

## 블록

### 바퀴 속도 설정하기

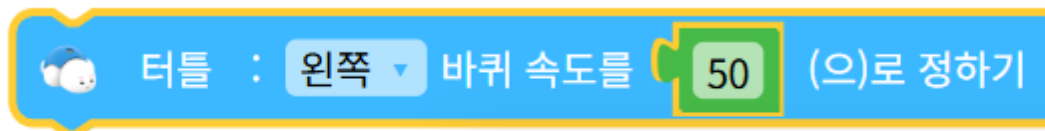
터틀의 바퀴 속도를 설정합니다.

바퀴 속도가 양수이면 앞쪽 방향으로 회전하고, 바퀴 속도가 음수이면 뒤쪽 방향으로 회전합니다.

예를 들어, 바퀴 속도가 100 이라면, 앞쪽 방향으로 100 의 속도로 회전하고,

바퀴 속도가 -100 이라면, 뒤쪽 방향으로 100 의 속도로 회전합니다.

한번 바퀴 속도를 설정하면, 다시 바퀴 속도를 설정하기 전까지 해당 속도로 터틀이 이동합니다.



바퀴 속도를 결정합니다. 속도의 범위는 -100 ~ 100 입니다.

### 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
wheel	드롭다운 옵션	바퀴 종류	왼쪽 (left), 오른쪽 (right), 양쪽 (left, right)
velocity	입력값	바퀴 속도	-100 ~ 100 정수, 0: 정지

### 자바스크립트 코드

```
// 왼쪽 바퀴 속도를 50(으) 로 정하기
if($('Turtle*0:wheel.move').d != 0) {
    $('Turtle*0:wheel.move').d = 0;
}
$('Turtle*0:wheel.speed.left').d = __getSpeed('Turtle*0', 50);

// 오른쪽 바퀴 속도를 40(으) 로 정하기
if($('Turtle*0:wheel.move').d != 0) {
    $('Turtle*0:wheel.move').d = 0;
}
$('Turtle*0:wheel.speed.right').d = __getSpeed('Turtle*0', 40);
```

```
// 양쪽 바퀴 속도를 30(으) 로 정하기
if($('Turtle*0:wheel.move').d != 0) {
    $('Turtle*0:wheel.move').d = 0;
}
$('Turtle*0:wheel.speed.left').d = __getSpeed('Turtle*0', 30);
$('Turtle*0:wheel.speed.right').d = __getSpeed('Turtle*0', 30);
```

## 파이썬 코드

```
# 왼쪽 바퀴 속도를 50(으) 로 정하기
if __('Turtle*0:wheel.move').d != 0:
    __('Turtle*0:wheel.move').d = 0
__('Turtle*0:wheel.speed.left').d = __getSpeed('Turtle*0', 50)

# 오른쪽 바퀴 속도를 40(으) 로 정하기
if __('Turtle*0:wheel.move').d != 0:
    __('Turtle*0:wheel.move').d = 0
__('Turtle*0:wheel.speed.right').d = __getSpeed('Turtle*0', 40)

# 양쪽 바퀴 속도를 30(으) 로 정하기
if __('Turtle*0:wheel.move').d != 0:
    __('Turtle*0:wheel.move').d = 0
__('Turtle*0:wheel.speed.left').d = __getSpeed('Turtle*0', 30)
__('Turtle*0:wheel.speed.right').d = __getSpeed('Turtle*0', 30)
```

## 거리 이동하기

터틀이 이동할 거리를 설정합니다.

바퀴 속도가 설정되어 있지 않은 경우, 이동하지 않습니다.

거리 값이 0 일 경우에는, 현재 설정되어 있는 바퀴 속도대로 멈추지 않고 이동합니다.

기다리기를 체크하면, 이동이 완료될 때까지 기다립니다.

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

터틀 :   이동하기 | 기다리기 ☒

- ✓ cm
- mm
- 인치

이동할 거리를 설정합니다.  
바퀴 속도를 설정하지 않은 경우, 이동하지 않습니다.  
거리 값이 0일 경우, 현재 바퀴 속도에 따라 계속 이동합니다.  
기다리기를 체크하면, 이동이 완료될 때까지 기다립니다.

## 드롭다운 옵션 및 입력값

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

## 자바스크립트 코드

```
// 50cm 이동하기 | 기다리기 0
$('Turtle*0:wheel.move').d = __getDistance('Turtle*0', 50, 'cm');
await $('Turtle*0:wheel.!move').w();

// 50mm 이동하기 | 기다리기 X
$('Turtle*0:wheel.move').d = __getDistance('Turtle*0', 50, 'mm');

// 50inch 이동하기 | 기다리기 X
$('Turtle*0:wheel.move').d = __getDistance('Turtle*0', 50, 'inch');
```

## 파이썬 코드

```
# 50cm 이동하기 | 기다리기 0
__('Turtle*0:wheel.move').d = __getDistance('Turtle*0', 50, 'cm')
await __('Turtle*0:wheel.!move').w()

# 50mm 이동하기 | 기다리기 X
__('Turtle*0:wheel.move').d = __getDistance('Turtle*0', 50, 'mm')
```

# 50inch 이동하기 | 기다리기 X

```
__('Turtle*0:wheel.move').d = __getDistance('Turtle*0', 50, 'inch')
```

## 시간 이동하기

터틀이 이동할 시간을 설정합니다.

바퀴 속도가 설정되어 있지 않은 경우, 이동하지 않습니다.

기다리기를 체크하면, 이동이 완료될 때까지 기다립니다.

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



✓ 초  
밀리초

이동할 시간을 설정합니다.  
바퀴 속도를 설정하지 않은 경우, 이동하지 않습니다.  
기다리기를 체크하면, 이동이 완료될 때까지 기다립니다.

## 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
time	입력값	시간 값	0 이상 실수
unit	드롭다운 옵션	시간 단위	초 (seconds), 밀리초 (milliseconds)

옵션을 밀리초 (milliseconds) 로 설정한 경우에는, time 값을 1000 으로 나눈 값이 입력됩니다.

## 자바스크립트 코드

```
// 5 초 이동하기 | 기다리기 0
```

```
await __stopAfterDelay('Turtle*0', 5, true);
```

```
// 5 밀리초 이동하기 | 기다리기 X
```

```
__stopAfterDelay('Turtle*0', 0.005, false);
```

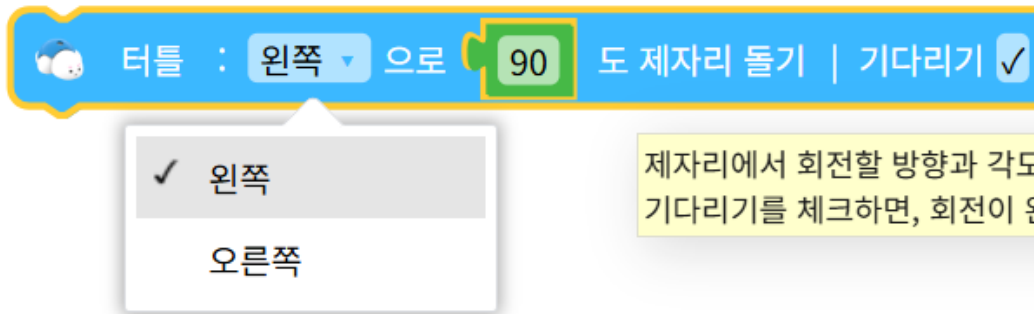
## 파이썬 코드

```
# 5 초 이동하기 | 기다리기 0
await __stopAfterDelay('Turtle*0', 5, True)

# 5 밀리초 이동하기 | 기다리기 X
__stopAfterDelay('Turtle*0', 0.005, False)
```

## 제자리 돌기

터들이 제자리에서 회전할 방향과 각도를 설정합니다.  
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
direction	드롭다운 옵션	회전 방향	왼쪽 (left), 오른쪽 (right)
degree	입력값	회전 각도	0 이상 정수

## 자바스크립트 코드

```
// 왼쪽으로 90 도 제자리 돌기 | 기다리기 0
await __turn_degree_left('Turtle*0', 90, true);

// 오른쪽으로 270 도 제자리 돌기 | 기다리기 X
__turn_degree_right('Turtle*0', 270, false);
```

## 파이썬 코드

```
# 왼쪽으로 90 도 제자리 돌기 | 기다리기 0
await __turn_degree_left('Turtle*0', 90, True)

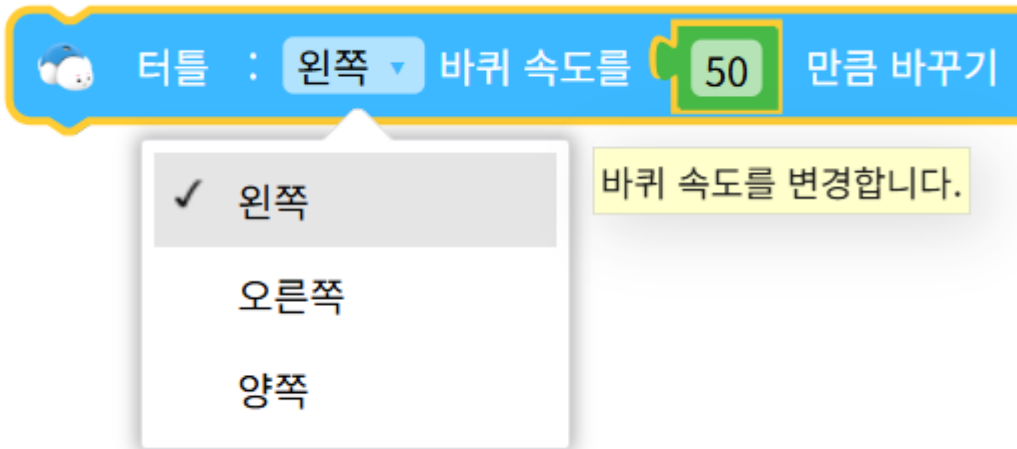
# 오른쪽으로 270 도 제자리 돌기 | 기다리기 X
__turn_degree_right('Turtle*0', 270, False)
```

## 바퀴 속도 변경하기

터틀의 바퀴 속도를 변경합니다.

