

블록

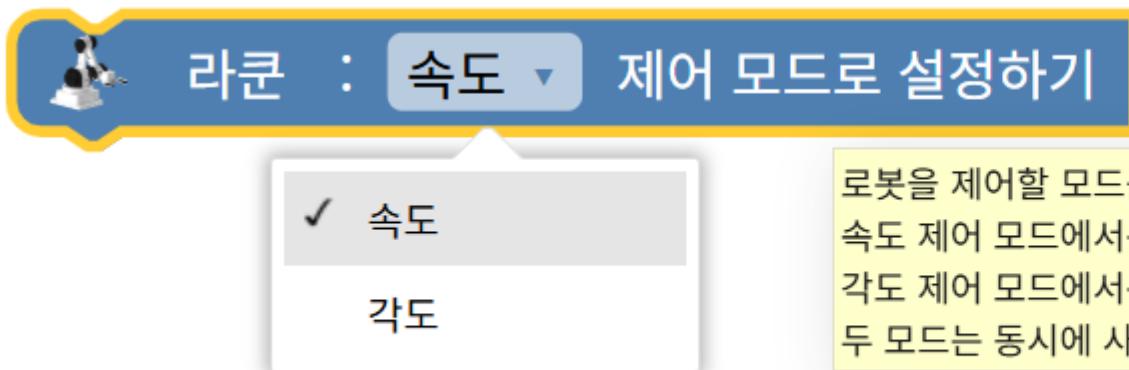
제어 모드 설정하기

라쿤봇을 제어할 모드를 설정합니다.

속도 제어 모드에서는 각 관절의 속도만을 통해 관절을 움직입니다.

각도 제어 모드에서는 각도 제어 속도, 각 관절 각도를 통해 제어하거나 목표 좌표를 통해 움직입니다.

두 모드는 동시에 사용할 수 없습니다.



드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
mode	드롭다운 옵션	라쿤봇 제어 모드	속도 (0), 각도 (1)

자바스크립트 코드

```
// 라쿤봇 속도 제어 모드로 설정하기
$('Raccoon4*0:mode').d = 0;

// 라쿤봇 각도 제어 모드로 설정하기
$('Raccoon4*0:mode').d = 1;
$('Raccoon4*0:angle.joints').d = [ $('Raccoon4*0:encoder.joint_1').d,
$('Raccoon4*0:encoder.joint_2').d, $('Raccoon4*0:encoder.joint_3').d,
$('Raccoon4*0:encoder.joint_4').d];
```

파이썬 코드

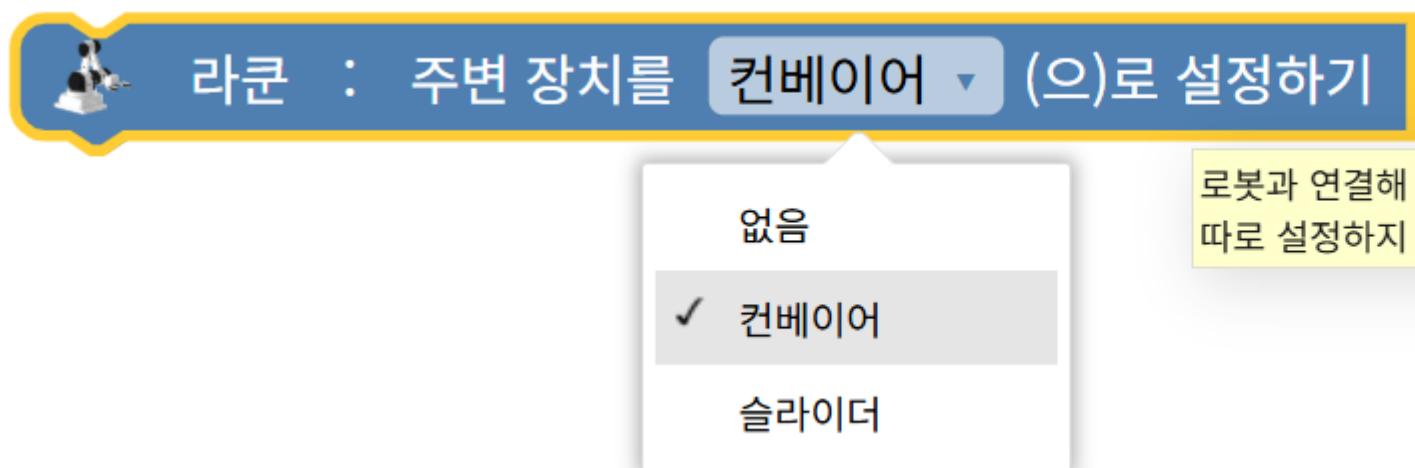
```
# 라쿤봇 속도 제어 모드로 설정하기
__('Raccoon4*0:mode').d = 0

# 라쿤봇 각도 제어 모드로 설정하기
__('Raccoon4*0:mode').d = 1
__('Raccoon4*0:angle.joints').d = [__('Raccoon4*0:encoder.joint_1').d,
__('Raccoon4*0:encoder.joint_2').d, __('Raccoon4*0:encoder.joint_3').d,
__('Raccoon4*0:encoder.joint_4').d]
```

주변 장치 설정하기

라쿤봇과 연결해 사용할 주변 장치를 설정합니다.

따로 설정하지 않아도, 연결된 주변 장치를 자동으로 인식하여 제어할 수 있습니다.



드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
peripheral	드롭다운 옵션	주변 장치	컨베이어 (1), 슬라이더 (2)

자바스크립트 코드

```
// 주변 장치를 컨베이어 (으)로 설정하기
$( 'Raccoon4*0:peripheral' ).d = 1;
```

```
// 주변 장치를 슬라이더 (으) 로 설정하기  
$('Raccoon4*0:peripheral>').d = 2;
```

파이썬 코드

```
# 주변 장치를 컨베이어 (으) 로 설정하기  
_(_('Raccoon4*0:peripheral>').d = 1  
  
# 주변 장치를 슬라이더 (으) 로 설정하기  
_(_('Raccoon4*0:peripheral>').d = 2
```

관절 모터 제어 켜기 / 끄기

각 관절 모터에 가해져 있는 제어를 풀지 유지할지 결정합니다.

설정하지 않을 경우, 모든 관절의 모터 제어가 켜진 상태로 시작합니다.



드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
motor	드롭다운 옵션	관절 번호	1(joint_1), 2(joint_2), 3(joint_3), 4(joint_4), 전 체 (joint_1, joint_2, joint_3, joint_4)
power	드롭다운 옵션	켜기/끄기	켜기 (0), 끄기 (1)

자바스크립트 코드

```
// 관절 1 모터 제어 켜기
$('Raccoon4*0:motor_off.joint_1').d = 0; // 켜기

// 관절 2 모터 제어 끄기
$('Raccoon4*0:motor_off.joint_2').d = 1; // 끄기

// 관절 3 모터 제어 켜기
$('Raccoon4*0:motor_off.joint_3').d = 0; // 켜기

// 관절 4 모터 제어 끄기
$('Raccoon4*0:motor_off.joint_4').d = 1; // 끄기

// 관절 전체 모터 제어 켜기
$('Raccoon4*0:motor_off.joint_1').d = 0; // 켜기
$('Raccoon4*0:motor_off.joint_2').d = 0; // 켜기
$('Raccoon4*0:motor_off.joint_3').d = 0; // 켜기
$('Raccoon4*0:motor_off.joint_4').d = 0; // 켜기
```

