작성했을 때 엔트리봇이 말하는 것은 무엇을 의미하는지 골라봅시다.



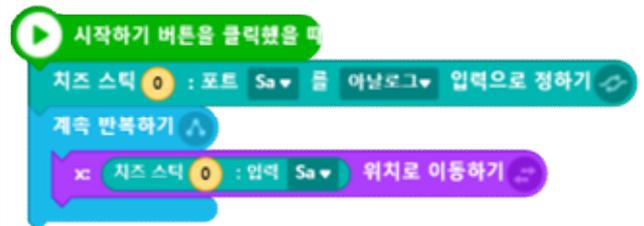
① 가변저항의 값

② 엔트리봇의 회전각도

③ 엔트리봇의 크기

④ 치즈스틱의 배터리

[문제2] 보기와 같이 가변저항으로 움직이는 판의 명령어를 작성했을 때 예상되는 문제점은 무엇인지 골라봅시다.



① 판이 움직이지 않는다.

② 판이 위아래로 움직인다.

③ 판이 실행화면 밖으로 나가버릴 수 있다.

④ 판이 천천히 움직인다.

[문제3] 가변저항의 값을 원하는 범위의 값으로 바꾸기 위한 블록으로 올바른 것을 골라봅시다.

<p>치즈 스틱 0 : 입력 Sa ▼ 의 범위 0 ~ 255 을(를) 0 ~ 100 소수점 앞을(를) 으로 정하기</p> <p>()</p> <p>치즈 스틱 0 : 입력 Sa ▼</p> <p>()</p>	<p>치즈 스틱 0 : 포트 Sa ▼ 를 메이키▼ 입력으로 정하기</p> <p>()</p> <p>치즈 스틱 0 : 버튼 Sa ▼ 를 클릭했는가▼ ?</p> <p>()</p>
--	--

[문제4] 공 튀기기 게임을 만들 때 공의 속도를 조절하기 위해서는 어떤 부분을 고쳐야 하는지 생각해 봅시다.

Basic Programing

8 밝기 조절 가로등 만들기

가변저항과 LED를 이용하여 밝기 조절이 되는 가로등을 만들 수 있어요.



가변저항과 LED를 이용하여 밝기가 조절되는 가로등을 만들어 봅시다.



준비를 해봅시다.



가변저항



LED



치즈스틱 본체



사용할 블록을 알아보시다.



포트 Sa, Sb, Sc에 연결되는 입력·출력 장치를 정할 때 필요합니다.



포트 Sa에 연결된 입력·출력 장치의 값이 저장되어 있습니다.



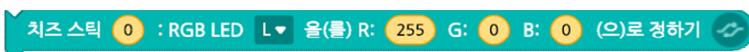
포트 Sa에 연결된 입력·출력 장치의 값을 원하는 범위로 변경합니다.



LED의 색과 밝기(어두운, 기본, 밝은)를 지정합니다.



LED의 색을 명도, 채도, 색상으로 지정합니다.



LED를 RGB값으로 조절합니다.



오브젝트가 말풍선으로 말을 하게 합니다.



오브젝트의 모양을 바꿉니다.



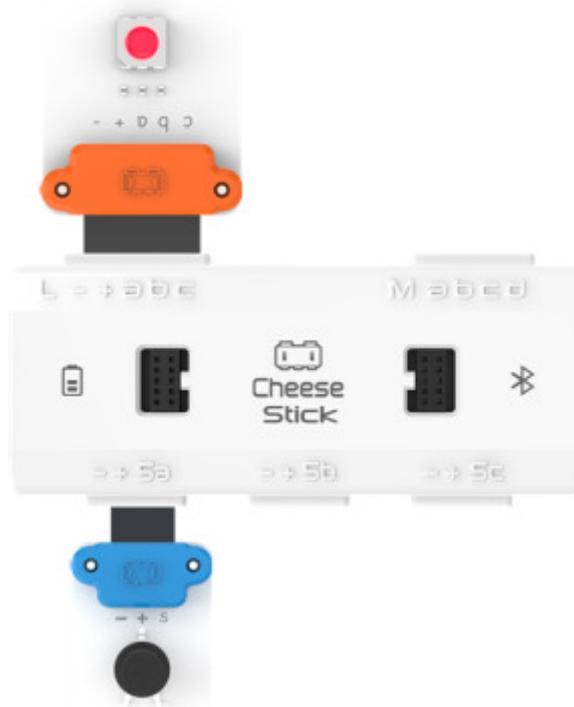
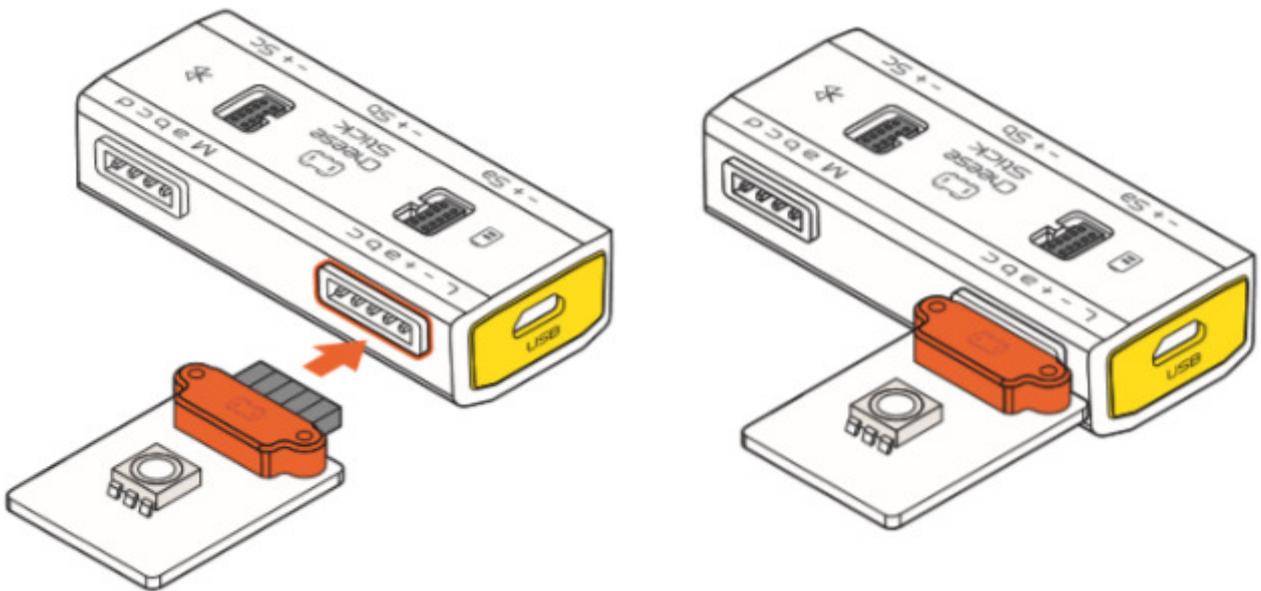
값의 크기 비교를 통해 명령을 설정합니다.



변수에 값을 저장합니다.



가변저항과 LED를 치즈스틱에 연결해 봅시다. 가변저항은 Sa 포트에 연결하고, LED는 L포트에 연결합니다.