현재의 바퀴 속도에 입력한 속도를 더한 값이 새로운 바퀴 속도가 됩니다.

새롭게 설정된 바퀴 속도의 범위는 -100 ~ 100 으로 설정됩니다.



## 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
wheel	드롭다운 옵션	바퀴 종류	왼쪽 (left), 오른쪽 (right), 양쪽 (left, right)
velocity	입력값	현재 바퀴 속도에 더할 속도 값	-200 ~ 200 정수

## 자바스크립트 코드

```

// 왼쪽 바퀴 속도를 50 만큼 바꾸기
if($('Turtle*0:wheel.move').d != 0) {
    $('Turtle*0:wheel.move').d = 0;
}
$('Turtle*0:wheel.speed.left').d = $('Turtle*0:wheel.speed.left').d + __getSpeed('Turtle*0', 50);

// 오른쪽 바퀴 속도를 -40 만큼 바꾸기
if($('Turtle*0:wheel.move').d != 0) {
    $('Turtle*0:wheel.move').d = 0;
}
$('Turtle*0:wheel.speed.right').d = $('Turtle*0:wheel.speed.right').d + __getSpeed('Turtle*0',
(-40));

// 양쪽 바퀴 속도를 200 만큼 바꾸기
if($('Turtle*0:wheel.move').d != 0) {
    $('Turtle*0:wheel.move').d = 0;
}
$('Turtle*0:wheel.speed.left').d = $('Turtle*0:wheel.speed.left').d + __getSpeed('Turtle*0',
200);
$('Turtle*0:wheel.speed.right').d = $('Turtle*0:wheel.speed.right').d + __getSpeed('Turtle*0',
200);

```

## 파이썬 코드

```

# 왼쪽 바퀴 속도를 50 만큼 바꾸기
if __('Turtle*0:wheel.move').d != 0:
    __('Turtle*0:wheel.move').d = 0
__('Turtle*0:wheel.speed.left').d = __('Turtle*0:wheel.speed.left').d + __getSpeed('Turtle*0',
50)

# 오른쪽 바퀴 속도를 -40 만큼 바꾸기
if __('Turtle*0:wheel.move').d != 0:
    __('Turtle*0:wheel.move').d = 0

```

```

__('Turtle*0:wheel.speed.right').d = __('Turtle*0:wheel.speed.right').d + __getSpeed('Turtle*0',
(-40))

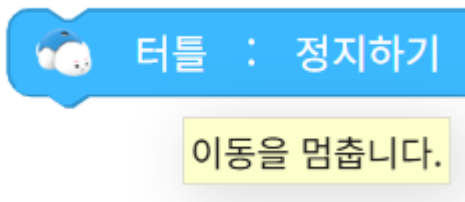
# 양쪽 바퀴 속도를 200 만큼 바꾸기
if __('Turtle*0:wheel.move').d != 0:
    __('Turtle*0:wheel.move').d = 0
__('Turtle*0:wheel.speed.left').d = __('Turtle*0:wheel.speed.left').d + __getSpeed('Turtle*0',
200)
__('Turtle*0:wheel.speed.right').d = __('Turtle*0:wheel.speed.right').d + __getSpeed('Turtle*0',
200)

```

## 정지하기

터틀의 이동을 멈춥니다.

터틀의 양쪽 바퀴 속도가 모두 0 으로 초기화됩니다.



## 자바스크립트 코드

```
__stopMove('Turtle*0');
```

## 파이썬 코드

```
__stopMove('Turtle*0')
```

## 바퀴가 움직이는 중인가?

터틀의 바퀴가 움직이고 있는지 아닌지 여부를 참 (1) / 거짓 (0) (으) 로 반환합니다.





터틀 : 바퀴가 움직이는 중인가?

바퀴가 움직이는 중이면 true, 멈춰있으면 false를 반환합니다.

## 자바스크립트 코드

```
$('Turtle*0:wheel.speed.left').d !== 0 || $('Turtle*0:wheel.speed.right').d !== 0
```

## 파이썬 코드

```
__('Turtle*0:wheel.speed.left').d != 0 or __('Turtle*0:wheel.speed.right').d != 0
```

## 바퀴 기준 회전하기

터틀이 제자리에서 회전할 기준과 방향, 각도를 설정합니다.

기다리기를 체크하면, 이동이 완료될 때까지 기다립니다.

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



터틀 : 왼쪽 ▾ 바퀴 기준 앞쪽 ▾ 방향으로 90 도 돌기 | 기다리기 ✓



터틀 : 왼쪽 ▾ 바퀴 기준 뒤쪽 ▾ 방향으로 90 도 돌기 | 기다리기

✓ 왼쪽  
오른쪽

회전할 기준과 방향, 각도를 설정합니다.  
기다리기를 체크하면, 회전이 완료될 때까지 기다립니다.

## 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
pivot	드롭다운 옵션	회전 기준	왼쪽 바퀴 (left wheel), 오른쪽 바퀴 (right wheel)
direction	드롭다운 옵션	방향	앞쪽 (forward), 뒤쪽 (backward)
degree	입력값	회전 각도	0 이상 정수

## 자바스크립트 코드

```
// 왼쪽 바퀴 기준 앞쪽 방향으로 90 도 돌기 | 기다리기 0
await __pivot('Turtle*0', 'left wheel', 'forward', 90, true);

// 왼쪽 바퀴 기준 뒤쪽 방향으로 90 도 돌기 | 기다리기 X
__pivot('Turtle*0', 'left wheel', 'backward', 90, false);

// 오른쪽 바퀴 기준 앞쪽 방향으로 90 도 돌기 | 기다리기 0
await __pivot('Turtle*0', 'right wheel', 'forward', 90, true);

// 오른쪽 바퀴 기준 뒤쪽 방향으로 90 도 돌기 | 기다리기 X
__pivot('Turtle*0', 'right wheel', 'backward', 90, false);
```

## 파이썬 코드

```
# 왼쪽 바퀴 기준 앞쪽 방향으로 90 도 돌기 | 기다리기 0
await __pivot('Turtle*0', 'left wheel', 'forward', 90, True)

# 왼쪽 바퀴 기준 뒤쪽 방향으로 90 도 돌기 | 기다리기 X
__pivot('Turtle*0', 'left wheel', 'backward', 90, False)

# 오른쪽 바퀴 기준 앞쪽 방향으로 90 도 돌기 | 기다리기 0
await __pivot('Turtle*0', 'right wheel', 'forward', 90, True)

# 오른쪽 바퀴 기준 뒤쪽 방향으로 90 도 돌기 | 기다리기 X
```

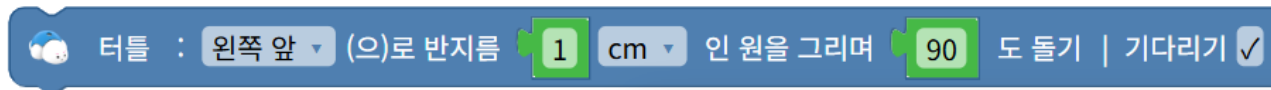
```
__pivot('Turtle*0', 'right wheel', 'backward', 90, False)
```

## 원 그리며 돌기

펜을 이용해 원을 그릴 때 회전할 방향, 반지름, 각도를 설정합니다.

기다리기를 체크하면, 이동이 완료될 때까지 기다립니다.

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



펜을 이용해 원을 그릴 때, 회전할 방향, 반지름, 각도를 설정합니다.  
기다리기를 체크하면, 회전이 완료될 때까지 기다립니다.