파이썬 코드

```
# 관절 1 모터 제어 켜기
__($('Raccoon4*0:motor_off.joint_1').d = 0 # 켜기

# 관절 2 모터 제어 끄기
```

```

__('Raccoon4*0:motor_off.joint_2').d = 1 # 끄기

# 관절 3 모터 제어 켜기
__('Raccoon4*0:motor_off.joint_3').d = 0 # 켜기

# 관절 4 모터 제어 끄기
__('Raccoon4*0:motor_off.joint_4').d = 1 # 끄기

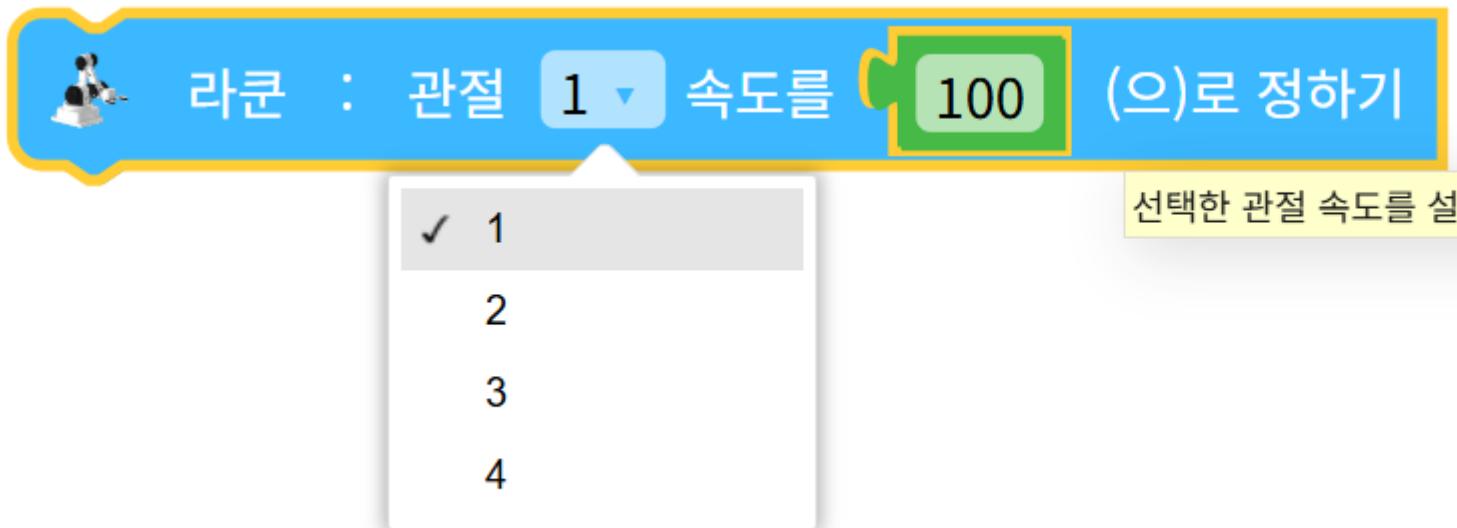
# 관절 전체 모터 제어 켜기
__('Raccoon4*0:motor_off.joint_1').d = 0 # 켜기
__('Raccoon4*0:motor_off.joint_2').d = 0 # 켜기
__('Raccoon4*0:motor_off.joint_3').d = 0 # 켜기
__('Raccoon4*0:motor_off.joint_4').d = 0 # 켜기

```

관절 속도 설정하기

선택한 관절 속도를 설정합니다.

관절 속도의 범위는 -100 ~ 100입니다.



드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
motor	드롭다운 옵션	관절 번호	1(joint_1), 2(joint_2), 3(joint_3), 4(joint_4)
speed	입력 값	관절 속도	-100 ~ 100 사이 실수

자바스크립트 코드

```
// 관절 1 속도를 100(으)로 정하기
$('Raccoon4*0:speed.joint_1').d = 100;

// 관절 2 속도를 90(으)로 정하기
$('Raccoon4*0:speed.joint_2').d = 90;

// 관절 3 속도를 80(으)로 정하기
$('Raccoon4*0:speed.joint_3').d = 80;

// 관절 4 속도를 70(으)로 정하기
$('Raccoon4*0:speed.joint_4').d = 70;
```

파이썬 코드

```
# 관절 1 속도를 100(으)로 정하기
__($('Raccoon4*0:speed.joint_1').d = 100

# 관절 2 속도를 90(으)로 정하기
__($('Raccoon4*0:speed.joint_2').d = 90

# 관절 3 속도를 80(으)로 정하기
__($('Raccoon4*0:speed.joint_3').d = 80

# 관절 4 속도를 70(으)로 정하기
__($('Raccoon4*0:speed.joint_4').d = 70
```

관절 속도 변경하기

선택한 관절의 속도를 입력값만큼 변경합니다.

입력값의 범위는 -200 ~ 200 입니다.



드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
motor	드롭다운 옵션	관절 번호	1(joint_1), 2(joint_2), 3(joint_3), 4(joint_4)
speed	입력 값	변경 속도	-200 ~ 200 사이 실수

자바스크립트 코드

```
// 관절 1 속도를 -200 만큼 바꾸기
$('Raccoon4*0:speed.joint_1').d = $('Raccoon4*0:speed.joint_1').d + -200;

// 관절 2 속도를 40 만큼 바꾸기
$('Raccoon4*0:speed.joint_2').d = $('Raccoon4*0:speed.joint_2').d + 40;

// 관절 3 속도를 30 만큼 바꾸기
$('Raccoon4*0:speed.joint_3').d = $('Raccoon4*0:speed.joint_3').d + 30;

// 관절 4 속도를 20 만큼 바꾸기
```

```
$('Raccoon4*0:speed.joint_4').d = $('Raccoon4*0:speed.joint_4').d + 20;
```

파이썬 코드

```
# 관절 1 속도를 -200 만큼 바꾸기  
__('Raccoon4*0:speed.joint_1').d = __('Raccoon4*0:speed.joint_1').d + (-200)  
  
# 관절 2 속도를 40 만큼 바꾸기  
__('Raccoon4*0:speed.joint_2').d = __('Raccoon4*0:speed.joint_2').d + 40  
  
# 관절 3 속도를 30 만큼 바꾸기  
__('Raccoon4*0:speed.joint_3').d = __('Raccoon4*0:speed.joint_3').d + 30  
  
# 관절 4 속도를 20 만큼 바꾸기  
__('Raccoon4*0:speed.joint_4').d = __('Raccoon4*0:speed.joint_4').d + 20
```

관절 속도 배열로 설정하기

네 관절의 속도를 한 번에 설정합니다.

각 관절 속도의 범위는 -100 ~ 100 입니다.



라쿤 : 각 관절의 속도를 **배열 [0, 0, 0, 0] (으)로**

네 관절의 속도를 한 번에 설정합니다. 각 관절 속도의 범위는 -100 ~ 100입니다.

드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
speed	입력 값	속도	-100 ~ 100 사이 실수 배열

자바스크립트 코드

```
// 각 관절의 속도를 [10, 20, 30, 40] (으)로 정하기
$('Raccoon4*0:speed.joint_1').d = 10;
$('Raccoon4*0:speed.joint_2').d = 20;
$('Raccoon4*0:speed.joint_3').d = 30;
$('Raccoon4*0:speed.joint_4').d = 40;
```

파이썬 코드

```
# 각 관절의 속도를 [10, 20, 30, 40] (으)로 정하기
__('Raccoon4*0:speed.joint_1').d = 10
__('Raccoon4*0:speed.joint_2').d = 20
__('Raccoon4*0:speed.joint_3').d = 30
__('Raccoon4*0:speed.joint_4').d = 40
```

관절 각도 제어 속도 설정하기

각도 제어 모드에서 관절들을 제어할 속도를 설정합니다.

속도의 범위는 0 ~ 100입니다.



라쿤 : 관절 각도 제어 속도를 100 (으)로 정하기

각도 제어 모드에서 관절들을 제어할 속도를 설정합니다

드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
speed	입력값	관절 제어 속도	0 ~ 100 실수

자바스크립트 코드

```
// 관절 각도 제어 속도를 100(으)로 정하기  
$( 'Raccoon4*0:angle.max_speed' ).d = 100;
```

파이썬 코드

```
# 관절 각도 제어 속도를 100(으)로 정하기  
__('Raccoon4*0:angle.max_speed').d = 100
```

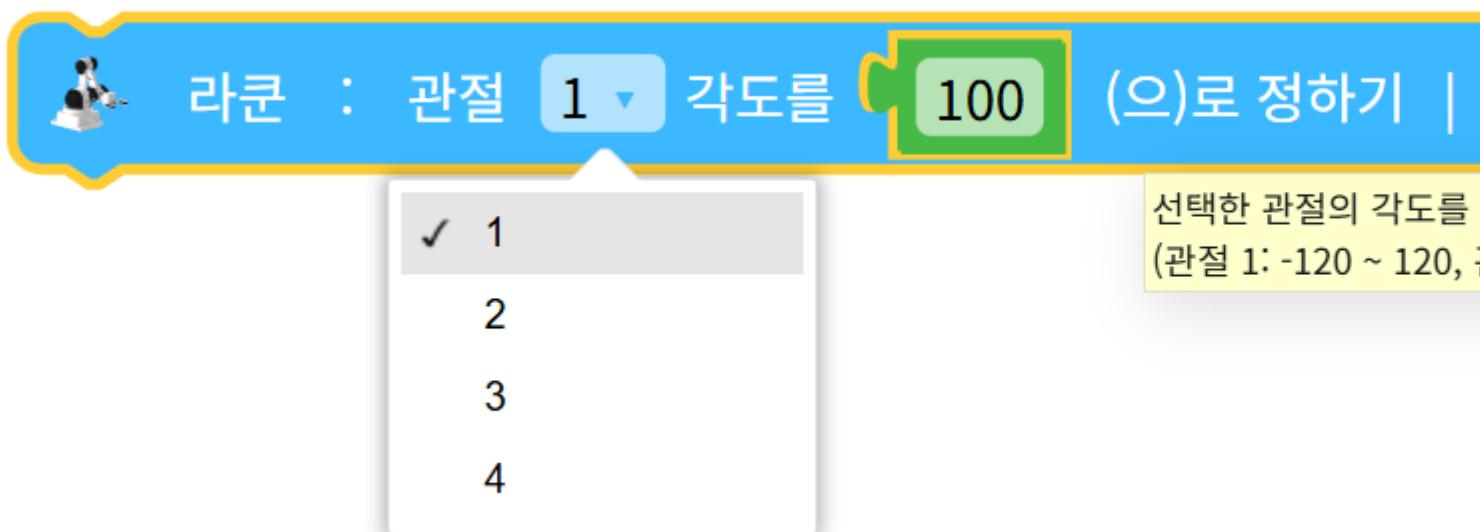
관절 각도 설정하기

선택한 관절의 각도를 설정합니다. 각 관절 각도의 범위는 다음과 같습니다.