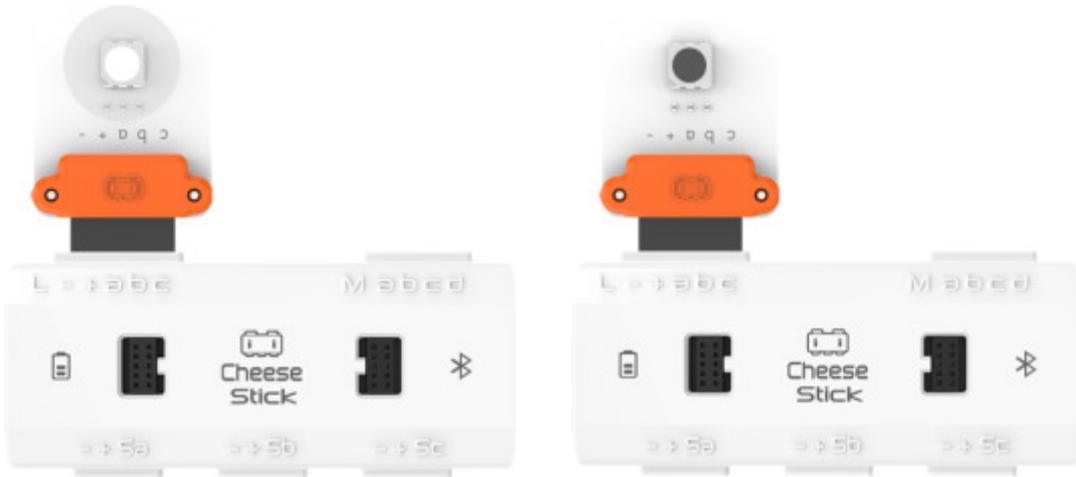
LED의 빛 세기를 강하게 하면 눈이 아플 수 있으니 빛의 세기를 약하게 해서 사용하거나 LED위에 종이컵을 덮어두면 좋습니다.

[연습과제1]



LED를 흰색으로 켜 봅시다.

[연습과제2]



LED가 흰색으로 깜빡거리도록 만들어 봅시다.

확인하기

[연습과제1]



[연습과제2]





LED를 흰색으로 켜는 여러 가지 방법을 살펴봅시다.

시작하기 버튼을 클릭했을 때

치즈 스틱 0 : RGB LED L 울(를) 기본 하얀색 으로 정하기

빨간색

주황색

노란색

초록색

하늘색

...

시작하기 버튼을 클릭했을 때

치즈 스틱 0 : RGB LED L 울(를) 으로 정하기

색상 0

채도 0

명도 100

빨강(R) 255

녹색(G) 255

파랑(B) 255

시작하기 버튼을 클릭했을 때

치즈 스틱 0 : RGB LED L 울(를) R: 255 G: 255 B: 255 (으)로 정하기



RGB 블록은 빨강(Red), 초록(Green), 파랑(Blue) 색상의 밝기를 조절하여 여러 색으로 만들 수 있습니다. R, G, B의 숫자를 모두 같이 하는 경우 흰색으로 표현됩니다. 각 자리의 최소값은 0, 최대값은 255입니다.



LED의 색을 흰색으로 하면서 밝기를 조절하기 위한 명령을 만들어 봅시다.

▶ 시작하기 버튼을 클릭했을 때

치즈 스틱 0 : RGB LED L 음(음) R: 10 G: 10 B: 10 (으)로 정하기

▶ 시작하기 버튼을 클릭했을 때

치즈 스틱 0 : RGB LED L 음(음) R: 50 G: 50 B: 50 (으)로 정하기

▶ 시작하기 버튼을 클릭했을 때

치즈 스틱 0 : RGB LED L 음(음) R: 255 G: 255 B: 255 (으)로 정하기



RGB 각각의 입력칸에 같은 숫자를 넣으면 빛의 세기가 조절되는 흰색을 만들 수 있습니다.



'전등' 오브젝트를 추가하고 흰색으로 LED를 켜기 위한 명령을 만들어 봅시다.



▶ 시작하기 버튼을 클릭했을 때

치즈 스틱 0 : RGB LED L 음(음) R: 255 G: 255 B: 255 (으)로 정하기



가변저항의 손잡이를 돌리면 빛의 세기가 변하는 LED를 만들겠습니다.
가변저항에서 입력되는 값을 인식하도록 명령을 만들어 봅시다.



시작하기 버튼을 클릭했을 때

치즈 스틱 0 : 포트 Sa 를 아날로그 입력으로 정하기

치즈 스틱 0 : RGB LED L 을(를) R: 255 G: 255 B: 255 (으)로 정하기



가변저항에서 입력되는 값을 RGB에서 인식할 수 있는 범위로 바꾸어 줍니다.



시작하기 버튼을 클릭했을 때

치즈 스틱 0 : 포트 Sa 를 아날로그 입력으로 정하기

치즈 스틱 0 : 입력 Sa 의 범위 0 ~ 1023 을(를) 0 ~ 255 소수점 없음 으로 정하기

치즈 스틱 0 : RGB LED L 을(를) R: 255 G: 255 B: 255 (으)로 정하기



가변저항은 0~1023의 범위를 가지고, RGB는 0~255 사이의 수를 인식합니다.
따라서 가변저항의 0~1023 범위를 0~255로 바꾸어 주어야 합니다.



가변저항의 값을 LED 켜기 블록에 넣어줍니다.



시작하기 버튼을 클릭했을 때

치즈 스틱 0 : 포트 Sa 를 아날로그 입력으로 정하기

치즈 스틱 0 : 입력 Sa 의 범위 0 ~ 1023 을(를) 0 ~ 255 소수점 없음 으로 정하기

치즈 스틱 0 : RGB LED L 을(를) R: 치즈 스틱 0 : 입력 Sa G: 치즈 스틱 0 : 입력 Sb B: 치즈 스틱 0 : 입력 Sc (으)로 정하기



밝기 조절을 계속 할 수 있도록 반복 블록을 사용해 봅시다.

▶ 시작하기 버튼을 클릭했을 때

치즈 스틱 0 : 포트 Sa 를 아날로그 입력으로 정하기

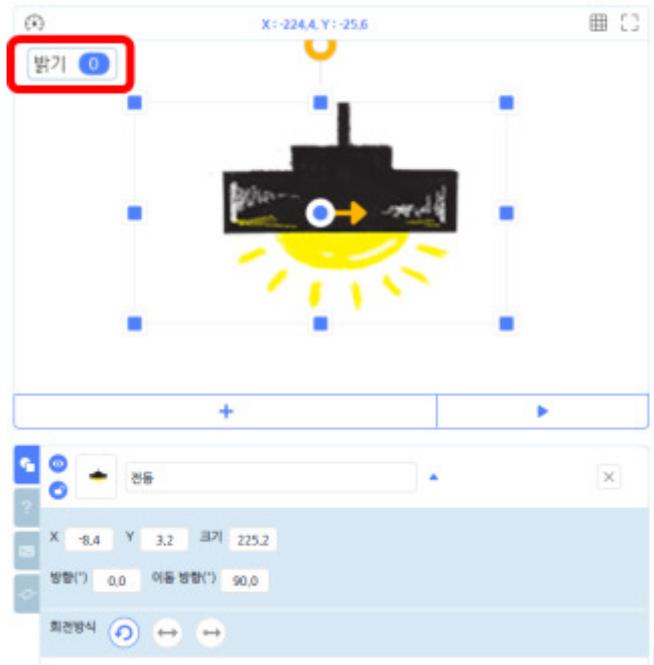
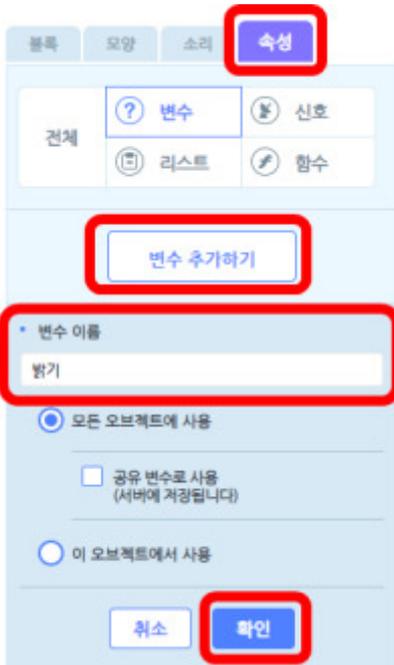
치즈 스틱 0 : 입력 Sa 의 범위 0 ~ 1023 을(를) 0 ~ 255 소수점 없음 으로 정하기

계속 반복하기

치즈 스틱 0 : RGB LED L 을(를) R: 치즈 스틱 0 : 입력 Sa G: 치즈 스틱 0 : 입력 Sb B: 치즈 스틱 0 : 입력 Sc (으)로 정하기



현재 전등의 밝기가 얼마인지 실행화면에 나타내 봅시다. ‘밝기’ 변수를 만들어 나타냅니다.