## 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
direction	드롭다운 옵션	방향	왼쪽 앞 (left forward), 왼쪽 뒤 (left backward), 오른쪽 앞 (right forward), 오른쪽 뒤 (right backward)
radius	입력값	원의 반지름	0 이상 실수
unit	드롭다운 옵션	거리 단위	cm, mm, 인치 (inch)
degree	입력값	회전 각도	0 이상 실수

## 자바스크립트 코드

```
// 왼쪽 앞 (으) 로 반지름 1cm 인 원을 그리며 90 도 돌기 | 기다리기 0
```

```
await __pivot_circle('Turtle*0', '', 'left forward', __getDistance('Turtle*0', 1, 'cm'), 90, true); //(robot, pivot, direction, circle_info, degree, wait_w)
```

```
// 왼쪽 뒤 (으) 로 반지름 1mm 인 원을 그리며 90 도 돌기 | 기다리기 0
```

```
await __pivot_circle('Turtle*0', '', 'left backward', __getDistance('Turtle*0', 1, 'mm'), 90, true); //(robot, pivot, direction, circle_info, degree, wait_w)
```

```
// 오른쪽 앞 (으) 로 반지름 1inch 인 원을 그리며 90 도 돌기 | 기다리기 X
__pivot_circle('Turtle*0', '', 'right forward', __getDistance('Turtle*0', 1, 'inch'), 90, false);
//(robot, pivot, direction, circle_info, degree, wait_w)

// 오른쪽 뒤 (으) 로 반지름 1cm 인 원을 그리며 90 도 돌기 | 기다리기 X
__pivot_circle('Turtle*0', '', 'right backward', __getDistance('Turtle*0', 1, 'cm'), 90, false);
//(robot, pivot, direction, circle_info, degree, wait_w)
```

## 파이썬 코드

```
# 왼쪽 앞 (으) 로 반지름 1cm 인 원을 그리며 90 도 돌기 | 기다리기 0
await __pivot_circle('Turtle*0', '', 'left forward', __getDistance('Turtle*0', 1, 'cm'), 90,
True) #(robot, pivot, direction, circle_info, degree, wait_w)

# 왼쪽 뒤 (으) 로 반지름 1mm 인 원을 그리며 90 도 돌기 | 기다리기 0
await __pivot_circle('Turtle*0', '', 'left backward', __getDistance('Turtle*0', 1, 'mm'), 90,
True) #(robot, pivot, direction, circle_info, degree, wait_w)

# 오른쪽 앞 (으) 로 반지름 1inch 인 원을 그리며 90 도 돌기 | 기다리기 X
__pivot_circle('Turtle*0', '', 'right forward', __getDistance('Turtle*0', 1, 'inch'), 90, False)
#(robot, pivot, direction, circle_info, degree, wait_w)

# 오른쪽 뒤 (으) 로 반지름 1cm 인 원을 그리며 90 도 돌기 | 기다리기 X
__pivot_circle('Turtle*0', '', 'right backward', __getDistance('Turtle*0', 1, 'cm'), 90, False)
#(robot, pivot, direction, circle_info, degree, wait_w)
```

## 센서로 선 따라가기

터틀이 바닥의 컬러 센서를 이용하여 특정한 선을 따라갑니다.

기다리기를 체크하면, 이동이 완료될 때까지 기다립니다.

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

터틀 : 검정색 ▾ 선을 따라가기 | 기다리기 ✓

- ✓ 검정색
- 빨간색
- 초록색
- 파란색
- 아무 색

바닥의 컬러 센서를 이용해, 특정 색의 선을 따라 이동합니다.

### 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
color	드롭다운 옵션	선의 색	검정색 (8), 빨간색 (9), 초록색 (11), 파란색 (13), 아무 색 (15)

### 자바스크립트 코드

```
// 검정색 선을 따라가기 (8) | 기다리기 0
$('Turtle*0:trace.mode').d = 8;
await $('Turtle*0:trace.!mode').w();

// 빨간색 선을 따라가기 (9) | 기다리기 0
$('Turtle*0:trace.mode').d = 9;
await $('Turtle*0:trace.!mode').w();

// 초록색 선을 따라가기 (11) | 기다리기 0
$('Turtle*0:trace.mode').d = 11;
await $('Turtle*0:trace.!mode').w();

// 파란색 선을 따라가기 (13) | 기다리기 X
```

```
$('#Turtle*0:trace.mode').d = 13;
```

```
// 아무색 선을 따라가기 (15) | 기다리기 X
```

```
$('#Turtle*0:trace.mode').d = 15;
```

## 파이썬 코드

```
# 검정색 선을 따라가기 (8) | 기다리기 0
```

```
__('Turtle*0:trace.mode').d = 8
```

```
await __('Turtle*0:trace.!mode').w()
```

```
# 빨간색 선을 따라가기 (9) | 기다리기 0
```

```
__('Turtle*0:trace.mode').d = 9
```

```
await __('Turtle*0:trace.!mode').w()
```

```
# 초록색 선을 따라가기 (11) | 기다리기 0
```

```
__('Turtle*0:trace.mode').d = 11
```

```
await __('Turtle*0:trace.!mode').w()
```

```
# 파란색 선을 따라가기 (13) | 기다리기 X
```

```
__('Turtle*0:trace.mode').d = 13
```

```
# 아무색 선을 따라가기 (15) | 기다리기 X
```

```
__('Turtle*0:trace.mode').d = 15
```

## 특정 색 까지 검정색 선을 따라 이동하기

터틀 바닥의 컬러 센서를 이용해, 특정 색을 만날 때 까지 검정색 선을 따라 이동합니다.

기다리기를 체크하면, 이동이 완료될 때까지 기다립니다.

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



터틀 : 검정색 선을 따라 빨간색 까지 이동하기 | 기다리기 ✓

- ✓ 빨간색
- 초록색
- 하늘색
- 파란색
- 자주색
- 아무 색

바닥의 컬러 센서를 이용해, 특정 색을 만날 때까지 검정색 선을 따라 이동합니다.

## 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
color	드롭다운 옵션	선의 색	빨간색 (49), 초록색 (51), 하늘색 (52), 파란색 (53), 자주색 (54), 아무색 (55)

## 자바스크립트 코드

```
// 검정색 선을 따라 빨간색까지 이동하기 (49) | 기다리기 0
$('Turtle*0:trace.mode').d = 49;
await $('Turtle*0:trace.!mode').w();

// 검정색 선을 따라 초록색까지 이동하기 (51) | 기다리기 0
$('Turtle*0:trace.mode').d = 51;
await $('Turtle*0:trace.!mode').w();

// 검정색 선을 따라 하늘색까지 이동하기 (52) | 기다리기 0
$('Turtle*0:trace.mode').d = 52;
await $('Turtle*0:trace.!mode').w();

// 검정색 선을 따라 파란색까지 이동하기 (53) | 기다리기 X
$('Turtle*0:trace.mode').d = 53;
```

```
// 검정색 선을 따라 자주색까지 이동하기 (54) | 기다리기 X
$('Turtle*0:trace.mode').d = 54;

// 검정색 선을 따라 아무색까지 이동하기 (55) | 기다리기 X
$('Turtle*0:trace.mode').d = 55;
```

## 파이썬 코드

```
# 검정색 선을 따라 빨간색까지 이동하기 (49) | 기다리기 0
__('Turtle*0:trace.mode').d = 49
await __('Turtle*0:trace.!mode').w()

# 검정색 선을 따라 초록색까지 이동하기 (51) | 기다리기 0
__('Turtle*0:trace.mode').d = 51
await __('Turtle*0:trace.!mode').w()

# 검정색 선을 따라 하늘색까지 이동하기 (52) | 기다리기 0
__('Turtle*0:trace.mode').d = 52
await __('Turtle*0:trace.!mode').w()

# 검정색 선을 따라 파란색까지 이동하기 (53) | 기다리기 X
__('Turtle*0:trace.mode').d = 53

# 검정색 선을 따라 자주색까지 이동하기 (54) | 기다리기 X
__('Turtle*0:trace.mode').d = 54

# 검정색 선을 따라 아무색까지 이동하기 (55) | 기다리기 X
__('Turtle*0:trace.mode').d = 55
```

## 검정색 까지 특정 색 선을 따라 이동하기

터틀 바닥의 컬러 센서를 이용해, 검정 색을 만날 때 까지 특정 색 선을 따라 이동합니다.  
 기다리기를 체크하면, 이동이 완료될 때까지 기다립니다.  
 단, 기다리기를 체크한 경우에는 async 함수 내에서만 사용할 수 있습니다.



터틀 : 빨간색 ▾ 선을 따라 검정색까지 이동하기 | 기다리기 ✓

- ✓ 빨간색
- 초록색
- 파란색
- 아무 색

바닥의 컬러 센서를 이용해, 검정색을 만날 때까지 특정 색 선을 따라 이동합니다.