관절 1 : -120 ~ 120, 관절 2 : -90 ~ 30, 관절 3 : -150 ~ 0, 관절 4 : -105 ~ 105

기다리기를 체크하면, 이동이 완료될 때까지 기다립니다.

단, 기다리기를 체크한 경우에는 async 함수 내에서만 사용할 수 있습니다.



드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
motor	드롭다운 옵션	관절 번호	1(joint_1), 2(joint_2), 3(joint_3), 4(joint_4)

이름	구분	설명	범위 / 종류
angle	입력 값	관절 각도	관절 1 : -120 ~ 120, 관절 2 : -90 ~ 30, 관절 3 : -150 ~ 0, 관절 4 : -105 ~ 105

자바스크립트 코드

```
// 관절 1 각도를 100 (으)로 정하기 | 기다리기 0
$('Raccoon4*0:angle.joints').d = $('Raccoon4*0:angle.joints').d.map((data, i) => (i + 1 === 1) ? 100 : Math.floor(data));
await $('Raccoon4*0:angle.!joints').w();

// 관절 2 각도를 30 (으)로 정하기 | 기다리기 0
$('Raccoon4*0:angle.joints').d = $('Raccoon4*0:angle.joints').d.map((data, i) => (i + 1 === 2) ? 30 : Math.floor(data));
await $('Raccoon4*0:angle.!joints').w();

// 관절 3 각도를 0 (으)로 정하기 | 기다리기 X
$('Raccoon4*0:angle.joints').d = $('Raccoon4*0:angle.joints').d.map((data, i) => (i + 1 === 3) ? 0 : Math.floor(data));

// 관절 4 각도를 100 (으)로 정하기 | 기다리기 X
$('Raccoon4*0:angle.joints').d = $('Raccoon4*0:angle.joints').d.map((data, i) => (i + 1 === 4) ? 100 : Math.floor(data));
```

파이썬 코드

```
# 관절 1 각도를 100 (으)로 정하기 | 기다리기 0
__('Raccoon4*0:angle.joints').d = [100 if i + 1 == 1 else __('Raccoon4*0:angle.joints').d[i] for i in range(4)]
await __('Raccoon4*0:angle.!joints').w()

# 관절 2 각도를 30 (으)로 정하기 | 기다리기 0
```

```

__('Raccoon4*0:angle.joints').d = [30 if i + 1 == 2 else __('Raccoon4*0:angle.joints').d[i] for i
in range(4)]
await __('Raccoon4*0:angle.!joints').w()

# 관절 3 각도를 0 (으)로 정하기 | 기다리기 X
__('Raccoon4*0:angle.joints').d = [0 if i + 1 == 3 else __('Raccoon4*0:angle.joints').d[i] for i
in range(4)]

# 관절 4 각도를 100 (으)로 정하기 | 기다리기 X
__('Raccoon4*0:angle.joints').d = [100 if i + 1 == 4 else __('Raccoon4*0:angle.joints').d[i] for
i in range(4)]

```

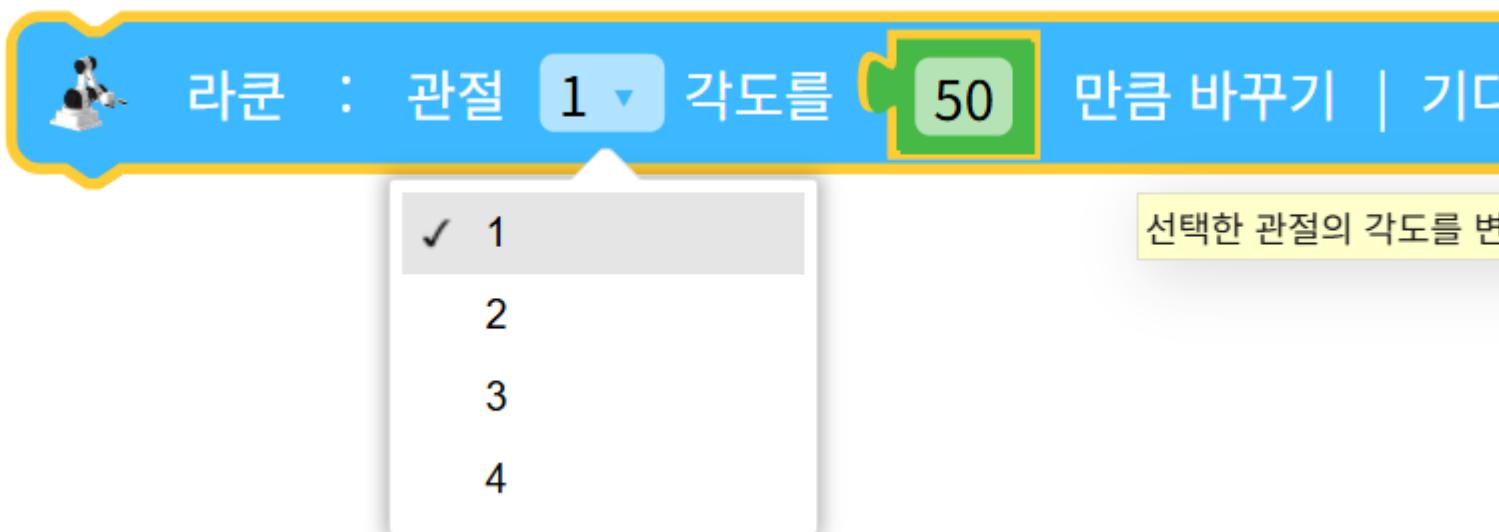
관절 각도 변경하기

선택한 관절의 각도를 입력값만큼 변경합니다. 각 입력값의 범위는 다음과 같습니다.

관절 1 : -240 ~ 240, 관절 2 : -120 ~ 120, 관절 3 : -150 ~ 150, 관절 4 : -210 ~ 210

기다리기를 체크하면, 이동이 완료될 때까지 기다립니다.

단, 기다리기를 체크한 경우에는 `async` 함수 내에서만 사용할 수 있습니다.



드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
motor	드롭다운 옵션	관절 번호	1(joint_1), 2(joint_2), 3(joint_3), 4(joint_4)
angle	입력 값	관절 변경값	관절 1 : -240 ~ 240, 관절 2 : -120 ~ 120, 관절 3 : -150 ~ 150, 관절 4 : -210 ~ 210

자바스크립트 코드

```
// 관절 1 각도를 50 만큼 바꾸기 | 기다리기 0
$('Raccoon4*0:angle.joints').d = $('Raccoon4*0:angle.joints').d.map((data, i) => (i + 1 === 1) ?
data + 50 : data);
await $('Raccoon4*0:angle.!joints').w();

// 관절 2 각도를 40 만큼 바꾸기 | 기다리기 0
$('Raccoon4*0:angle.joints').d = $('Raccoon4*0:angle.joints').d.map((data, i) => (i + 1 === 2) ?
data + 40 : data);
await $('Raccoon4*0:angle.!joints').w();

// 관절 3 각도를 30 만큼 바꾸기 | 기다리기 X
$('Raccoon4*0:angle.joints').d = $('Raccoon4*0:angle.joints').d.map((data, i) => (i + 1 === 3) ?
data + 30 : data);

// 관절 4 각도를 20 만큼 바꾸기 | 기다리기 X
$('Raccoon4*0:angle.joints').d = $('Raccoon4*0:angle.joints').d.map((data, i) => (i + 1 === 4) ?
data + 20 : data);
```

파이썬 코드

```
# 관절 1 각도를 50 만큼 바꾸기 | 기다리기 0
__('Raccoon4*0:angle.joints').d = __('Raccoon4*0:angle.joints').d[i] + 50 if i + 1 == 1 else
__('Raccoon4*0:angle.joints').d[i] for i in range(4)
await __('Raccoon4*0:angle.!joints').w()
```

```

# 관절 2 각도를 40 만큼 바꾸기 | 기다리기 0
__('Raccoon4*0:angle.joints').d = [__(('Raccoon4*0:angle.joints').d[i] + 40 if i + 1 == 2 else
__('Raccoon4*0:angle.joints').d[i] for i in range(4)]
await __('Raccoon4*0:angle.!joints').w()

# 관절 3 각도를 30 만큼 바꾸기 | 기다리기 X
__('Raccoon4*0:angle.joints').d = [__(('Raccoon4*0:angle.joints').d[i] + 30 if i + 1 == 3 else
__('Raccoon4*0:angle.joints').d[i] for i in range(4)]

# 관절 4 각도를 20 만큼 바꾸기 | 기다리기 X
__('Raccoon4*0:angle.joints').d = [__(('Raccoon4*0:angle.joints').d[i] + 20 if i + 1 == 4 else
__('Raccoon4*0:angle.joints').d[i] for i in range(4)]

```

관절 각도 배열로 설정하기

네 관절의 각도를 한 번에 설정합니다. 각 관절 각도의 범위는 다음과 같습니다.