▶ 시작하기 버튼을 클릭했을 때

치즈 스틱 0 : 포트 Sa 를 아날로그 입력으로 정하기

치즈 스틱 0 : 입력 Sa 의 범위 0 ~ 1023 을(를) 0 ~ 255 소수점 없음 으로 정하기

계속 반복하기

밝기 를 치즈 스틱 0 : 입력 Sa (으)로 정하기

밝기 값 을(를) 말하기

치즈 스틱 0 : RGB LED L 을(를) R: 치즈 스틱 0 : 입력 Sa G: 치즈 스틱 0 : 입력 Sb B: 치즈 스틱 0 : 입력 Sc (으)로 정하기



실행화면에서도 전등의 불이 켜졌다 꺼지도록 만들어 봅시다.



```

▶ 시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : 포트 Sa 를 아날로그 입력으로 정하기
  치즈 스틱 0 : 입력 Sa 의 범위 0 ~ 1023 을(를) 0 ~ 255 소수점 없음 으로 정하기
  계속 반복하기
    밝기 를 치즈 스틱 0 : 입력 Sa (으)로 정하기
    밝기 값 을(를) 말하기
    치즈 스틱 0 : RGB LED L 을(를) R: 치즈 스틱 0 : 입력 Sa G: 치즈 스틱 0 : 입력 Sb B: 치즈 스틱 0 : 입력 Sc (으)로 정하기
    만일 치즈 스틱 0 : 입력 Sa = 0 (이)라면
      전등_꺼짐 모양으로 바꾸기
    아니면
      전등_켜짐 모양으로 바꾸기
  
```



가변저항의 값에 따라 불이 켜지고 꺼지고, 점점 밝아졌다 어두워졌다 합니다. 따라서 가변저항의 값이 0일 때는 불이 꺼져있는 상태입니다.



학습문제로 공부한 내용을 정리해 봅시다.

[문제1] 아래 명령 중 흰색 LED로 나타나지 않는 경우를 선택해 봅시다.

- ① 시작하기 버튼을 클릭했을 때
치즈 스틱 0 : RGB LED L 을(를) R: 10 G: 10 B: 10 (으)로 정하기
- ② 시작하기 버튼을 클릭했을 때
치즈 스틱 0 : RGB LED L 을(를) R: 50 G: 50 B: 50 (으)로 정하기
- ③ 시작하기 버튼을 클릭했을 때
치즈 스틱 0 : RGB LED L 을(를) R: 255 G: 255 B: 255 (으)로 정하기
- ④ 시작하기 버튼을 클릭했을 때
치즈 스틱 0 : RGB LED L 을(를) R: 150 G: 0 B: 150 (으)로 정하기

[문제2] 가변저항을 이용하여 밝기가 조절되는 흰색 LED를 만들려고 합니다. 아래의 그림처럼 프로그램을 만들어 실행하였으나 밝기가 조절되지 않는 문제가 발견되었습니다. 어디를 어떻게 수정해야 할지 찾아봅시다.

```

시작하기 버튼을 클릭했을 때
치즈 스틱 0 : 포트 Sa 를 아날로그 입력으로 정하기
치즈 스틱 0 : 입력 Sa 의 범위 0 ~ 1023 을(를) 0 ~ 255 소수점 없음 으로 정하기
계속 반복하기
  밝기 를 치즈 스틱 0 : 입력 Sa (으)로 정하기
  밝기 값 을(를) 말하기
치즈 스틱 0 : RGB LED L 을(를) R: 치즈 스틱 0 : 입력 Sa G: 치즈 스틱 0 : 입력 Sb B: 치즈 스틱 0 : 입력 Sa (으)로 정하기
만일 치즈 스틱 0 : 입력 Sa - 0 (이)라면
  전등_꺼짐 모양으로 바꾸기
아니면
  전등_켜짐 모양으로 바꾸기
  
```

Basic Programing

9 춤추는 종이컵 인형 만들기

가변저항과 서보모터, 종이컵을 이용해 춤추는 종이컵 인형을 만들 수 있어요.



가변저항과 서보모터를 이용하여 재미있는 장난감을 만들 수 있어요. 종이컵 인형도 함께 만들어 인형을 춤추게 만들어 봅시다.



그림 출처 : 보법샘의 쉽게 떠먹는 미술수업 (<http://sblog.i-scream.co.kr/platon1026/168?sso=ok#none>)



준비를 해봅시다.



가변저항

서보모터

종이컵

치즈스틱 본체



사용할 블록을 알아보시다.



포트 Sa, Sb, Sc에 연결되는
입력·출력 장치를 정할 때
필요합니다.



포트 Sa에 연결된 입력·출력
장치의 값이 저장되어
있습니다.



포트 Sa에 연결된 입력·출력
장치의 값을 원하는 범위로
변경합니다.



오브젝트가 말풍선으로 말을
하게 합니다.



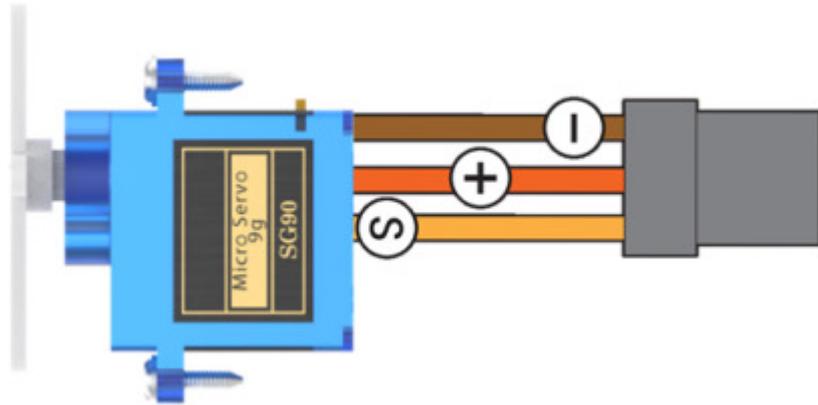
서보모터의 각도를
설정합니다.



설정된 시간만큼 기다린 후
다음 블록을 실행합니다.

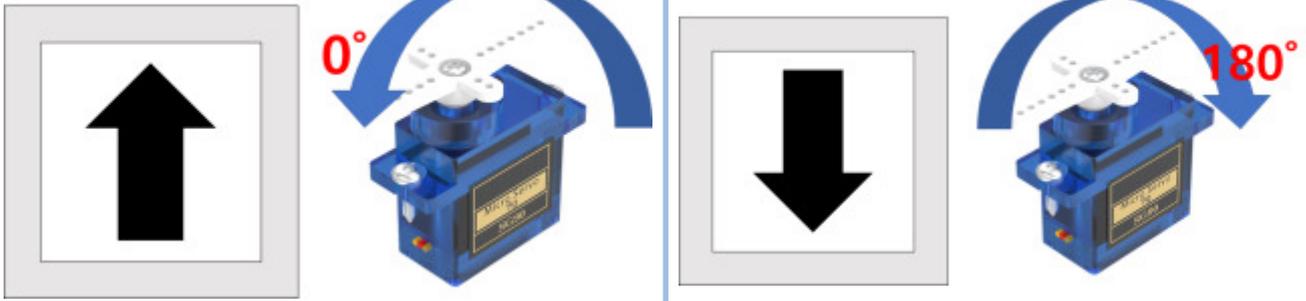


가변저항과 서보모터를 치즈스틱에 연결해 봅시다.



서보모터는 S포트 중 한 곳에 연결합니다. 서보모터에서 이어지는 케이블의 색을 잘 확인하여 올바른 방향으로 꽂아주세요. 서보모터를 여러 개 사용하는 경우는 전력 부족으로 정상적으로 작동하지 않을 수도 있습니다.

[연습과제1]



서보모터를 움직여 봅시다. 위쪽 화살표를 누르면 0°, 아래쪽 화살표를 누르면 180°로 이동하도록 만들어 봅시다.

[연습과제2]



서보모터가 자동으로 0°에서 180°도를 왔다갔다 하도록 만들어 봅시다.

확인하기

[연습과제1]

서보 모터 Sc 의 각도를 0 도로 정하기

서보 모터 Sc 의 각도를 180 도로 정하기

[연습과제2]

계속 반복하기

0.5 초 기다리기

0.5 초 기다리기



가변저항의 값 범위를 서보모터 동작 범위로 바꾸어 줍니다.