## 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
color	드롭다운 옵션	선의 색	빨간색 (57), 초록색 (59), 파란색 (61), 아무색 (63)

## 자바스크립트 코드

```
// 빨간색 선을 따라 검정색까지 이동하기 (57) | 기다리기 0
$('Turtle*0:trace.mode').d = 57;
await $('Turtle*0:trace.!mode').w();

// 초록색 선을 따라 검정색까지 이동하기 (59) | 기다리기 0
$('Turtle*0:trace.mode').d = 59;
await $('Turtle*0:trace.!mode').w();

// 파란색 선을 따라 검정색까지 이동하기 (61) | 기다리기 X
$('Turtle*0:trace.mode').d = 61;

// 아무 색 선을 따라 검정색까지 이동하기 (63) | 기다리기 X
$('Turtle*0:trace.mode').d = 63;
```

## 파이썬 코드

# 빨간색 선을 따라 검정색까지 이동하기 (57) | 기다리기 0

```
__('Turtle*0:trace.mode').d = 57
```

```
await __('Turtle*0:trace.!mode').w()
```

# 초록색 선을 따라 검정색까지 이동하기 (59) | 기다리기 0

```
__('Turtle*0:trace.mode').d = 59
```

```
await __('Turtle*0:trace.!mode').w()
```

# 파란색 선을 따라 검정색까지 이동하기 (61) | 기다리기 X

```
__('Turtle*0:trace.mode').d = 61
```

# 아무 색 선을 따라 검정색까지 이동하기 (63) | 기다리기 X

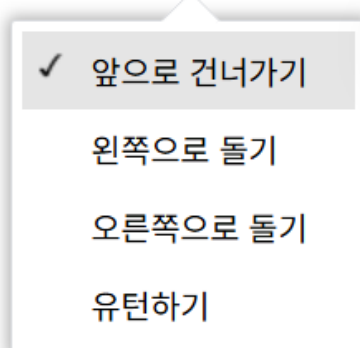
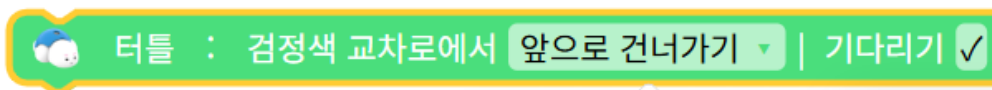
```
__('Turtle*0:trace.mode').d = 63
```

## 검정색 교차로에서 행동 설정하기

터틀 바닥의 컬러 센서를 이용해, 검정색 교차로를 인식하면 지정된 행동을 합니다.

기다리기를 체크하면, 이동이 완료될 때까지 기다립니다.

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



검정색 교차로에서 지정된 행동을 합니다.

## 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
behavior	드롭다운 옵션	검정색 교차로에서의 행동	앞으로 건너가기 (16), 왼쪽으로 돌기 (24), 오른쪽으로 돌기 (32), 유턴하기 (40)

### 자바스크립트 코드

```
// 검정색 교차로에서 앞으로 건너가기 (16) | 기다리기 0
$('Turtle*0:trace.mode').d = 16;
await $('Turtle*0:trace.!mode').w();

// 검정색 교차로에서 왼쪽으로 돌기 (24) | 기다리기 0
$('Turtle*0:trace.mode').d = 24;
await $('Turtle*0:trace.!mode').w();

// 검정색 교차로에서 오른쪽으로 돌기 (32) | 기다리기 X
$('Turtle*0:trace.mode').d = 32;

// 검정색 교차로에서 유턴하기 (40) | 기다리기 X
$('Turtle*0:trace.mode').d = 40;
```

### 파이썬 코드

```
# 검정색 교차로에서 앞으로 건너가기 (16) | 기다리기 0
__('Turtle*0:trace.mode').d = 16
await __('Turtle*0:trace.!mode').w()

# 검정색 교차로에서 왼쪽으로 돌기 (24) | 기다리기 0
__('Turtle*0:trace.mode').d = 24
await __('Turtle*0:trace.!mode').w()

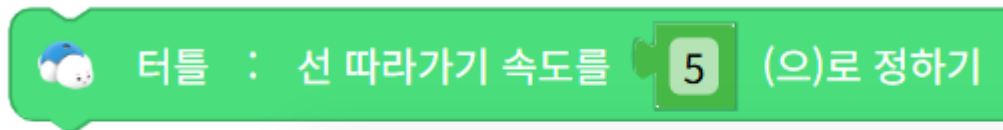
# 검정색 교차로에서 오른쪽으로 돌기 (32) | 기다리기 X
__('Turtle*0:trace.mode').d = 32
```

```
# 검정색 교차로에서 유턴하기 (40) | 기다리기 X
__('Turtle*0:trace.mode').d = 40
```

## 선 따라가기 속도 설정

터틀의 선 따라가기 속도를 설정합니다.

속도의 범위는 1 ~ 10 입니다.



선 따라가기 속도를 설정합니다. 속도의 범위는 1 ~ 10 입니다.

## 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
velocity	입력값	선 따라가기 속도 값	1 ~ 10 사이 정수

## 자바스크립트 코드

```
// 선 따라가기 속도 5 로 정하기
$('Turtle*0:trace.speed').d = 5;
```

## 파이썬 코드

```
# 선 따라가기 속도 5 로 정하기
__('Turtle*0:trace.speed').d = 5
```

## 선 따라가기 방향 변화량 설정

터틀의 선 따라가기 방향 변화량을 설정합니다.

변화량의 범위는 1 ~ 10 입니다.



터틀 : 선 따라가기 방향 변화량을 5 (으)로 정하기

선 따라가기 방향 변화량을 설정합니다. 변화량의 범위는 1 ~ 10 입니다.

#### 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
differential	입력값	선 따라가기 방향 변화량 값	1 ~ 10 사이 정수

#### 자바스크립트 코드

```
$('Turtle*0:trace.gain').d = 5; // 선 따라가기 방향 변화량 5 로 설정하기
```

#### 파이썬 코드

```
__('Turtle*0:trace.gain').d = 5 # 선 따라가기 방향 변화량 5 로 설정하기
```

#### 선 따라가기 멈추기

터틀의 선 따라가기 기능을 종료합니다.



터틀 : 선 따라가기 멈추기

선 따라가기 기능을 종료합니다.

#### 자바스크립트 코드

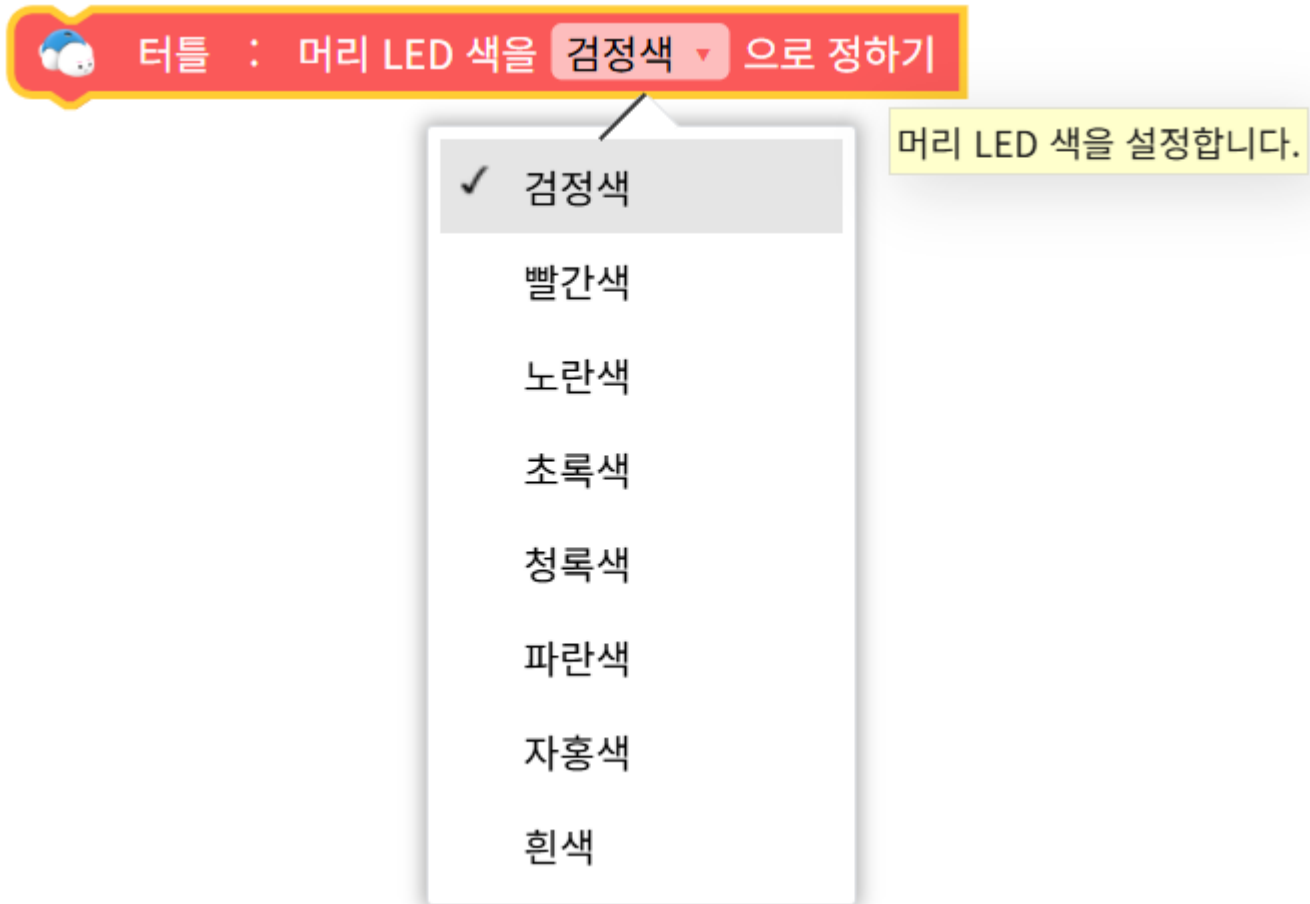
```
$('Turtle*0:trace.mode').d = 0; // 선 따라가기 멈추기
```