관절 1 : -120 ~ 120, 관절 2 : -90 ~ 30, 관절 3 : -150 ~ 0, 관절 4 : -105 ~ 105

기다리기를 체크하면, 이동이 완료될 때까지 기다립니다.

단, 기다리기를 체크한 경우에는 `async` 함수 내에서만 사용할 수 있습니다.



라쿤 : 각 관절의 각도를 **배열** **[0, 0, 0, 0]** (으)로

네 관절의 각도를 한 번에 설정합니다. 각 관절 각도의 범위는
(관절 1: -120 ~ 120, 관절 2: -90 ~ 30, 관절 3: -150 ~ 0, 관

드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
angle	입력 값	각도	관절 1 : -120 ~ 120, 관절 2 : -90 ~ 30, 관절 3 : -150 ~ 0, 관절 4 : -105 ~ 105

자바스크립트 코드

```
// 각 관절의 각도를 [0, 0, 0, 0] (으)로 정하기 | 기다리기 X
$('Raccoon4*0:angle.joints').d = [0,0,0,0];

// 각 관절의 각도를 [10, 10, -10, 10] (으)로 정하기 | 기다리기 0
$('Raccoon4*0:angle.joints').d = [10,10,-10,10];
await $('Raccoon4*0:angle.!joints').w();
```

파이썬 코드

```
# 각 관절의 각도를 [0, 0, 0, 0] (으)로 정하기 | 기다리기 X
__($('Raccoon4*0:angle.joints').d = [0,0,0,0]

# 각 관절의 각도를 [10, 10, -10, 10] (으)로 정하기 | 기다리기 0
__($('Raccoon4*0:angle.joints').d = [10,10,-10,10]
await __($('Raccoon4*0:angle.!joints').w()
```

XYZ 좌표로 이동하기

로봇 팔을 이동시킬 x, y, z 좌표를 설정합니다.

각 좌표의 범위는 다음과 같습니다.

손목 기준 => x : -200mm ~ 200mm, y : -100mm ~ 200mm, z : -20mm ~ 280mm

말단 장치 기준 => x : -280mm ~ 280mm, y : -180mm ~ 280mm, z : -100mm ~ 360mm

말단 장치 기준으로 제어할 경우, 연결된 장치에 따라 범위가 달라질 수 있습니다.

말단 장치의 방향은 지면과 수평 혹은 수직으로 고정됩니다.

이동할 수 없는 좌표를 입력했을 경우, 명령을 건너뜁니다.

기다리기를 체크하면, 이동이 완료될 때까지 기다립니다.

단, 기다리기를 체크한 경우에는 async 함수 내에서만 사용할 수 있습니다.



라쿤 : 손목 ▾ 기준

배열 [0, 100, 100]

xyz 좌

✓ 손목

말단 장치

로봇 팔을
각 좌표의
손목 기준
말단 장치
말단 장치
말단 장치
이동할 수

드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
pivot	드롭다운 옵션	좌표 기준	말단장치 (end_effector), 손목 (wrist)
coordinate	입력 값	이동 좌표 x,y,z	실수 배열

자바스크립트 코드

```
// 말단장치 기준 [0, 100, 100] x,y,z 좌표로 이동하기 | 기다리기 0
$('Raccoon4*0:angle.joints').d = __xyz_to_angles('Raccoon4*0', [0,100,100], 'end_effector');
await $('Raccoon4*0:angle.!joints').w();

// 손목 기준 [0, 100, 100] x,y,z 좌표로 이동하기 | 기다리기 X
$('Raccoon4*0:angle.joints').d = __xyz_to_angles('Raccoon4*0', [0,100,100], 'wrist');
```

파이썬 코드

```
# 말단장치 기준 [0, 100, 100] x,y,z 좌표로 이동하기 | 기다리기 0
__('Raccoon4*0:angle.joints').d = __xyz_to_angles('Raccoon4*0', [0,100,100], 'end_effector')
await __('Raccoon4*0:angle.!joints').w()
```

```
# 손목 기준 [0, 100, 100] x,y,z 좌표로 이동하기 | 기다리기 X
__('Raccoon4*0:angle.joints').d = __xyz_to_angles('Raccoon4*0', [0,100,100], 'wrist')
```

관절 상태 초기화 하기

네 관절의 각도를 선택한 기본 옵션으로 설정합니다.

기본 옵션의 각도는 다음과 같습니다.

zero : [0,0,0,0], park : [0,25,-145,-60], home : [0,-10,-140,60]

기다리기를 체크하면, 이동이 완료될 때까지 기다립니다.

단, 기다리기를 체크한 경우에는 async 함수 내에서만 사용할 수 있습니다.



라쿤 : 관절 상태를 zero (으)로 초기화하기 | 기다리기

- zero
- park
- home

네 관절의 각도를 선택한 기본
(zero: [0,0,0,0], park: [0,25,-145,-60])

드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
option	드롭다운 옵션	기본 옵션	zero, park, home

자바스크립트 코드

```
// 관절 상태 zero(으)로 초기화하기 | 기다리기 0
$('Raccoon4*0:angle.joints').d = [0, 0, 0, 0];
await $('Raccoon4*0:angle.!joints').w();
```

```
// 관절 상태 park(으)로 초기화하기 | 기다리기 X
$('Raccoon4*0:angle.joints').d = [0, 25, -145, -60];
```

```
// 관절 상태 home(으)로 초기화하기 | 기다리기 X
$('Raccoon4*0:angle.joints').d = [0, -10, -140, 60];
```

파이썬 코드

```
# 관절 상태 zero(으)로 초기화하기 | 기다리기 0
__('Raccoon4*0:angle.joints').d = [0, 0, 0, 0]
await __('Raccoon4*0:angle.!joints').w()

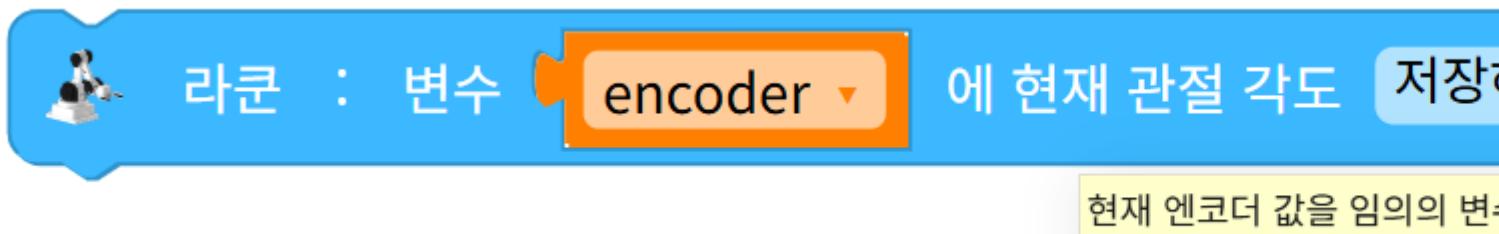
# 관절 상태 park(으)로 초기화하기 | 기다리기 X
__('Raccoon4*0:angle.joints').d = [0, 25, -145, -60]

# 관절 상태 home(으)로 초기화하기 | 기다리기 X
__('Raccoon4*0:angle.joints').d = [0, -10, -140, 60]
```

관절 각도 저장하기

저장하기 버튼을 눌러 지정한 변수에 현재 관절 각도 값을 저장합니다.

또한, 현재 관절 각도 값을 배열로 동시에 받습니다.