▶ 시작하기 버튼을 클릭했을 때

```

치즈 스틱 0 : 포트 Sa 를 아날로그 입력으로 정하기
치즈 스틱 0 : 입력 Sa 의 범위 0 ~ 1023 을(를) 0 ~ 180 소수점 없음 으로 정하기
  
```



가변저항의 변환된 값을 확인하기 위한 명령을 만들어 봅시다.

▶ 시작하기 버튼을 클릭했을 때

```

치즈 스틱 0 : 포트 Sa 를 아날로그 입력으로 정하기
치즈 스틱 0 : 입력 Sa 의 범위 0 ~ 1023 을(를) 0 ~ 180 소수점 없음 으로 정하기
계속 반복하기
  치즈 스틱 0 : 입력 Sa 을(를) 말하기
  
```



가변저항의 값을 서보모터의 각도에 입력하여 줍니다.

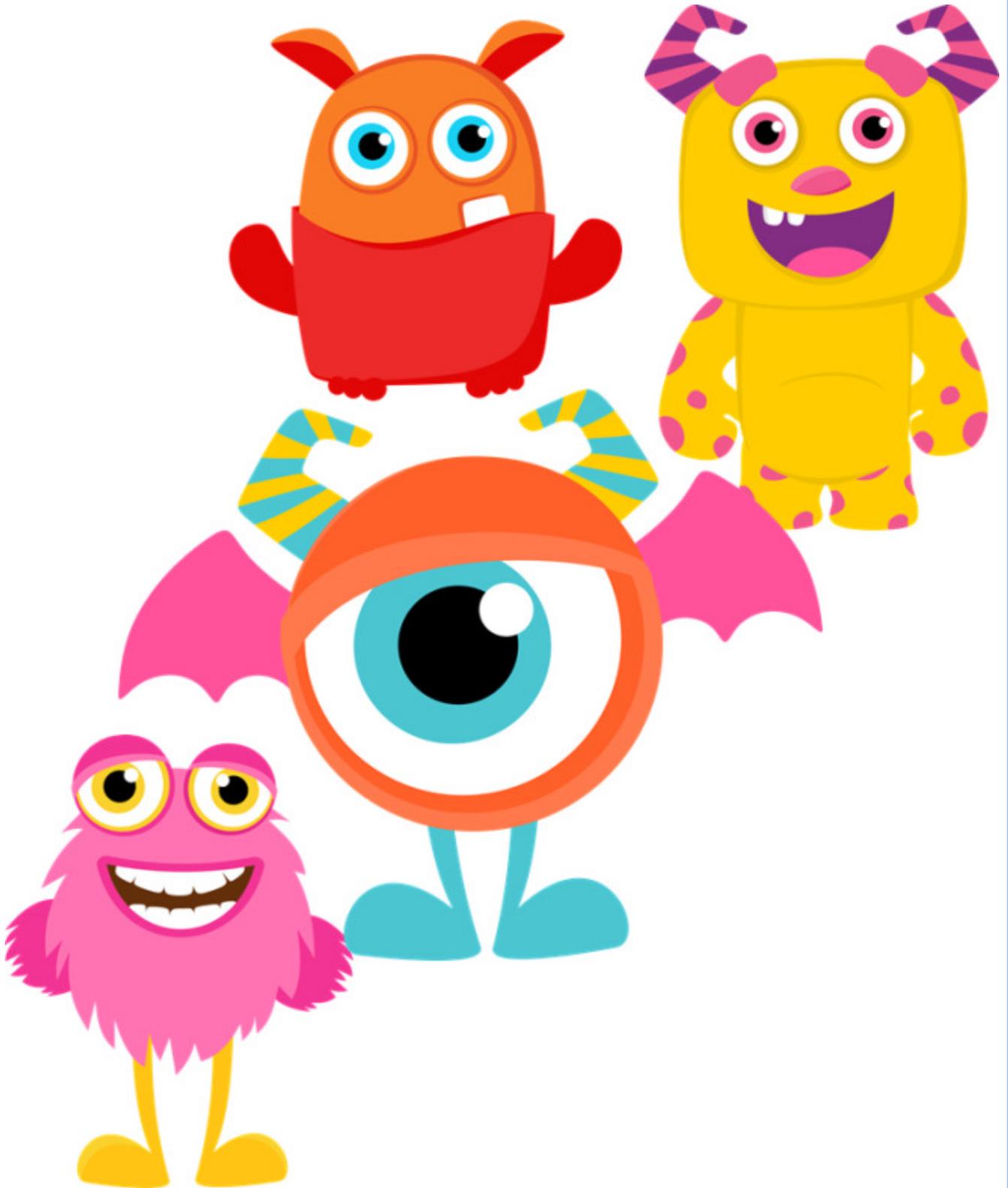
▶ 시작하기 버튼을 클릭했을 때

```

치즈 스틱 0 : 포트 Sa 를 아날로그 입력으로 정하기
치즈 스틱 0 : 입력 Sa 의 범위 0 ~ 1023 을(를) 0 ~ 180 소수점 없음 으로 정하기
계속 반복하기
  치즈 스틱 0 : 입력 Sa 을(를) 말하기
  치즈 스틱 0 : 서보 모터 Sc 의 각도를 치즈 스틱 0 : 입력 Sa 도로 정하기
  
```



종이컵을 사용하여 재미있는 인형을 만들어 봅시다. <부록>의 눈, 코, 입을 사용하여도 좋습니다.





서보모터와 종이컵 인형을 연결하고 프로그램을 실행시켜 봅시다.



가변저항의 값이 작으면 서보모터도 작게 왔다갔다 거리고, 가변저항의 값이 크면 서보모터도 크게 왔다갔다 거리는 프로그램을 만들어 봅시다.

```

시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : 포트 Sa 를 아날로그 입력으로 정하기
  치즈 스틱 0 : 입력 Sa 의 범위 0 ~ 1023 을(를) 0 ~ 180 소수점 없음 으로 정하기
  계속 반복하기
    치즈 스틱 0 : 입력 Sa 을(를) 말하기
    2 번 반복하기
      치즈 스틱 0 : 서보 모터 Sc 의 각도를 치즈 스틱 0 : 입력 Sa 도로 정하기
      0.2 초 기다리기
      치즈 스틱 0 : 서보 모터 Sc 의 각도를 0 도로 정하기
      0.2 초 기다리기
  
```



학습문제로 공부한 내용을 정리해 봅시다.

[문제1] 입력 센서의 값을 내가 원하는 범위로 바꾸어 사용할 때 쓰는 명령 블록을 선택해 봅시다.

- ① 치즈 스틱 0 : 스텝 모터 100 스텝을 속도 300 스텝/초로 회전하기
- ② 치즈 스틱 0 : 입력 Sa 의 범위 0 ~ 255 을(를) 0 ~ 100 소수점 없음 으로 정하기
- ③ 치즈 스틱 0 : 서보 모터 Sc 의 각도를 0 도로 정하기
- ④ 치즈 스틱 0 : 포트 Sa 를 아날로그 입력으로 정하기

[문제2] 가변저항 값에 따라 서보모터가 움직임을 반복하는 속도를 좀 더 빠르게 만들려고 합니다. 아래 명령 코드에서 어떤 부분을 수정하면 될지 생각해 봅시다.

```

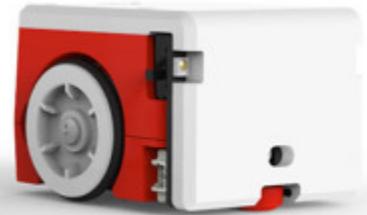
시작하기 버튼을 클릭했을 때
  치즈 스틱 0 : 포트 Sa 를 아날로그 입력으로 정하기
  치즈 스틱 0 : 입력 Sa 의 범위 0 ~ 1023 을(를) 0 ~ 180 소수점 없음 으로 정하기
  계속 반복하기
    치즈 스틱 0 : 입력 Sa 을(를) 말하기
    2 번 반복하기
      치즈 스틱 0 : 서보 모터 Sc 의 각도를 치즈 스틱 0 : 입력 Sa 도로 정하기
      0.2 초 기다리기
      치즈 스틱 0 : 서보 모터 Sc 의 각도를 0 도로 정하기
      0.2 초 기다리기
  
```

10 햄스터 로봇 알아보기

햄스터를 컴퓨터에 연결해 프로그래밍 할 수 있어요.