## 파이썬 코드

```
__('Turtle*0:trace.mode').d = 0 # 선 따라가기 멈추기
```

## LED 색 설정하기

터틀의 머리 LED 색을 설정합니다.



드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
color	드롭다운 옵션	색상	검정색 ([0, 0, 0]), 빨간색 ([255, 0, 0]), 노란색 ([255, 255, 0]), 초록색 ([0, 255, 0]), 청록색 ([0, 255, 255]), 파란색 ([0, 0, 255]), 자홍색 ([255, 0, 255]), 흰색 ([255, 255, 255])

## 자바스크립트 코드

```
// 머리 LED 색을 검정색으로 정하기
$('Turtle*0:head_led').d = [0, 0, 0];

// 머리 LED 색을 빨간색으로 정하기
$('Turtle*0:head_led').d = [255, 0, 0];

// 머리 LED 색을 노란색으로 정하기
$('Turtle*0:head_led').d = [255, 255, 0];

// 머리 LED 색을 초록색으로 정하기
$('Turtle*0:head_led').d = [0, 255, 0];

// 머리 LED 색을 청록색으로 정하기
$('Turtle*0:head_led').d = [0, 255, 255];

// 머리 LED 색을 파란색으로 정하기
$('Turtle*0:head_led').d = [0, 0, 255];

// 머리 LED 색을 자홍색으로 정하기
$('Turtle*0:head_led').d = [255, 0, 255];

// 머리 LED 색을 흰색으로 정하기
$('Turtle*0:head_led').d = [255, 255, 255];
```

## 파이썬 코드

```
# 머리 LED 색을 검정색으로 정하기
__('Turtle*0:head_led').d = [0, 0, 0]

# 머리 LED 색을 빨간색으로 정하기
__('Turtle*0:head_led').d = [255, 0, 0]

# 머리 LED 색을 노란색으로 정하기
__('Turtle*0:head_led').d = [255, 255, 0]

# 머리 LED 색을 초록색으로 정하기
__('Turtle*0:head_led').d = [0, 255, 0]

# 머리 LED 색을 청록색으로 정하기
__('Turtle*0:head_led').d = [0, 255, 255]

# 머리 LED 색을 파란색으로 정하기
__('Turtle*0:head_led').d = [0, 0, 255]

# 머리 LED 색을 자홍색으로 정하기
__('Turtle*0:head_led').d = [255, 0, 255]

# 머리 LED 색을 흰색으로 정하기
__('Turtle*0:head_led').d = [255, 255, 255]
```

## LED 색 색상 카테고리 블록으로 정하기

색상 카테고리에 있는 블록들로 터틀의 머리 LED 색을 설정합니다.



터틀 : 머리 LED 색을 기본 색상 으로 정하기



색상 카테고리에 있는 블록들로 머리 LED 색을 설정합니다.

## 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
color	입력값	LED 색상	RGB 배열 ([255,255,255])

## 자바스크립트 코드

```
// 머리 LED 색상을 R:255, G:0, B:0 색상으로 정하기
$('Turtle*0:head_led').d = [255, 0, 0];

// 머리 LED 색상을 R:255, G:255, B:0 색상으로 정하기
$('Turtle*0:head_led').d = [255, 255, 0];

// 머리 LED 색상을 무작위 색상으로 정하기
$('Turtle*0:head_led').d = __randomColor();
```

## 파이썬 코드

```
# 머리 LED 색상을 R:255, G:0, B:0 색상으로 정하기
__('Turtle*0:head_led').d = [255, 0, 0]

# 머리 LED 색상을 R:255, G:255, B:0 색상으로 정하기
__('Turtle*0:head_led').d = [255, 255, 0]

# 머리 LED 색상을 무작위 색상으로 정하기
__('Turtle*0:head_led').d = __randomColor()
```

## LED 색 색상 지정 RGB 만큼 변경하기

지정한 R,G,B 값만큼 터틀의 머리 LED 색을 변경합니다.



터틀 : 머리 LED 색을 R: 0 G: 0 B: 0 만큼 바꾸기

지정한 R, G, B 값만큼 머리 LED 색을 변경합니다.

### 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
color	입력값	변경 R,G,B 값	R,G,B 각각 -255~255 사이 정수

### 자바스크립트 코드

```
// 머리 LED 색을 R : 1, G : 2, B : 3 만큼 바꾸기
$('Turtle*0:head_led').d = [$('Turtle*0:head_led').d[0] + 1, $('Turtle*0:head_led').d[1] + 2,
$('Turtle*0:head_led').d[2] + 3];
```

### 파이썬 코드

```
# 머리 LED 색을 R : 1, G : 2, B : 3 만큼 바꾸기
__('Turtle*0:head_led').d = [__('Turtle*0:head_led').d[0] + 1, __('Turtle*0:head_led').d[1] + 2,
__('Turtle*0:head_led').d[2] + 3]
```

## LED 끄기

터틀의 머리 LED 색을 없앱니다.



터틀 : 머리 LED 끄기

머리 LED 색을 없앱니다.

## 자바스크립트 코드

```
// 머리 LED 끄기  
$('Turtle*0:head_led').d = [0, 0, 0];
```

## 파이썬 코드

```
# 머리 LED 끄기  
__('Turtle*0:head_led').d = [0, 0, 0]
```

## 버저음 설정하기

지정된 주파수로 터틀의 버저음을 설정합니다.

주파수의 범위는 10hz ~ 4200hz 입니다.



지정된 주파수로 버저음을 설정합니다. 주파수의 범위는 10hz ~ 4200hz 입니다.

## 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
sound	입력값	버저음 주파수	10 ~ 4200(hz)

## 자바스크립트 코드

```
// 버저음 주파수 4200hz 로 설정하기  
$('Turtle*0:sound.buzz').d = 4200;
```

## 파이썬 코드

```
# 버저음 주파수 4200hz 로 설정하기  
__('Turtle*0:sound.buzz').d = 4200
```

## 음계 연주하기

터틀이 지정된 음계를 재생합니다.



특정 음계를 재생합니다.

## 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
$('Turtle*0:sound.note').d = 4;

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

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

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

## 파이썬 코드

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

# 1 옥타브 레 (Re) 음을 연주하기
__('Turtle*0:sound.note').d = 6

# 2 옥타브 도 (Do) 음을 연주하기
__('Turtle*0:sound.note').d = 16

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

## 소리 재생하기

터틀이 특정 사운드 클립을 재생합니다.

기다리기를 체크하면, 재생이 완료될 때까지 기다립니다.

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

터틀 : beep 소리 재생하기 | 기다리기 ✓

- ✓ beep
- beep2
- beep3
- beep\_repeat
- beep\_random
- beep2\_random
- beep3\_random
- beep\_random\_repeat
- siren
- siren\_repeat
- engine

특정 사운드 클립을 재생합니다.  
기다리기를 체크하면, 재생이 완료될 때까지 기다립니다.