드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
variable	입력값	관절 값을 받을 변수	Block composer 변수

자바스크립트 코드

```
// 현재 각도 값 [0,-10,-140,60] 일 때, 저장하기 클릭 시  
var encoder;  
encoder = [0,-10,-140,60];
```

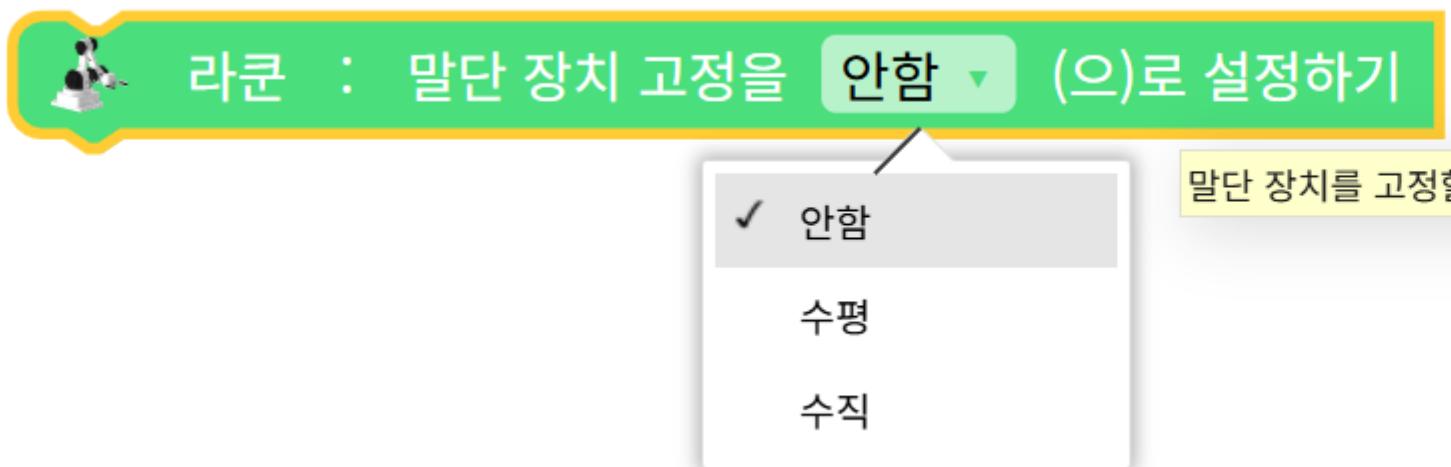
파이썬 코드

```
# 현재 각도 값 [0,-10,-140,60] 일 때, 저장하기 클릭 시  
encoder = None  
encoder = [0,-10,-140,60]
```

말단 장치 고정 설정하기

말단 장치를 고정할 방향을 설정합니다.

옵션이 수평 혹은 수직으로 설정된 이후로는, 관절 4의 속도나 각도를 제어할 수 없습니다.



드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
fix	드롭다운 옵션	고정 설정값	안함 (0), 수평 (1), 수직 (2)

자바스크립트 코드

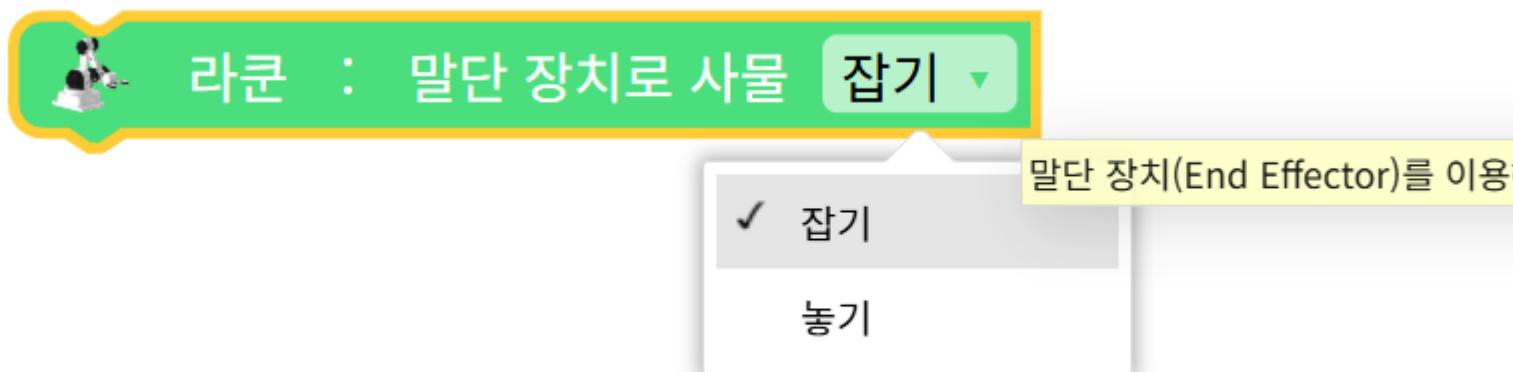
```
// 말단 장치 고정을 수평 (으) 로 설정하기  
$('Raccoon4*0:end_effector.lock').d = 1;  
  
// 말단 장치 고정을 수직 (으) 로 설정하기  
$('Raccoon4*0:end_effector.lock').d = 2;  
  
// 말단 장치 고정을 안함 (으) 로 설정하기  
$('Raccoon4*0:end_effector.lock').d = 0;
```

파이썬 코드

```
# 말단 장치 고정을 수평 (으) 로 설정하기  
__('Raccoon4*0:end_effector.lock').d = 1  
  
# 말단 장치 고정을 수직 (으) 로 설정하기  
__('Raccoon4*0:end_effector.lock').d = 2  
  
# 말단 장치 고정을 안함 (으) 로 설정하기  
__('Raccoon4*0:end_effector.lock').d = 0
```

말단 장치로 사물 잡기 / 놓기

말단 장치를 이용하여 물건을 잡거나 놓습니다.



드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
toggle	드롭다운 옵션	말단 장치 사용 여부	잡기 (1), 놓기 (0)

자바스크립트 코드

```
// 말단 장치로 사물 잡기
$( 'Raccoon4*0:end_effector.control' ).d = 1;

// 말단 장치로 사물 놓기
$( 'Raccoon4*0:end_effector.control' ).d = 0;
```

파이썬 코드

```
# 말단 장치로 사물 잡기
__('Raccoon4*0:end_effector.control').d = 1

# 말단 장치로 사물 놓기
__('Raccoon4*0:end_effector.control').d = 0
```

말단 장치 값

현재 연결되어 있는 말단 장치 정보에 따라 숫자를 반환합니다.

통합 그리퍼 일 시 1, 진동 그리퍼 일 시 2, 서보 그리퍼 일 시 3, DC 그리퍼 일 시 4 를 반환합니다.



현재 연결되어 있는 말단 장치 번호
(1, 3, 4: 집게 그리퍼, 2: 진공 그리퍼)

자바스크립트 코드

```
$('Raccoon4*0:end_effector.device').d; // 말단 장치 정보 값
```

파이썬 코드

```
_('Raccoon4*0:end_effector.device').d # 말단 장치 정보 값
```

말단 장치 상태

말단 장치가 사물을 잡고 있는 상태에 따라 숫자를 반환합니다.

놓은 상태일 시 0, 잡은 상태일 시 1을 반환합니다.



라쿤 : 말단 장치 상태

말단 장치가 사물을 잡고 있는 상태
(0: 놓은 상태, 1: 잡은 상태)

자바스크립트 코드

```
$('Raccoon4*0:end_effector.status').d; // 말단 장치 상태 값
```

파이썬 코드

```
_('Raccoon4*0:end_effector.status').d # 말단 장치 상태 값
```

음계 연주하기

라쿤봇이 지정된 음계를 재생합니다.



- 1
- 2
- 3
- ✓ 4
- 5
- 6
- 7

드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
note	드롭다운 옵션	음계	도 (Do), 도 #(Do#), 레 (Re), 레 #(Re#), 미 (Mi), 파 (Fa), 파 #(Fa#), 솔 (So), 솔 #(So#), 라 (La), 라 #(La#), 시 (Ti)
octave	드롭다운 옵션	옥타브	1 ~ 7

자바스크립트 코드

```
// 1 옥타브 도 (Do) 음을 연주하기
$('Raccoon4*0:sound.note').d = 4;
```

```
// 1 옥타브 레 (Re) 음을 연주하기  
$('Raccoon4*0:sound.note').d = 6;
```

```
// 2 옥타브 도 (Do) 음을 연주하기  
$('Raccoon4*0:sound.note').d = 16;
```

```
// 7 옥타브 시 (Ti) 음을 연주하기  
$('Raccoon4*0:sound.note').d = 87;
```

파이썬 코드

```
# 1 옥타브 도 (Do) 음을 연주하기  
_('_Raccoon4*0:sound.note').d = 4
```

```
# 1 옥타브 레 (Re) 음을 연주하기  
_('_Raccoon4*0:sound.note').d = 6
```

```
# 2 옥타브 도 (Do) 음을 연주하기  
_('_Raccoon4*0:sound.note').d = 16
```

```
# 7 옥타브 시 (Ti) 음을 연주하기  
_('_Raccoon4*0:sound.note').d = 87
```

소리 재생하기

라쿤봇이 특정 사운드 클립을 재생합니다.

기다리기를 체크하면, 재생이 완료될 때까지 기다립니다.

단, 기다리기를 체크한 경우에는 `async` 함수 내에서만 사용할 수 있습니다.



라쿤 :

beep

소리 재생하기

기다리기



✓ beep

beep2

beep3

beep_repeat

beep_random

beep_random_repeat

snore

snore_repeat

siren

siren_repeat

engine

특정 사운드 클립을 재생합니다.