여러 가지 센서와 LED, 모터를 프로그래밍 할 수 있는 로봇인 햄스터에 대하여 알아보시다.



10개의 센서와 2개의 LED, 2개의 DC모터를 프로그래밍 할 수 있는 로봇



햄스터를 충전하고 전원을 켜봅시다.



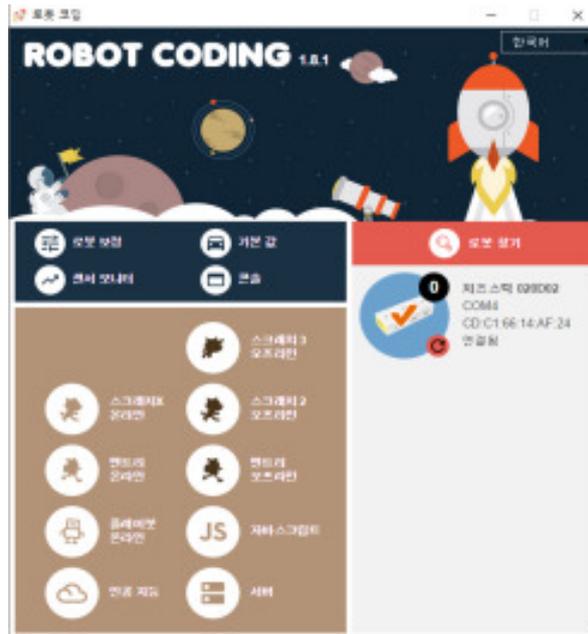
5핀 케이블로 충전합니다.



전원버튼을 위로 올리면 켜지고
아래로 내리면 꺼집니다.



치즈스틱과 햄스터를 엔트리에 연결해 봅시다.



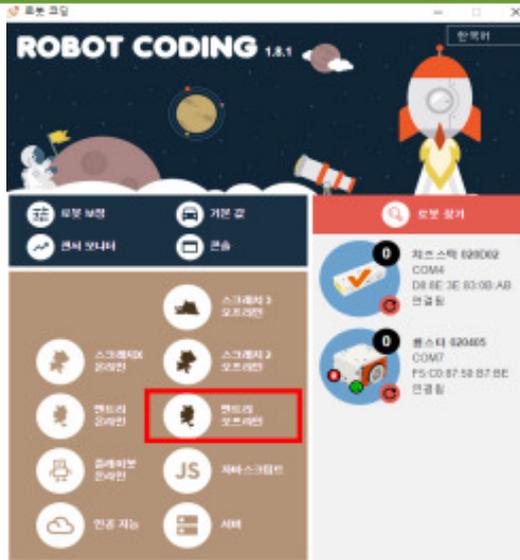
로봇 코딩 소프트웨어를 사용해 치즈스틱을 컴퓨터와 연결합니다.



컴퓨터에 'USB 동글'을 하나 더 연결하고 햄스터의 전원을 켭니다.



로봇 찾기를 클릭합니다.



치즈스틱과 햄스터가 연결되었으면 '엔트리 오프라인'을 클릭합니다.



여러 로봇을 선택하고 새 파일을 클릭합니다.



하드웨어 탭에 치즈스틱, 햄스터 블록이 모두 생성된 것을 확인합니다.



사용할 블록을 알아봅시다.

햄스터 0 : 앞으로 1 초 이동하기

햄스터가 1초 동안 앞으로 이동합니다.

햄스터 0 : 뒤로 1 초 이동하기

햄스터가 1초 동안 뒤로 이동합니다.

햄스터 0 : 왼쪽으로 1 초 돌기

햄스터가 1초 동안 왼쪽(오른쪽)으로 회전합니다.

햄스터 0 : 왼쪽 바퀴 10 오른쪽 바퀴 10 만큼 바꾸기

햄스터 양 쪽 바퀴의 속도를 각각 바꿉니다. 양의 값을 넣으면 전진, 음의 값을 넣으면 후진합니다.

햄스터 0 : 왼쪽 바퀴 30 오른쪽 바퀴 30 (으)로 정하기

햄스터 양 쪽 바퀴의 속도를 각각 정합니다. 양의 값을 넣으면 전진, 음의 값을 넣으면 후진합니다.

햄스터 0 : 말판 앞으로 한 칸 이동하기

햄스터 0 : 말판 왼쪽으로 한 번 돌기



햄스터는 검은 선으로 이루어진 격자판 위에 놓으면 한 점에서 다른 점 까지를 한 칸으로 인식합니다. 두꺼운 사인펜이나 매직을 이용해 말판을 만들어 활용할 수 있습니다.



햄스터를 움직여 봅시다.

▶ 시작하기 버튼을 클릭했을 때
 햄스터 0 : 앞으로 1 초 이동하기

▶ 시작하기 버튼을 클릭했을 때
 햄스터 0 : 뒤로 1 초 이동하기

▶ 시작하기 버튼을 클릭했을 때
 계속 반복하기
 햄스터 0 : 앞으로 1 초 이동하기
 햄스터 0 : 뒤로 1 초 이동하기

▶ 시작하기 버튼을 클릭했을 때
 햄스터 0 : 왼쪽으로 1 초 돌기

▶ 시작하기 버튼을 클릭했을 때
 햄스터 0 : 오른쪽으로 1 초 돌기



햄스터 ○로 ○초 이동하기/돌기 블록을 사용하면 정해진 시간 동안 햄스터를 움직일 수 있습니다. 단, 햄스터의 속력을 제어할 수는 없습니다.

```

시작하기 버튼을 클릭했을 때
햄스터 0 : 왼쪽 바퀴 30 오른쪽 바퀴 30 (으)로 정하기

```

```

시작하기 버튼을 클릭했을 때
햄스터 0 : 왼쪽 바퀴 -30 오른쪽 바퀴 -30 (으)로 정하기

```

```

시작하기 버튼을 클릭했을 때
계속 반복하기
  햄스터 0 : 왼쪽 바퀴 30 오른쪽 바퀴 30 (으)로 정하기
  1 초 기다리기
  햄스터 0 : 왼쪽 바퀴 -30 오른쪽 바퀴 -30 (으)로 정하기
  1 초 기다리기

```

햄스터 바퀴 ○으로 정하기 블록을 이용하면 속력을 변화시키며 햄스터를 움직일 수 있습니다. 이때 시간을 제어하기 위해서는 ○초 기다리기 블록을 함께 사용합니다.

```

시작하기 버튼을 클릭했을 때
계속 반복하기
  햄스터 0 : 왼쪽 바퀴 30 오른쪽 바퀴 30 만큼 바꾸기

```

```

시작하기 버튼을 클릭했을 때
계속 반복하기
  햄스터 0 : 왼쪽 바퀴 30 오른쪽 바퀴 30 (으)로 정하기

```

 햄스터 바퀴 ○으로 정하기 블록과 햄스터 바퀴 ○만큼 바꾸기 블록은 쓰임이 다릅니다. 정하기는 속력을 일정하게 유지, 바꾸기는 속력을 일정하게 증가시킵니다. 두 개의 명령어를 각각 실행시켜보며 차이점을 살펴봅시다.