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), beep2_random(6), beep3_random(7), beep_random_repeat(8), siren(16), siren_repeat(17), engine(32), engine_repeat(33), noise_random(34), r2d2(48), toy_march(64), birthday(65), dibidibidip(66), mission_complete(67)

### 자바스크립트 코드

```
// dibidibidip(66) 소리 재생하기 | 기다리기 0
$('Turtle*0:sound.clip').d = 66;
await $('Turtle*0:sound.!clip').w();

// mission_complete(67) 소리 재생하기 | 기다리기 x
$('Turtle*0:sound.clip').d = 67;
```

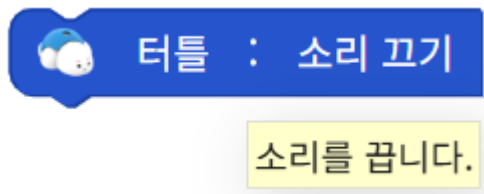
### 파이썬 코드

```
# dibidibidip(66) 소리 재생하기 | 기다리기 0
__('Turtle*0:sound.clip').d = 66
await __('Turtle*0:sound.!clip').w()
```

```
# mission_complete(67) 소리 재생하기 | 기다리기 X  
__('Turtle*0:sound.clip').d = 67
```

## 소리 끄기

터틀의 소리를 끕니다.



## 자바스크립트 코드

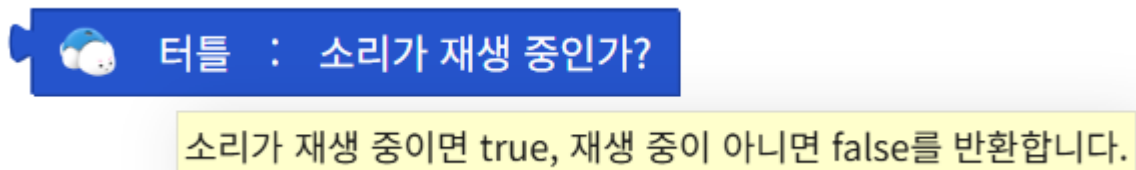
```
// 터틀 소리 끄기  
__stopSound('Turtle*0');
```

## 파이썬 코드

```
# 터틀 소리 끄기  
__stopSound('Turtle*0')
```

## 소리가 재생 중인가?

터틀의 소리가 재생중인지 아닌지 여부를 참 (1) / 거짓 (0) (으) 로 반환합니다.



## 자바스크립트 코드

```
// 터틀의 소리가 재생 중인가? - 재생 시 true, 아닐시 false  
$('Turtle*0:sound.playing').d;
```

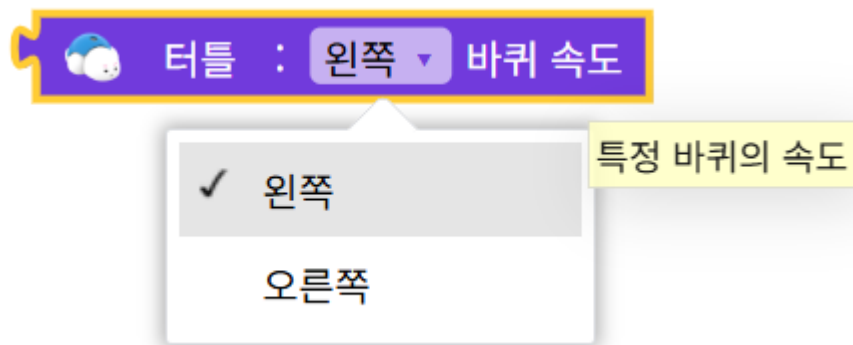


## 파이썬 코드

```
# 터틀의 소리가 재생 중인가? - 재생 시 True, 아닐시 False
__('Turtle*0:sound.playing').d
```

## 바퀴 속도 값

터틀의 지정한 바퀴 속도 값을 가져옵니다.



## 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
direction	드롭다운 옵션	방향	왼쪽 (left), 오른쪽 (right)

## 자바스크립트 코드

```
// 왼쪽 바퀴 속도
__getSpeedInput('Turtle*0', $('Turtle*0:wheel.speed.left').d);

// 오른쪽 바퀴 속도
__getSpeedInput('Turtle*0', $('Turtle*0:wheel.speed.right').d);
```

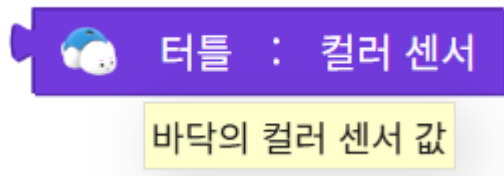
## 파이썬 코드

```
# 왼쪽 바퀴 속도
__getSpeedInput('Turtle*0', __('Turtle*0:wheel.speed.left').d)

# 오른쪽 바퀴 속도
__getSpeedInput('Turtle*0', __('Turtle*0:wheel.speed.right').d)
```

## 컬러 센서 값

터틀의 컬러 센서 값을 가져옵니다.



## 자바스크립트 코드

```
// 컬러 센서 값
$('Turtle*0:line').d;
```

## 파이썬 코드

```
# 컬러 센서 값
__('Turtle*0:line').d
```

## 카드 색깔 번호 값

터틀 바닥의 컬러 센서를 통해 읽은 카드의 색깔 번호를 반환합니다.



바닥의 컬러 센서를 통해 읽은 카드의 색깔의 번호

## 자바스크립트 코드

```
// 읽은 카드 색깔 번호 값  
$('Turtle*0:color.name').d;
```

## 파이썬 코드

```
# 읽은 카드 색깔 번호 값  
__('Turtle*0:color.name').d
```

## 카드 색깔 패턴 값

터틀 바닥의 컬러 센서를 통해 읽은 카드의 색깔의 패턴을 반환합니다.



바닥의 컬러 센서를 통해 읽은 카드 색깔의 패턴

## 자바스크립트 코드

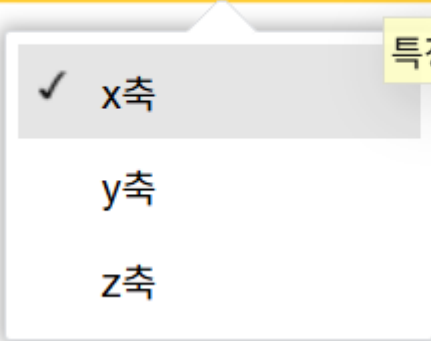
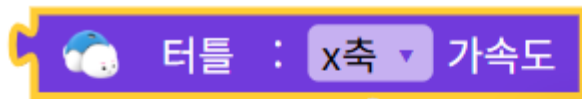
```
// 카드 색깔 패턴 값  
$('Turtle*0:card').d;
```

## 파이썬 코드

```
# 카드 색깔 패턴 값  
__('Turtle*0:card').d
```

## 중력 가속도 값

터틀의 특정 축의 중력 가속도 값을 가져옵니다.



특정 축의 중력 가속도 값

### 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
axis	드롭다운 옵션	축 기준	x 축, y 축, z 축

### 자바스크립트 코드

```
// x 축 기준 중력 가속도 값
$('Turtle*0:acceleration.x').d;

//y 축 기준 중력 가속도 값
$('Turtle*0:acceleration.y').d;

//z 축 기준 중력 가속도 값
$('Turtle*0:acceleration.z').d;
```

### 파이썬 코드

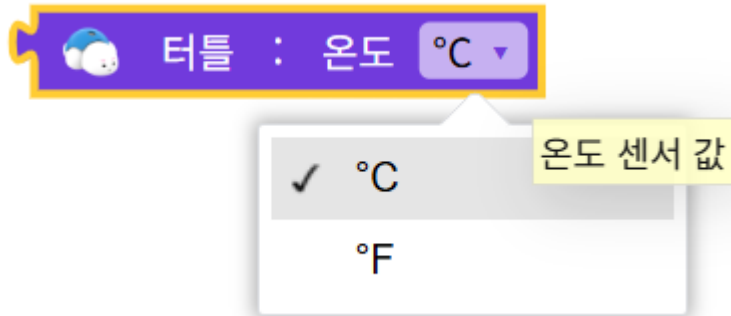
```
# x 축 기준 중력 가속도 값
__('Turtle*0:acceleration.x').d

# y 축 기준 중력 가속도 값
__('Turtle*0:acceleration.y').d
```

```
# z 축 기준 중력 가속도 값
__('Turtle*0:acceleration.z').d
```

## 온도 센서 값

터틀의 온도 센서 값을 가져옵니다.



## 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
unit	드롭다운 옵션	온도 단위	섭씨 (°C), 화씨 (°F)

## 자바스크립트 코드

```
// 섭씨 기준 온도센서 값
__getTemperature($('Turtle*0:temperature').d, '°C');

// 화씨 기준 온도센서 값
__getTemperature($('Turtle*0:temperature').d, '°F');
```

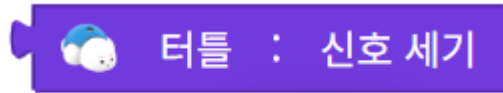
## 파이썬 코드

```
# 섭씨 기준 온도센서 값
__getTemperature(__('Turtle*0:temperature').d, '°C')

# 화씨 기준 온도센서 값
__getTemperature(__('Turtle*0:temperature').d, '°F')
```

## 신호 세기 값

터틀의 신호 세기 값을 가져옵니다.