기다리기를 체크하면, 재생이 완료될 때까

드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
sound_clip	드롭다운 옵션	사운드 클립	beep(1), beep2(2), beep3(3), beep_repeat(4), beep_random(5), beep_random_repeat(6), snore(7), snore_repeat(8), siren(9), siren_repeat(10), engine(11), engine_repeat(12), fart_a(13), fart_b(14), noise(15), noise_repeat(16), whistle(17), chop_chop(18), chop_chop_repeat(19), random(20), r2d2(21), connected(22), dee_bee(33), finish(34), power_on(35), start(36), power_off(37)

자바스크립트 코드

```
// finish(34) 소리 재생하기 | 기다리기 0
$('Raccoon4*0:sound.clip').d = 34;
await $('Raccoon4*0:sound.!clip').w();

// power_on(35) 소리 재생하기 | 기다리기 X
$('Raccoon4*0:sound.clip').d = 35;
```

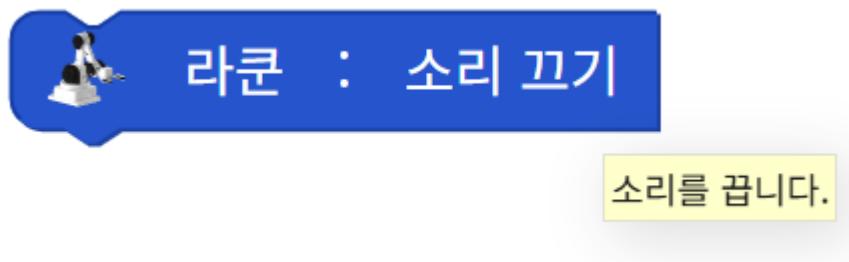
파이썬 코드

```
# finish(34) 소리 재생하기 | 기다리기 0
__('Raccoon4*0:sound.clip').d = 34
await __('Raccoon4*0:sound.!clip').w()

# power_on(35) 소리 재생하기 | 기다리기 X
__('Raccoon4*0:sound.clip').d = 35
```

소리 끄기

라쿤봇의 소리를 끕니다.



자바스크립트 코드

```
// 소리 끄기
__stopSound('Raccoon4*0');
```

파이썬 코드

```
# 소리 끄기
__stopSound('Raccoon4*0')
```

관절 엔코더 값

선택한 관절의 엔코더 값을 반환합니다.



라쿤 : 관절 1 엔코더 값

✓ 1

2

3

4

전체

선택한 관절의 엔코더 값

드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
motor	드롭다운 옵션	관절 번호	1(joint_1), 2(joint_2), 3(joint_3), 4(joint_4), 전 체 (joint_1, joint_2, joint_3, joint_4)

자바스크립트 코드

```
// 관절 1 엔코더 값
$('Raccoon4*0:encoder.joint_1').d;

// 관절 2 엔코더 값
$('Raccoon4*0:encoder.joint_2').d;

// 관절 3 엔코더 값
$('Raccoon4*0:encoder.joint_3').d;

// 관절 4 엔코더 값
$('Raccoon4*0:encoder.joint_4').d;
```

```
// 관절 전체 엔코더 값  
[$('Raccoon4*0:encoder.joint_1').d, $('Raccoon4*0:encoder.joint_2').d,  
$('Raccoon4*0:encoder.joint_3').d, $('Raccoon4*0:encoder.joint_4').d];
```

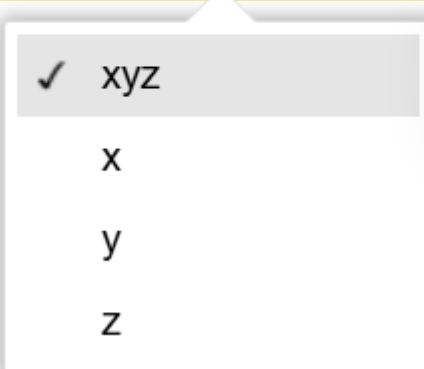
파이썬 코드

```
# 관절 1 엔코더 값  
__($('Raccoon4*0:encoder.joint_1').d  
  
# 관절 2 엔코더 값  
__($('Raccoon4*0:encoder.joint_2').d  
  
# 관절 3 엔코더 값  
__($('Raccoon4*0:encoder.joint_3').d  
  
# 관절 4 엔코더 값  
__($('Raccoon4*0:encoder.joint_4').d  
  
# 관절 전체 엔코더 값  
[__(($('Raccoon4*0:encoder.joint_1')).d, __($('Raccoon4*0:encoder.joint_2')).d,  
__(($('Raccoon4*0:encoder.joint_3')).d, __($('Raccoon4*0:encoder.joint_4')).d]
```

선택한 요소의 좌표

선택한 요소의 현재 좌표를 반환합니다.

연결된 말단 장치가 없을 경우, 손목 기준 위치가 반환됩니다.



선택한 요소의 현재 위치
연결된 말단 장치가 없을

드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
pivot	드롭다운 옵션	선택 요소	손목 (wrist), 말단 장치 (end_effector)
coordinate	드롭다운 옵션	좌표 축	xyz, x, y, z

자바스크립트 코드

```
// 손목 기준 xyz 좌표
_angles_to_xyz('Raccoon4*0', [$( 'Raccoon4*0:encoder.joint_1').d,
$( 'Raccoon4*0:encoder.joint_2').d, $( 'Raccoon4*0:encoder.joint_3').d,
$( 'Raccoon4*0:encoder.joint_4').d], 'wrist');
```

```
// 손목 기준 x 좌표
_angles_to_xyz('Raccoon4*0', [$( 'Raccoon4*0:encoder.joint_1').d,
$( 'Raccoon4*0:encoder.joint_2').d, $( 'Raccoon4*0:encoder.joint_3').d,
$( 'Raccoon4*0:encoder.joint_4').d], 'wrist')[0];
```

```
// 손목 기준 y 좌표
_angles_to_xyz('Raccoon4*0', [$( 'Raccoon4*0:encoder.joint_1').d,
$( 'Raccoon4*0:encoder.joint_2').d, $( 'Raccoon4*0:encoder.joint_3').d,
$( 'Raccoon4*0:encoder.joint_4').d], 'wrist')[1] 30
```

```

// 손목 기준 z 좌표
__angles_to_xyz('Raccoon4*0', [$( 'Raccoon4*0:encoder.joint_1').d,
$( 'Raccoon4*0:encoder.joint_2').d, $( 'Raccoon4*0:encoder.joint_3').d,
$( 'Raccoon4*0:encoder.joint_4').d], 'wrist')[2];

// 말단 장치 기준 xyz 좌표
__angles_to_xyz('Raccoon4*0', [$( 'Raccoon4*0:encoder.joint_1').d,
$( 'Raccoon4*0:encoder.joint_2').d, $( 'Raccoon4*0:encoder.joint_3').d,
$( 'Raccoon4*0:encoder.joint_4').d], 'end_effector');

// 말단 장치 기준 x 좌표
__angles_to_xyz('Raccoon4*0', [$( 'Raccoon4*0:encoder.joint_1').d,
$( 'Raccoon4*0:encoder.joint_2').d, $( 'Raccoon4*0:encoder.joint_3').d,
$( 'Raccoon4*0:encoder.joint_4').d], 'end_effector')[0];

// 말단 장치 기준 y 좌표
__angles_to_xyz('Raccoon4*0', [$( 'Raccoon4*0:encoder.joint_1').d,
$( 'Raccoon4*0:encoder.joint_2').d, $( 'Raccoon4*0:encoder.joint_3').d,
$( 'Raccoon4*0:encoder.joint_4').d], 'end_effector')[1];

// 말단 장치 기준 z 좌표
__angles_to_xyz('Raccoon4*0', [$( 'Raccoon4*0:encoder.joint_1').d,
$( 'Raccoon4*0:encoder.joint_2').d, $( 'Raccoon4*0:encoder.joint_3').d,
$( 'Raccoon4*0:encoder.joint_4').d], 'end_effector')[2];

```

파이썬 코드

```

# 손목 기준 xyz 좌표
__angles_to_xyz('Raccoon4*0', [__( 'Raccoon4*0:encoder.joint_1').d,
__( 'Raccoon4*0:encoder.joint_2').d, __('Raccoon4*0:encoder.joint_3').d,
__( 'Raccoon4*0:encoder.joint_4').d], 'wrist')

# 손목 기준 x 좌표

```

```
_angles_to_xyz('Raccoon4*0', [__('Raccoon4*0:encoder.joint_1').d,  
__('Raccoon4*0:encoder.joint_2').d, __(Raccoon4*0:encoder.joint_3').d,  
__(Raccoon4*0:encoder.joint_4).d], 'wrist')[0]
```

손목 기준 y 좌표

```
_angles_to_xyz('Raccoon4*0', [__('Raccoon4*0:encoder.joint_1').d,  
__('Raccoon4*0:encoder.joint_2').d, __(Raccoon4*0:encoder.joint_3').d,  
__(Raccoon4*0:encoder.joint_4).d], 'wrist')[1]
```

손목 기준 z 좌표

```
_angles_to_xyz('Raccoon4*0', [__('Raccoon4*0:encoder.joint_1').d,  
__('Raccoon4*0:encoder.joint_2').d, __(Raccoon4*0:encoder.joint_3').d,  
__(Raccoon4*0:encoder.joint_4).d], 'wrist')[2]
```