▶ 시작하기 버튼을 클릭했을 때

햄스터 0 : 왼쪽 바퀴 -30 오른쪽 바퀴 30 (으)로 정하기

가운데를 중심으로 왼쪽으로 제자리 회전

▶ 시작하기 버튼을 클릭했을 때

햄스터 0 : 왼쪽 바퀴 30 오른쪽 바퀴 -30 (으)로 정하기

가운데를 중심으로 오른쪽으로 제자리 회전

▶ 시작하기 버튼을 클릭했을 때

햄스터 0 : 왼쪽 바퀴 0 오른쪽 바퀴 30 (으)로 정하기

왼쪽 바퀴를 축으로 제자리 회전

▶ 시작하기 버튼을 클릭했을 때

햄스터 0 : 왼쪽 바퀴 30 오른쪽 바퀴 0 (으)로 정하기

오른쪽 바퀴를 축으로 제자리 회전

▶ 시작하기 버튼을 클릭했을 때

햄스터 0 : 왼쪽 바퀴 60 오른쪽 바퀴 30 (으)로 정하기

앞으로 이동하면서 오른쪽으로 회전

▶ 시작하기 버튼을 클릭했을 때

햄스터 0 : 왼쪽 바퀴 30 오른쪽 바퀴 60 (으)로 정하기

앞으로 이동하면서 왼쪽으로 회전



햄스터 바퀴 0으로 정하기 블록을 사용해 양쪽 바퀴의 속력을 다르게 하면 햄스터를 다양하게 회전시킬 수 있습니다.
각각의 명령어를 실행해보며 차이점을 살펴봅시다.



학습문제로 공부한 내용을 정리해 봅시다.

[문제1] 다음 명령 블록을 실행했을 때 햄스터가 어떻게 동작할지 각각 작성해 봅시다.

①

```

    when green flag clicked
      repeat continuously
        turn hamster 0 : 왼쪽 바퀴 30 오른쪽 바퀴 30 (으)로 정하기
        wait 1 초 기다리기
        turn hamster 0 : 왼쪽 바퀴 -30 오른쪽 바퀴 -30 (으)로 정하기
        wait 1 초 기다리기
  
```

②

```

    when green flag clicked
      repeat continuously
        turn hamster 0 : 왼쪽 바퀴 30 오른쪽 바퀴 30 만큼 바꾸기
  
```

③

```

    when green flag clicked
      repeat continuously
        turn hamster 0 : 왼쪽 바퀴 30 오른쪽 바퀴 30 (으)로 정하기
  
```



학습문제로 공부한 내용을 정리해 봅시다.

[문제2] 다음 명령 블록을 실행했을 때 햄스터가 어떻게 동작할지 각각 작성해 봅시다.

①

시작하기 버튼을 클릭했을 때
 햄스터 0 : 왼쪽 바퀴 -30 오른쪽 바퀴 30 (으)로 정하기

②

시작하기 버튼을 클릭했을 때
 햄스터 0 : 왼쪽 바퀴 0 오른쪽 바퀴 30 (으)로 정하기

③

시작하기 버튼을 클릭했을 때
 햄스터 0 : 왼쪽 바퀴 30 오른쪽 바퀴 60 (으)로 정하기

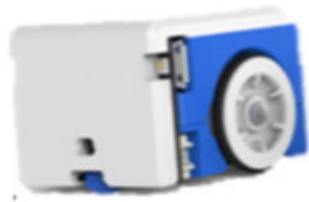
Basic Programing

11 햄스터 조종기 만들기

조이스틱을 사용해 햄스터를 프로그래밍 할 수 있어요.



조이스틱을 사용하여 햄스터를 다양하게 움직일 수 있어요. 조이스틱을 사용해 햄스터를 조종해 봅시다.



준비를 해봅시다.



조이스틱
서보모터



치즈스틱 본체



햄스터



사용할 블록을 알아보시다.



조이스틱을 사용할 때 필요합니다.



조이스틱의 좌우, 상하값을 설정할 때 필요합니다.



조이스틱의 측정값을 0~100의 값으로 설정합니다.



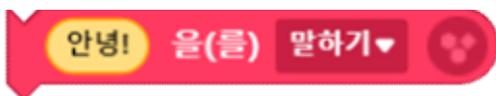
값의 크기 비교를 통해 명령을 설정합니다.



주어진 조건을 만족했을 때와 만족하지 않았을 때 서로 다른 명령을 실행합니다.



앞의 명령을 설정한 시간동안 유지시켜줍니다.



오브젝트가 말풍선으로 말을 하게 합니다.



치즈스틱의 조이스틱을 사용해 햄스터를 움직여 봅시다.

```

시작하기 버튼을 클릭했을 때
치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
치즈 스틱 0 : PID x1 의 범위 0 ~ 255 을(를) -100 ~ 100 소수점 없음 으로 정하기
치즈 스틱 0 : PID y1 의 범위 0 ~ 255 을(를) -100 ~ 100 소수점 없음 으로 정하기
  
```



x1은 좌, 우, y1은 상, 하로 조작한 값을 말합니다.
만든 블록을 삭제하지 않고 다른 블록을 가져옵니다.

```

시작하기 버튼을 클릭했을 때
  계속 반복하기
    만일 치즈 스틱 0 : PID y1 > 50 (이)라면
      햄스터 0 : 앞으로 1 초 이동하기
  
```

조이스틱을 위로 조작했을 때 햄스터가 앞으로 1초 이동하도록 합니다.

```

시작하기 버튼을 클릭했을 때
  계속 반복하기
    만일 -50 > 치즈 스틱 0 : PID y1 (이)라면
      햄스터 0 : 뒤로 1 초 이동하기
  
```

조이스틱을 아래로 조작했을 때 햄스터가 뒤로 1초 이동하도록 합니다.



조이스틱을 왼쪽으로 조작했을 때 햄스터가 왼쪽으로 1초 회전하도록 합니다.



조이스틱을 오른쪽으로 조작했을 때 햄스터가 오른쪽으로 1초 회전하도록 합니다.



치즈스틱의 조이스틱을 사용해 햄스터를 앞, 뒤, 좌, 우로 이동시켜봅시다.

시작하기 버튼을 클릭했을 때

1 초 기다리기

계속 반복하기

만일 치즈 스틱 0 : PID y1 > 50 (이)라면

햄스터 0 : 앞으로 1 초 이동하기

만일 -50 > 치즈 스틱 0 : PID y1 (이)라면

햄스터 0 : 뒤로 1 초 이동하기

만일 치즈 스틱 0 : PID x1 > 50 (이)라면

햄스터 0 : 오른쪽으로 1 초 돌기

만일 -50 > 치즈 스틱 0 : PID x1 (이)라면

햄스터 0 : 왼쪽으로 1 초 돌기

 기계에 따라 시작하기 버튼을 클릭했을 때 짧은 순간동안 조이스틱의 값이 -100으로 측정되는 경우가 있어 이로 인해 햄스터가 움직이는 것을 막기 위해 1초 기다리기 블록을 가장 먼저 사용합니다.

시작하기 버튼을 클릭했을 때

1 초 기다리기

계속 반복하기

만일 치즈 스틱 0 : PID y1 > 50 (이)라면

햄스터 0 : 왼쪽 바퀴 30 오른쪽 바퀴 30 (으)로 정하기

만일 -50 > 치즈 스틱 0 : PID y1 (이)라면

햄스터 0 : 왼쪽 바퀴 -30 오른쪽 바퀴 -30 (으)로 정하기

만일 치즈 스틱 0 : PID x1 > 50 (이)라면

햄스터 0 : 왼쪽 바퀴 30 오른쪽 바퀴 0 (으)로 정하기

만일 -50 > 치즈 스틱 0 : PID x1 (이)라면

햄스터 0 : 왼쪽 바퀴 0 오른쪽 바퀴 30 (으)로 정하기

만일 치즈 스틱 0 : PID 버튼 1 올(름) 클릭했는가 ? (이)라면

햄스터 0 : 왼쪽 바퀴 0 오른쪽 바퀴 0 (으)로 정하기

 바퀴의 속력을 변화시키며 햄스터를 이동시킬 경우 햄스터를 정지시키는 명령을 추가해주는 것이 좋습니다. 양쪽 바퀴의 속력을 0으로 하면 햄스터가 정지합니다.

 시간을 이용해 이동하기 명령을 실행하면 이동하는 시간 동안은 다른 명령을 수행할 수 없는 반면 속력을 이용해 이동하기 명령을 실행하면 다른 명령을 즉각 수행할 수 있습니다.



학습문제로 공부한 내용을 정리해 봅시다.

[문제1] 조이스틱을 위로 움직였을 때 햄스터가 앞으로 이동할 수 있도록 빈 칸에 명령을 작성해 봅시다.