신호 세기

## 자바스크립트 코드

```
// 신호 세기 값  
$('Turtle*0:signal_strength').d;
```

## 파이썬 코드

```
# 신호 세기 값  
__('Turtle*0:signal_strength').d
```

## 배터리 충전 상태 값

터틀의 배터리 충전 상태 값을 가져옵니다.



배터리의 충전 상태 값

## 자바스크립트 코드

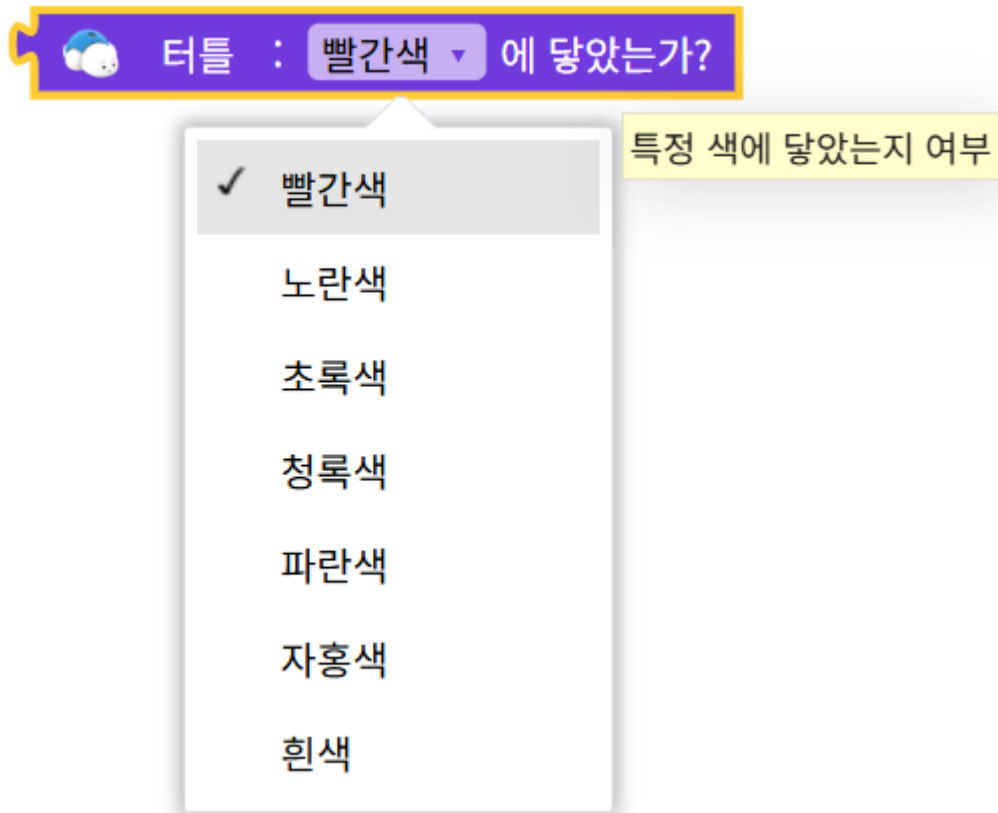
```
// 배터리 충전 상태 값  
$('Turtle*0:battery').d;
```

## 파이썬 코드

```
# 배터리 충전 상태 값
__('Turtle*0:battery').d
```

## 특정 색에 닿았는가?

지정한 특정 색에 닿았는지 터틀 컬러센서를 통해 측정하여 참 (1) / 거짓 (0) (으) 로 반환합니다.



## 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
color	드롭다운 옵션	컬러센서 인식 색	빨간색 (1), 노란색 (2), 초록색 (3), 청록색 (4), 파란색 (5), 자홍색 (6), 흰색 (7)

## 자바스크립트 코드

```
// 빨간색에 닿았는가?
$('Turtle*0:color.name').d === 1;
```

```

// 노란색에 닿았는가?
$('Turtle*0:color.name').d === 2;

// 초록색에 닿았는가?
$('Turtle*0:color.name').d === 3;

// 청록색에 닿았는가?
$('Turtle*0:color.name').d === 4;

// 파란색에 닿았는가?
$('Turtle*0:color.name').d === 5;

// 자홍색에 닿았는가?
$('Turtle*0:color.name').d === 6;

// 흰색에 닿았는가?
$('Turtle*0:color.name').d === 7;

```

## 파이썬 코드

```

# 빨간색에 닿았는가?
__('Turtle*0:color.name').d == 1

# 노란색에 닿았는가?
__('Turtle*0:color.name').d == 2

# 초록색에 닿았는가?
__('Turtle*0:color.name').d == 3

# 청록색에 닿았는가?
__('Turtle*0:color.name').d == 4

# 파란색에 닿았는가?

```



```
__('Turtle*0:color.name').d == 5
```

# 자홍색에 닿았는가?

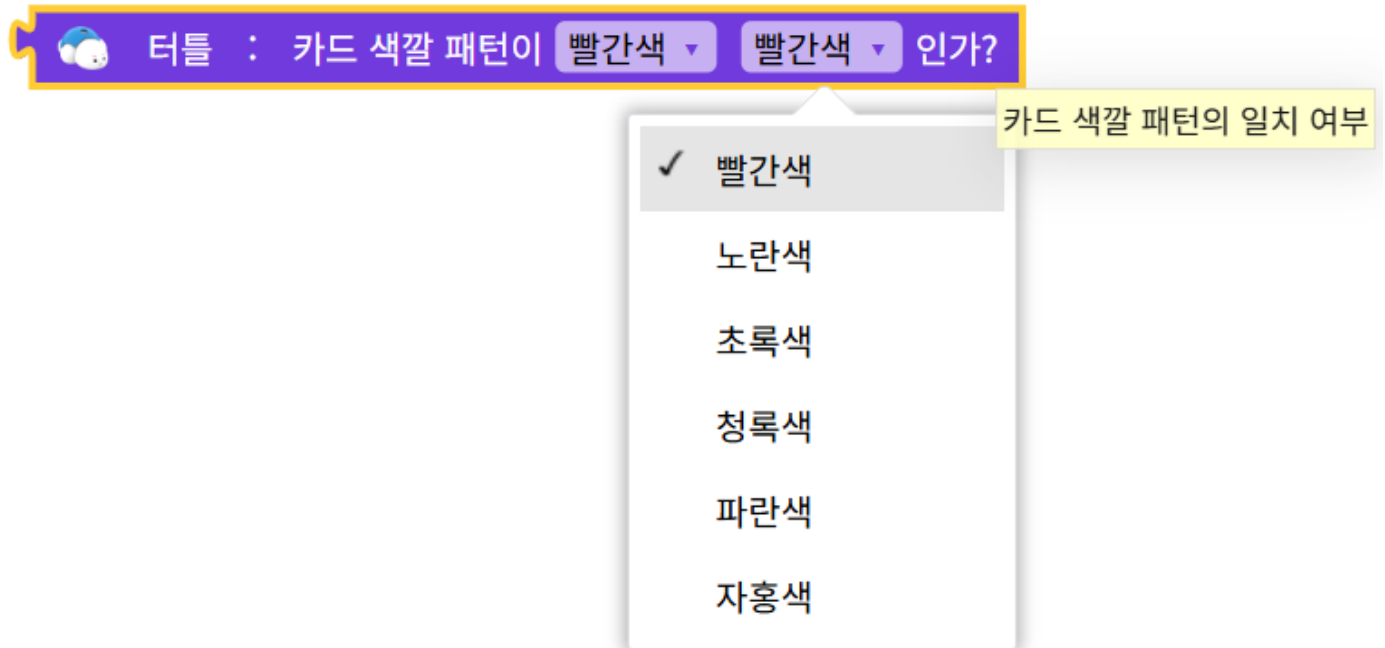
```
__('Turtle*0:color.name').d == 6
```

# 흰색에 닿았는가?

```
__('Turtle*0:color.name').d == 7
```

## 카드 색깔 패턴이 ~ 인가?

컬러센서에 인식 된 카드 색깔의 패턴 일치 여부에 따라 참 (1) / 거짓 (0) (으) 로 반환합니다.