말단 장치 기준 xyz 좌표

```
_angles_to_xyz('Raccoon4*0', [__('Raccoon4*0:encoder.joint_1').d,  
__('Raccoon4*0:encoder.joint_2').d, __(Raccoon4*0:encoder.joint_3').d,  
__(Raccoon4*0:encoder.joint_4).d], 'end_effector')
```

말단 장치 기준 x 좌표

```
_angles_to_xyz('Raccoon4*0', [__('Raccoon4*0:encoder.joint_1').d,  
__('Raccoon4*0:encoder.joint_2').d, __(Raccoon4*0:encoder.joint_3').d,  
__(Raccoon4*0:encoder.joint_4).d], 'end_effector')[0]
```

말단 장치 기준 y 좌표

```
_angles_to_xyz('Raccoon4*0', [__('Raccoon4*0:encoder.joint_1').d,  
__('Raccoon4*0:encoder.joint_2').d, __(Raccoon4*0:encoder.joint_3').d,  
__(Raccoon4*0:encoder.joint_4).d], 'end_effector')[1]
```

말단 장치 기준 z 좌표

```
_angles_to_xyz('Raccoon4*0', [__('Raccoon4*0:encoder.joint_1').d,  
__('Raccoon4*0:encoder.joint_2').d, __(Raccoon4*0:encoder.joint_3').d,  
__(Raccoon4*0:encoder.joint_4).d], 'end_effector')[2]
```

신호 세기 값

라쿤봇의 신호 세기를 반환합니다.



라쿤 : 신호 세기

신호 세기

자바스크립트 코드

```
// 라쿤봇 신호 세기  
$('Raccoon4*0:signal_strength').d;
```

파이썬 코드

```
# 라쿤봇 신호 세기  
_(_('Raccoon4*0:signal_strength')).d
```

배터리 충전 상태 값

라쿤봇의 배터리 충전 상태 값을 반환합니다.



라쿤 : 배터리

배터리의 충전 상태 값

자바스크립트 코드

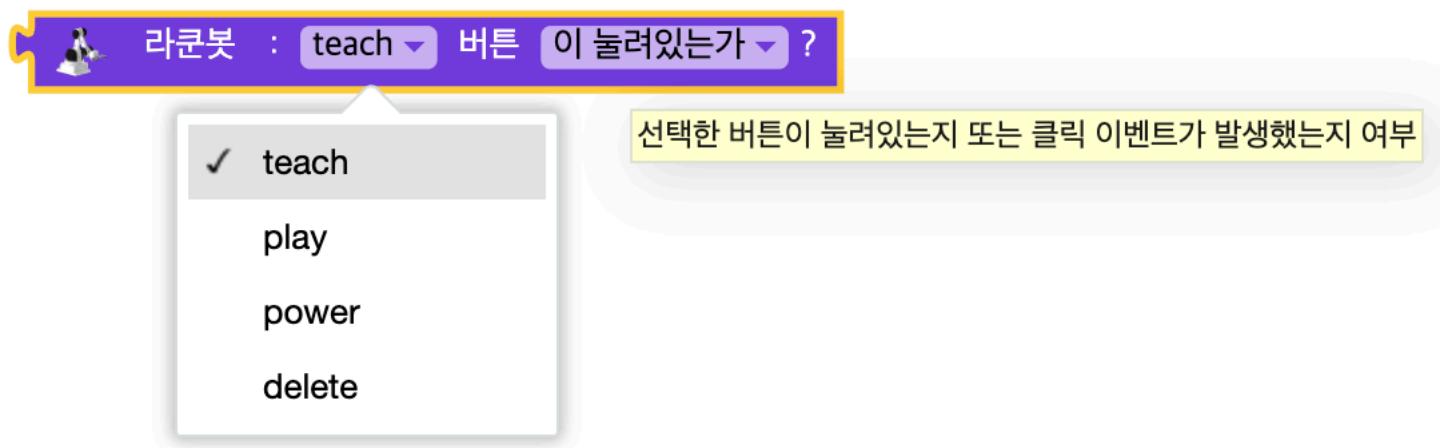
```
// 라쿤봇 배터리 충전 상태 값  
$('Raccoon4*0:battery.level').d;
```

파이썬 코드

```
# 라쿤봇 배터리 충전 상태 값  
_('_Raccoon4*0:battery.level').d
```

버튼이 눌려있는가?

선택한 버튼이 눌려있는지 또는 클릭 이벤트가 발생했는지 여부를 **참 (1) / 거짓 (0) (으)**로 반환합니다.
단, power 버튼을 길게 클릭시에는 전원 ON/OFF 이므로 길게 클릭했는지 여부를 확인할 수 없습니다.



드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
button	드롭다운 옵션	버튼 종류	teach, play, power, delete
state	드롭다운 옵션	클릭 이벤트	눌려있는가 (pressed), 클릭했는가 (click), 길게 클릭했는가 (long_click)

자바스크립트 코드

```
// teach 버튼이 눌려있는가?  
$('Raccoon4*0:button.pressed.teach').d;  
// teach 버튼을 클릭했는가?  
$('Raccoon4*0:button.click.teach').e;
```

```

// teach 버튼을 길게 클릭했는가?
$('Raccoon4*0:button.long_click.teach').e;

// play 버튼이 눌려있는가?
$('Raccoon4*0:button.pressed.play').d;
// play 버튼을 클릭했는가?
$('Raccoon4*0:button.click.play').e;
// play 버튼을 길게 클릭했는가?
$('Raccoon4*0:button.long_click.play').e;

// power 버튼이 눌려있는가?
$('Raccoon4*0:button.pressed.power').d;
// power 버튼을 클릭했는가?
$('Raccoon4*0:button.click.power').e;
// power 버튼을 길게 클릭했는가? -> X

// delete 버튼이 눌려있는가?
$('Raccoon4*0:button.pressed.delete').d;
// delete 버튼을 클릭했는가?
$('Raccoon4*0:button.click.delete').e;
// delete 버튼을 길게 클릭했는가?
$('Raccoon4*0:button.long_click.delete').e;

```

파이썬 코드

```

# teach 버튼이 눌려있는가?
__($('Raccoon4*0:button.pressed.teach')).d
# teach 버튼을 클릭했는가?
__($('Raccoon4*0:button.click.teach')).e
# teach 버튼을 길게 클릭했는가?
__($('Raccoon4*0:button.long_click.teach')).e

# play 버튼이 눌려있는가?
__($('Raccoon4*0:button.pressed.play')).d

```

```

# play 버튼을 클릭했는가?
__('Raccoon4*0:button.click.play').e
# play 버튼을 길게 클릭했는가?
__('Raccoon4*0:button.long_click.play').e

# power 버튼이 눌려있는가?
__('Raccoon4*0:button.pressed.power').d
# power 버튼을 클릭했는가?
__('Raccoon4*0:button.click.power').e

# delete 버튼이 눌려있는가?
__('Raccoon4*0:button.pressed.delete').d
# delete 버튼을 클릭했는가?
__('Raccoon4*0:button.click.delete').e
# delete 버튼을 길게 클릭했는가?
__('Raccoon4*0:button.long_click.delete').e

```

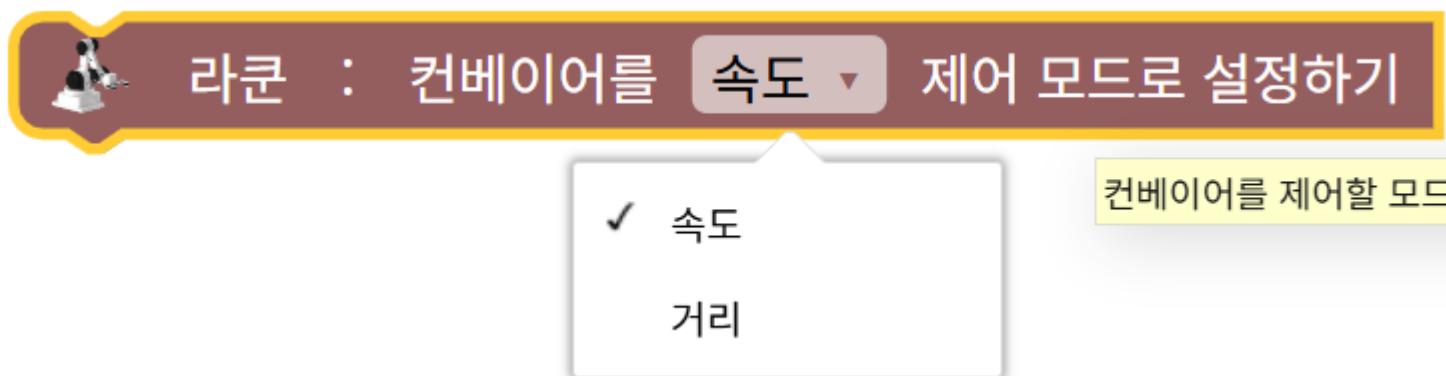
컨베이어 모드 설정하기

컨베이어를 제어할 모드를 설정합니다.