시작하기 버튼을 클릭했을 때

계속 반복하기

만일 치즈 스틱 0 : PID 50 (이)라면

햄스터 0 : 앞으로 1 초 이동하기

[문제2] 조이스틱을 오른쪽으로 움직였을 때 햄스터가 오른쪽으로 1초 회전할 수 있도록 빈 칸에 명령을 작성해 봅시다.

시작하기 버튼을 클릭했을 때

계속 반복하기

만일 치즈 스틱 0 : PID 50 (이)라면

햄스터 0 : 오른쪽으로 1 초 돌기



학습문제로 공부한 내용을 정리해 봅시다.

[문제3] 기계에 따라 시작하기 버튼을 클릭했을 때 짧은 순간동안 조이스틱의 값이 -100으로 측정되는 경우가 있어 나의 의도와 다르게 햄스터가 움직이는 경우가 있습니다. 이를 해결하기 위한 방법을 작성해 봅시다.

▶ 시작하기 버튼을 클릭했을 때

```

치즈 스틱 0 : PID-13 조이스틱과 버튼 ▶ 시작하기
치즈 스틱 0 : PID x1 의 범위 0 ~ 255 을(를) -100 ~ 100 소수점 없음 ▶ 으로 정하기
치즈 스틱 0 : PID y1 의 범위 0 ~ 255 을(를) -100 ~ 100 소수점 없음 ▶ 으로 정하기
  
```

▶ 시작하기 버튼을 클릭했을 때

```

1 초 기다리기
계속 반복하기
만일 치즈 스틱 0 : PID y1 > 50 (이)라면
  햄스터 0 : 왼쪽 바퀴 30 오른쪽 바퀴 30 (으)로 정하기
만일 -50 > 치즈 스틱 0 : PID y1 (이)라면
  햄스터 0 : 왼쪽 바퀴 -30 오른쪽 바퀴 -30 (으)로 정하기
만일 치즈 스틱 0 : PID x1 > 50 (이)라면
  햄스터 0 : 왼쪽 바퀴 30 오른쪽 바퀴 0 (으)로 정하기
만일 -50 > 치즈 스틱 0 : PID x1 (이)라면
  햄스터 0 : 왼쪽 바퀴 0 오른쪽 바퀴 30 (으)로 정하기
만일 치즈 스틱 0 : PID 버튼 1 을(를) 클릭했는가 ? (이)라면
  햄스터 0 : 왼쪽 바퀴 0 오른쪽 바퀴 0 (으)로 정하기
  
```

Basic Programing

12 지뢰탐지 로봇 만들기

햄스터의 바닥센서, 근접센서를 사용할 수 있어요.

지뢰를 감지하면 LED와 소리로 알려주는 지뢰탐지 로봇을 만들 수 있어요.



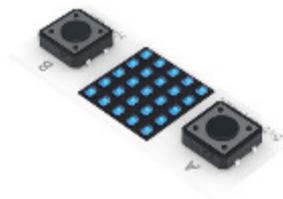
치즈스틱의 여러 버튼과 포텐쇼미터를 사용해 속력과 LED를 제어하며 햄스터를 조종해 봅시다.



준비를 해봅시다.



조이스틱



5×5매트릭스
LED



치즈스틱 본체



햄스터



햄스터의 바닥 센서에 대하여 알아봅시다.

왼쪽 바닥 센서

오른쪽 바닥 센서



햄스터의 바닥 센서는 적외선을 이용해 0~255단계를 감지합니다.

A4용지에 검정 사인펜으로 지뢰를 그린 후 바닥 센서가 흰 종이 부분과 지뢰 부분을 각각 어떻게 감지하는지 확인해 봅시다.

▶ 시작하기 버튼을 클릭했을 때

계속 반복하기 ↻

햄스터 0 : 왼쪽 바닥 센서 ▶ 을(를) 말하기 ▼

89



흰 종이 위

3



지뢰 위



조이스틱으로 햄스터를 제어할 수 있는 프로그램을 만들어 봅시다.

▶ 시작하기 버튼을 클릭했을 때

```

치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기
치즈 스틱 0 : PID x1 의 범위 0 ~ 255 을(를) -100 ~ 100 소수점 없음 으로 정하기
치즈 스틱 0 : PID y1 의 범위 0 ~ 255 을(를) -100 ~ 100 소수점 없음 으로 정하기
  
```

▶ 시작하기 버튼을 클릭했을 때

```

1 초 기다리기
계속 반복하기
  만일 치즈 스틱 0 : PID y1 > 50 (이)라면
    햄스터 0 : 왼쪽 바퀴 30 오른쪽 바퀴 30 (으)로 정하기
  만일 -50 > 치즈 스틱 0 : PID y1 (이)라면
    햄스터 0 : 왼쪽 바퀴 -30 오른쪽 바퀴 -30 (으)로 정하기
  만일 치즈 스틱 0 : PID x1 > 50 (이)라면
    햄스터 0 : 왼쪽 바퀴 30 오른쪽 바퀴 0 (으)로 정하기
  만일 -50 > 치즈 스틱 0 : PID x1 (이)라면
    햄스터 0 : 왼쪽 바퀴 0 오른쪽 바퀴 30 (으)로 정하기
  만일 치즈 스틱 0 : PID 버튼 1 을(를) 클릭했는가 ? (이)라면
    햄스터 0 : 왼쪽 바퀴 0 오른쪽 바퀴 0 (으)로 정하기
  
```



지뢰를 탐지하는 프로그램을 만들어 봅시다.

```

시작하기 버튼을 클릭했을 때
  계속 반복하기
    만일 <20> > 햄스터 <0> : 왼쪽 바닥 센사 (이)라면
    아니면
  
```

```

시작하기 버튼을 클릭했을 때
  계속 반복하기
    만일 <20> > 햄스터 <0> : 오른쪽 바닥 센사 (이)라면
    아니면
  
```

두 개의 바닥 센서 중 하나가 지뢰를 탐지할 경우 명령을 수행할 수 있어야
 하므로 위와 같이 왼쪽 바닥 센서와 오른쪽 바닥 센서가
 지뢰를 탐지하는 명령어를 각각 만듭니다.

```

시작하기 버튼을 클릭했을 때
  계속 반복하기
    만일 <20> > 햄스터 <0> : 왼쪽 바닥 센사 또는 <20> > 햄스터 <0> : 오른쪽 바닥 센사 (이)라면
    아니면
  
```

판단 탭의 '○ 또는 ○' 블록을 사용해 두 개의 명령을 하나로 합칠 수 있습니다.

시작하기 버튼을 클릭했을 때

계속 반복하기

만일 < 20 > 햄스터 0 : 왼쪽 바닥 센서 또는 < 20 > 햄스터 0 : 오른쪽 바닥 센서 (이)라면

치즈 스틱 0 : HAT-010 배경 빨간색 X 울(를) x: 0 y: 0 에 그리기

아니면

치즈 스틱 0 : HAT-010 배경 지우기

시작하기 버튼을 클릭했을 때

치즈 스틱 0 : PID-13 조이스틱과 버튼 시작하기

치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기

치즈 스틱 0 : PID x1 의 범위 0 ~ 255 울(를) -100 ~ 100 소수점 없음 으로 정하기

치즈 스틱 0 : PID y1 의 범위 0 ~ 255 울(를) -100 ~ 100 소수점 없음 으로 정하기

💡 지뢰를 탐지하면 치즈스틱의 5x5 매트릭스 LED에 X표시를 하도록 프로그래밍하기 위해서는

치즈 스틱 0 : HAT-010 5x5 매트릭스 시작하기 를 반드시 사용해야 합니다.

💡 지뢰를 탐지했을 때 LED 뿐 아니라 소리까지 나도록 하기 위해서

치즈 스틱 0 : 버저 음을 1000 Hz로 정하기 블록을 사용합니다.