속도 제어 모드에서는 컨베이어의 속도만을 통해 컨베이어를 움직입니다.

거리 제어 모드에서는 컨베이어의 이동 거리만을 통해 컨베이어를 움직입니다.

두 모드는 동시에 사용할 수 없습니다.



드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
mode	드롭다운 옵션	컨베이어 모드	속도 (0), 거리 (1)

자바스크립트 코드

```
// 컨베이어를 속도 제어 모드로 설정하기
$('Raccoon4*0:peripheral>conveyor+mode').d = 0;

// 컨베이어를 거리 제어 모드로 설정하기
$('Raccoon4*0:peripheral>conveyor+mode').d = 1;
```

파이썬 코드

```
# 컨베이어를 속도 제어 모드로 설정하기
__($('Raccoon4*0:peripheral>conveyor+mode').d = 0

# 컨베이어를 거리 제어 모드로 설정하기
__($('Raccoon4*0:peripheral>conveyor+mode').d = 1
```

컨베이어 속도 설정하기

컨베이어 벨트의 속도를 설정합니다.

속도의 범위는 -100 ~ 100입니다.



드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
speed	입력값	컨베이어 속도	-100 ~ 100 사이 실수

자바스크립트 코드

```
// 컨베이어 속도를 100(으)로 정하기
$('Raccoon4*0:peripheral>conveyor+speed').d = 100;
```

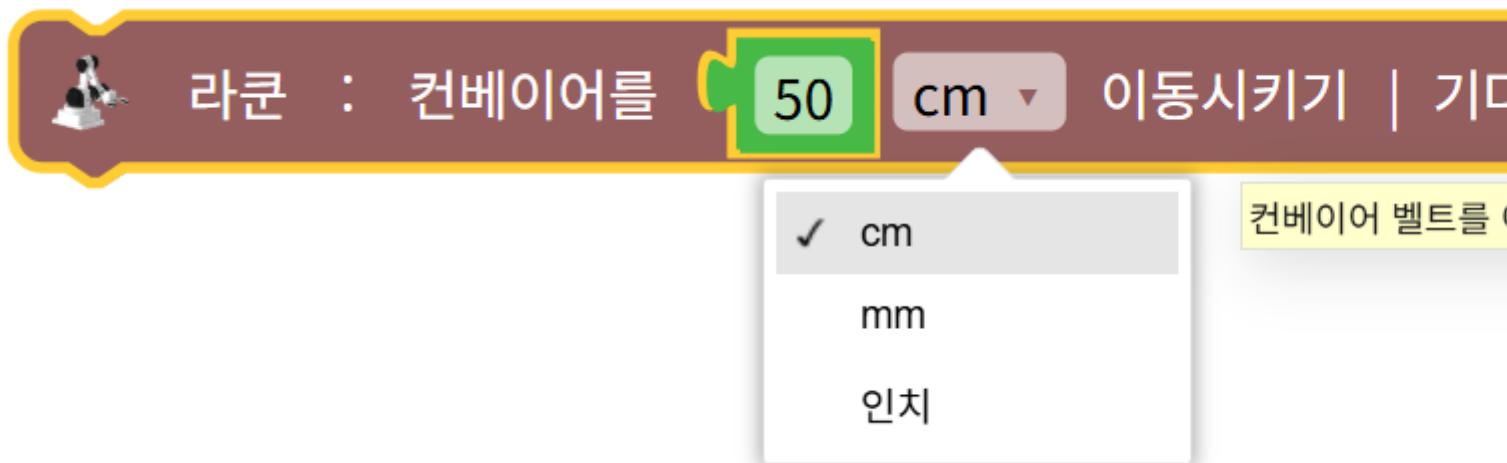
파이썬 코드

```
# 컨베이어 속도를 100(으)로 정하기
__('Raccoon4*0:peripheral>conveyor+speed').d = 100
```

컨베이어 이동 거리 설정하기

컨베이어 벨트를 이동시킬 거리를 설정합니다.

이 함수는 거리 제어 모드에서만 사용할 수 있습니다.



드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
distance	입력값	거리	0 이상 실수
unit	드롭다운 옵션	거리 단위	cm, mm, 인치 (inch)

자바스크립트 코드

```
// 컨베이어를 100cm 이동시키기 | 기다리기 0
$('Raccoon4*0:peripheral>conveyor+distance').d = __getDistance('Conveyor', 100, 'cm');
await $('Raccoon4*0:peripheral>conveyor+!distance').w();

// 컨베이어를 100mm 이동시키기 | 기다리기 0
$('Raccoon4*0:peripheral>conveyor+distance').d = __getDistance('Conveyor', 100, 'mm');
await $('Raccoon4*0:peripheral>conveyor+!distance').w();

// 컨베이어를 100 인치 이동시키기 | 기다리기 X
$('Raccoon4*0:peripheral>conveyor+distance').d = __getDistance('Conveyor', 100, 'inch');
```

파이썬 코드

```
# 컨베이어를 100cm 이동시키기 | 기다리기 0
__($('Raccoon4*0:peripheral>conveyor+distance').d = __getDistance('Conveyor', 100, 'cm'))
await __($('Raccoon4*0:peripheral>conveyor+!distance').w())

# 컨베이어를 100mm 이동시키기 | 기다리기 0
__($('Raccoon4*0:peripheral>conveyor+distance').d = __getDistance('Conveyor', 100, 'mm'))
await __($('Raccoon4*0:peripheral>conveyor+!distance').w())

# 컨베이어를 100 인치 이동시키기 | 기다리기 X
__($('Raccoon4*0:peripheral>conveyor+distance').d = __getDistance('Conveyor', 100, 'inch'))
```

컨베이어 속도 변경하기

컨베이어 벨트의 속도를 입력값만큼 변경합니다.



라쿤 : 컨베이어 속도를

50

만큼 바꾸기

컨베이어 벨트의 속도를 변경합니다.

드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
speed	입력 값	변경 속도	-200 ~ 200 사이 실수

자바스크립트 코드

```
// 컨베이어 속도를 -200 만큼 바꾸기
$('Raccoon4*0:peripheral>conveyor+speed').d = $('Raccoon4*0:peripheral>conveyor+speed').d + -200;

// 컨베이어 속도를 200 만큼 바꾸기
$('Raccoon4*0:peripheral>conveyor+speed').d = $('Raccoon4*0:peripheral>conveyor+speed').d + 200;
```

파이썬 코드

```
# 컨베이어 속도를 -200 만큼 바꾸기
__($('Raccoon4*0:peripheral>conveyor+speed').d = __($('Raccoon4*0:peripheral>conveyor+speed').d +
-200

# 컨베이어 속도를 200 만큼 바꾸기
__($('Raccoon4*0:peripheral>conveyor+speed').d = __($('Raccoon4*0:peripheral>conveyor+speed').d + 200
```

컨베이어가 동작 중인가?

컨베이어 벨트의 동작 여부를 참 (1) / 거짓 (0) (으)로 반환합니다.



라쿤 : 컨베이어가 동작 중인가?

컨베이어 벨트 동작 여부

자바스크립트 코드

```
// 컨베이어가 동작 중인가?  
$('Raccoon4*0:peripheral>conveyor+running').d;
```

파이썬 코드

```
# 컨베이어가 동작 중인가?  
_('Raccoon4*0:peripheral>conveyor+running').d
```

컨베이어 버튼이 눌려있는가?

컨베이어의 버튼이 눌려 있는지 또는 클릭 이벤트가 발생했는지 여부를 참 (1) / 거짓 (0) (으)로 반환합니다.



라쿤 : 컨베이어 버튼 이 눌려있는가 ?

✓ 이 눌려있는가

을 클릭했는가

을 길게 클릭했는가

컨베이어의 버튼

드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
state	드롭다운 옵션	클릭 이벤트	눌려있는가(pressed), 클릭했는가(click), 길게 클릭했는가(long_click)

자바스크립트 코드

```
// 컨베이어 버튼이 눌려있는가?
$( 'Raccoon4*0:peripheral>conveyor+button.pressed' ).d;

// 컨베이어 버튼을 클릭했는가?
$( 'Raccoon4*0:peripheral>conveyor+button.click' ).e;

// 컨베이어 버튼을 길게 클릭했는가?
$( 'Raccoon4*0:peripheral>conveyor+button.long_click' ).e;
```

파이썬 코드

```
# 컨베이어 버튼이 눌려있는가?
___( 'Raccoon4*0:peripheral>conveyor+button.pressed' ).d

# 컨베이어 버튼을 클릭했는가?
___( 'Raccoon4*0:peripheral>conveyor+button.click' ).e

# 컨베이어 버튼을 길게 클릭했는가?
___( 'Raccoon4*0:peripheral>conveyor+button.long_click' ).e
```