단, 지뢰를 탐지한 후 지뢰가 없는 곳으로 간 후에는 소리가 나지 않도록

치즈 스틱 0 : 소리 끄기 블록도 함께 사용합니다.

```

시작하기 버튼을 클릭했을 때
  계속 반복하기
    만일 < 20 > 햄스터 0 : 왼쪽 바닥 센서 > 또는 < 20 > 햄스터 0 : 오른쪽 바닥 센서 > (이)라면
      치즈 스틱 0 : HAT-010 배경 빨간색 X 울(를) x: 0 y: 0 에 그리기
      치즈 스틱 0 : 버저 음을 1000 Hz로 정하기
    아니면
      치즈 스틱 0 : HAT-010 배경 지우기
      치즈 스틱 0 : 소리 끄기
  
```



햄스터의 근접 센서에 대하여 알아보시다.



```

시작하기 버튼을 클릭했을 때
  계속 반복하기
    햄스터 0 : 왼쪽 근접 센서 울(를) 말하기
  
```

햄스터의 근접 센서는 적외선을 이용해 1~30mm를 감지합니다.
 근접센서 앞에 손을 갖다 대보고 떼보면서 근접 센서가
 대상을 어떻게 감지하는지 확인해 봅시다.

21



대상이 멀 때

39



대상이 가까울 때

15



대상이 너무 가까울 때



충돌을 예방하는 프로그램을 만들어 봅시다.

```

▶ 시작하기 버튼을 클릭했을 때
  계속 반복하기
    만일 햄스터 0 : 왼쪽 근접 센서 > 40 (이)라면
      치즈 스틱 0 : HAT-010 배경 빨간색 심음 울(를) x: 0 y: 0 에 그리기
    아니면
      치즈 스틱 0 : HAT-010 배경 지우기
  
```

장애물이 햄스터에 가까워지면 5x5 매트릭스 LED에 '싫음' 표정이 나타나게 합니다.

```

치즈 스틱 0 : 버저 음을 1000 Hz로 정하기
  
```

활용할 수도 있습니다.



학습문제로 공부한 내용을 정리해 봅시다.

[문제1] 다음 두 명령 블록을 실행했을 때 어떤 차이가 있는지 빈 칸에 작성해 봅시다.

시작하기 버튼을 클릭했을 때

계속 반복하기

만일 < 20 > 햄스터 0 : 왼쪽 바닥 센서 > 또는 < 20 > 햄스터 0 : 오른쪽 바닥 센서 > (이)라면

치즈 스틱 0 : HAT-010 배경 빨간색 X 울(를) x: 0 y: 0 에 그리기

치즈 스틱 0 : 버저 음을 1000 Hz로 정하기

아니면

치즈 스틱 0 : HAT-010 배경 지우기

치즈 스틱 0 : 소리 끄기

시작하기 버튼을 클릭했을 때

계속 반복하기

만일 < 20 > 햄스터 0 : 왼쪽 바닥 센서 > 그리고 < 20 > 햄스터 0 : 오른쪽 바닥 센서 > (이)라면

치즈 스틱 0 : HAT-010 배경 빨간색 X 울(를) x: 0 y: 0 에 그리기

치즈 스틱 0 : 버저 음을 1000 Hz로 정하기

아니면

치즈 스틱 0 : HAT-010 배경 지우기

치즈 스틱 0 : 소리 끄기

[문제2] 햄스터가 장애물을 감지하면 치즈스틱의 5x5 매트릭스 LED에 '싫음' 표정이 나타나도록 하려합니다. 빈 칸에 들어갈 내용을 작성해 봅시다.

▶ 시작하기 버튼을 클릭했을 때

계속 반복하기

만일 10 햄스터 0 : (이)라면

치즈 스틱 0 : HAT-010 배경 빨간색 싫음 음(를) x: 0 y: 0 에 그리기

아니면

치즈 스틱 0 : HAT-010 그림 1 지우기

[문제3] 지뢰를 탐지하거나 장애물이 있을 경우 치즈스틱에서 경고음이 나도록 할 때 필요한 블록은 무엇인지 선택해 봅시다.

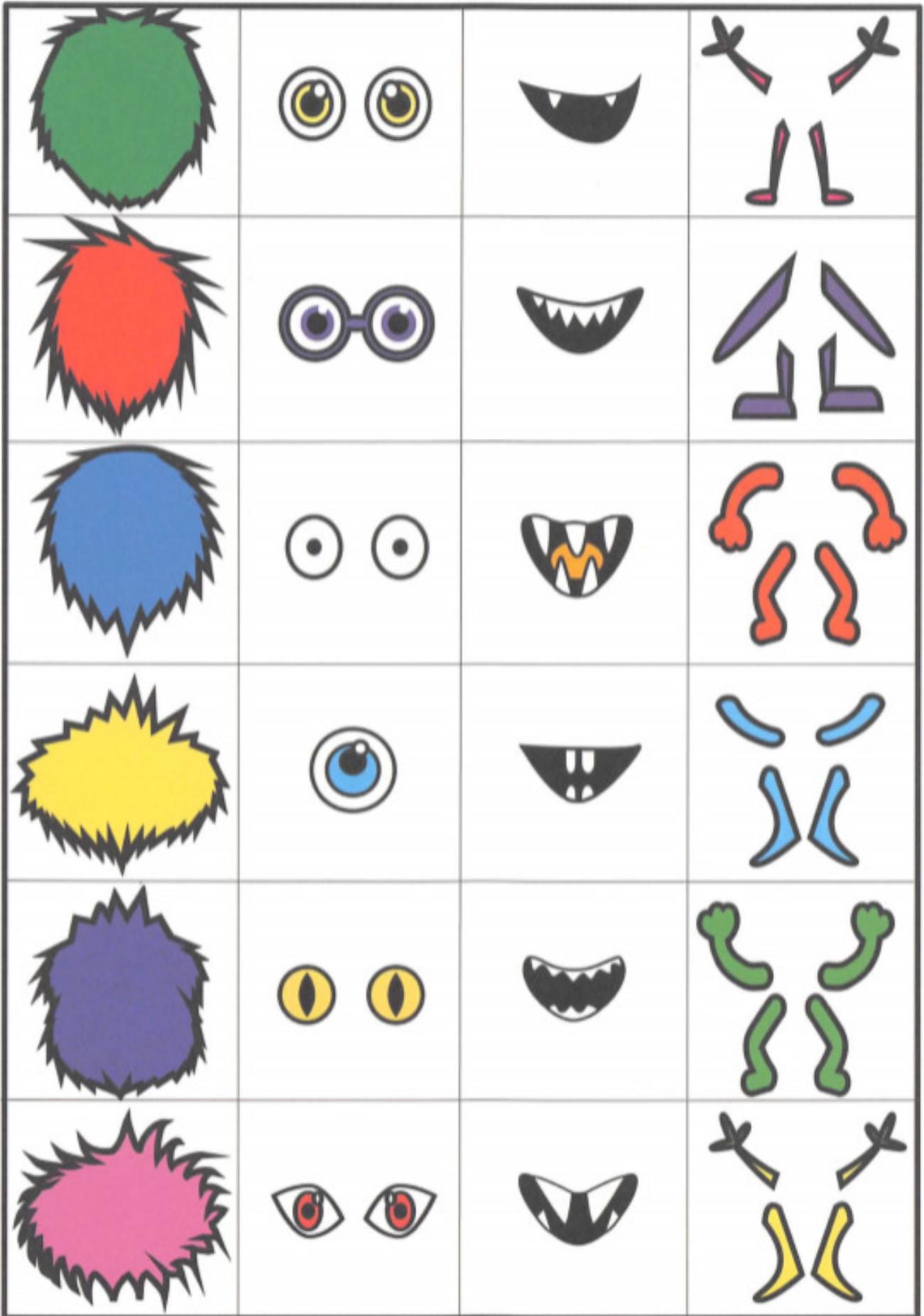
치즈 스틱 0 : 소리 출력을 내부 스피커로 정하기

치즈 스틱 0 : 버저 음을 1000 Hz로 정하기

치즈 스틱 0 : 연주 속도를 60 BPM으로 정하기

치즈 스틱 0 : 소리 끄기

<부록> 인형 꾸미기





발행인 문전일

발행처 한국로봇산업진흥원

주소 대구광역시 북구 노원로 77

문의 한국로봇산업진흥원 글로벌성장팀(053-210-9673)

홈페이지 www.kiria.org

이 책의 저작권은 한국로봇산업진흥원에 있습니다.

2020년도 로봇창의교육사업 창의교실