## 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
pattern 1	드롭다운 옵션	카드 색깔 패턴	빨간색 (red), 노란색 (yellow), 초록색 (green), 청록색 (cyan), 파란색 (blue), 자홍색 (magenta)

이름	구분	설명	범위 / 종류
pattern 2	드롭다운 옵션	카드 색깔 패턴	빨간색 (red), 노란색 (yellow), 초록색 (green), 청록색 (cyan), 파란색 (blue), 자홍색 (magenta)

## 자바스크립트 코드

```
// 카드 색깔 패턴이 빨간색 빨간색 인가?
```

```
$('#Turtle*0:card').d === 9;
```

```
// 카드 색깔 패턴이 빨간색 노란색 인가?
```

```
$('#Turtle*0:card').d === 10;
```

```
// 카드 색깔 패턴이 빨간색 초록색 인가?
```

```
$('#Turtle*0:card').d === 11;
```

```
// 카드 색깔 패턴이 빨간색 청록색 인가?
```

```
$('#Turtle*0:card').d === 12;
```

```
// 카드 색깔 패턴이 빨간색 파란색 인가?
```

```
$('#Turtle*0:card').d === 13;
```

```
// 카드 색깔 패턴이 빨간색 자홍색 인가?
```

```
$('#Turtle*0:card').d === 14;
```

```
// 카드 색깔 패턴이 노란색 빨간색 인가?
```

```
$('#Turtle*0:card').d === 17;
```

```
// 카드 색깔 패턴이 초록색 빨간색 인가?
```

```
$('#Turtle*0:card').d === 25;
```

```
// 카드 색깔 패턴이 청록색 빨간색 인가?
```

```
$('#Turtle*0:card').d === 33;
```

```
// 카드 색깔 패턴이 파란색 빨간색 인가?
$('Turtle*0:card').d === 41;

// 카드 색깔 패턴이 자홍색 빨간색 인가?
$('Turtle*0:card').d === 49;

// 카드 색깔 패턴이 자홍색 자홍색 인가?
$('Turtle*0:card').d === 54;
```

## 파이썬 코드

```
# 카드 색깔 패턴이 빨간색 빨간색 인가?
__('Turtle*0:card').d == 9

# 카드 색깔 패턴이 빨간색 노란색 인가?
__('Turtle*0:card').d == 10

# 카드 색깔 패턴이 빨간색 초록색 인가?
__('Turtle*0:card').d == 11

# 카드 색깔 패턴이 빨간색 청록색 인가?
__('Turtle*0:card').d == 12

# 카드 색깔 패턴이 빨간색 파란색 인가?
__('Turtle*0:card').d == 13

# 카드 색깔 패턴이 빨간색 자홍색 인가?
__('Turtle*0:card').d == 14

# 카드 색깔 패턴이 노란색 빨간색 인가?
__('Turtle*0:card').d == 17

# 카드 색깔 패턴이 초록색 빨간색 인가?
```

```
__('Turtle*0:card').d == 25
```

# 카드 색깔 패턴이 청록색 빨간색 인가?

```
__('Turtle*0:card').d == 33
```

# 카드 색깔 패턴이 파란색 빨간색 인가?

```
__('Turtle*0:card').d == 41
```

# 카드 색깔 패턴이 자홍색 빨간색 인가?

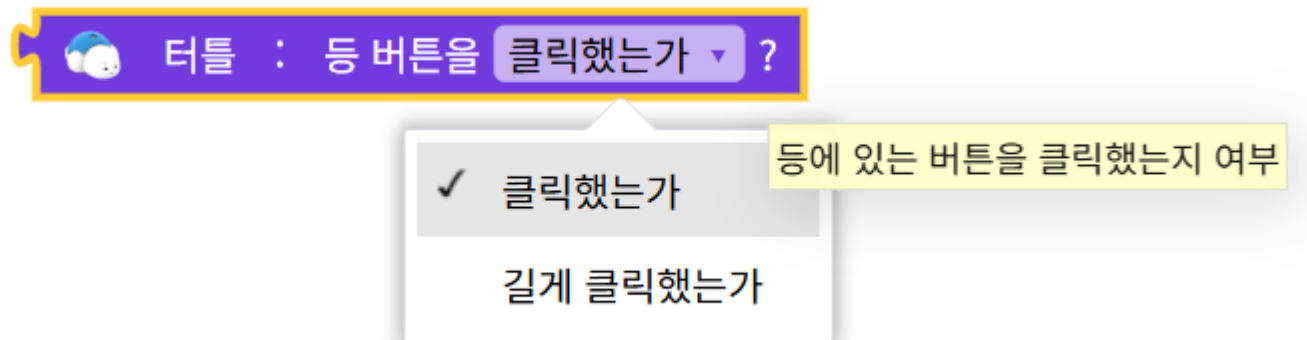
```
__('Turtle*0:card').d == 49
```

# 카드 색깔 패턴이 자홍색 자홍색 인가?

```
__('Turtle*0:card').d == 54
```

## 등 버튼을 클릭했는가?

등 버튼 클릭 여부에 따라 참 (1) / 거짓 (0) (으) 로 반환합니다.



## 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
action	드롭다운 옵션	버튼 클릭 방식	클릭했는가 (click), 길게 클릭했는가 (long_click)

## 자바스크립트 코드

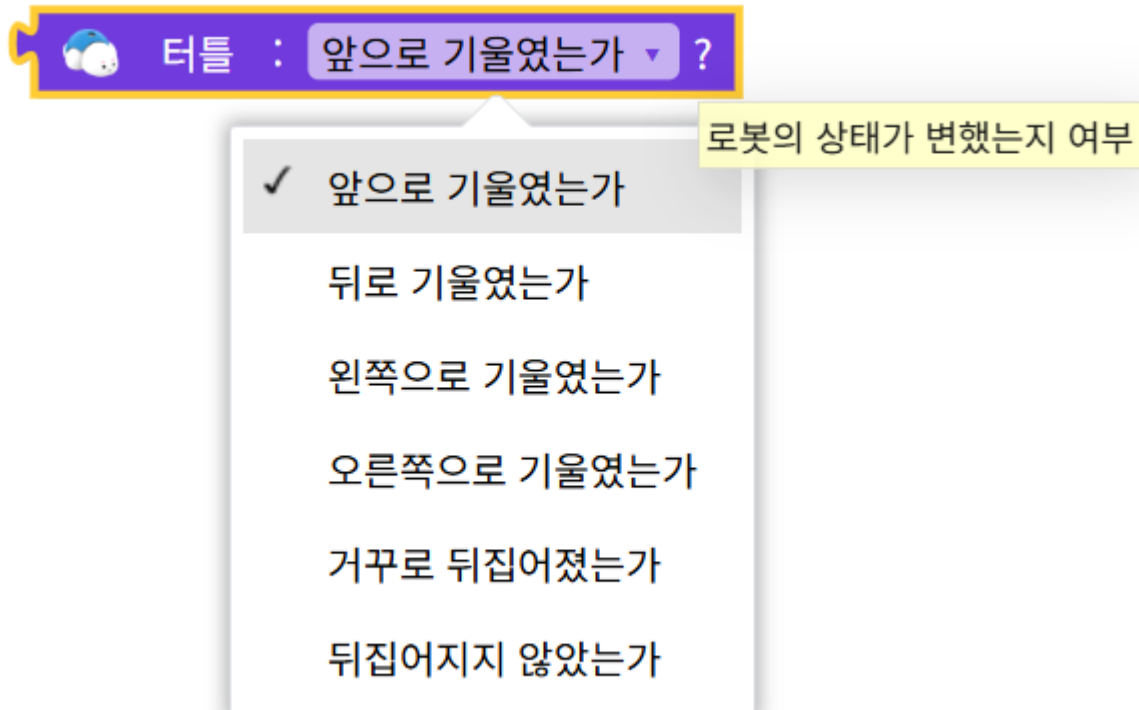
```
// 등 버튼을 클릭했는가?  
$('Turtle*0:button.click').e;  
  
// 등 버튼을 길게 클릭했는가?  
$('Turtle*0:button.long_click').e;
```

## 파이썬 코드

```
# 등 버튼을 클릭했는가?  
__('Turtle*0:button.click').e  
  
# 등 버튼을 길게 클릭했는가?  
__('Turtle*0:button.long_click').e
```

## 상태 변경 여부

터틀의 상태 변경 여부를 참 (1) / 거짓 (0) (으) 로 반환합니다.



## 드롭다운 옵션 및 입력값

이름	구분	설명	범위 / 종류
condition	드롭다운 옵션	위치 조건	앞으로 기울였는가, 뒤로 기울였는가, 왼쪽으로 기울였는가, 오른쪽으로 기울였는가, 거꾸로 뒤집어졌는가, 뒤집어지지 않았는가

## 자바스크립트 코드

```
// 터틀이 앞으로 기울였는가?  
$('Turtle*0:acceleration.x').d > 5000;  
  
// 터틀이 뒤로 기울였는가?  
$('Turtle*0:acceleration.x').d < -5000;  
  
// 터틀이 왼쪽으로 기울였는가?  
$('Turtle*0:acceleration.y').d > 5000;  
  
// 터틀이 오른쪽으로 기울였는가?  
$('Turtle*0:acceleration.y').d < -5000;  
  
// 터틀이 거꾸로 뒤집어졌는가?  
$('Turtle*0:acceleration.z').d > 0;  
  
// 터틀이 뒤집어지지 않았는가?  
$('Turtle*0:acceleration.z').d < -3000;
```

## 파이썬 코드

```
# 터틀이 앞으로 기울였는가?  
__('Turtle*0:acceleration.x').d > 5000
```

```
# 터틀이 뒤로 기울었는가?
__('Turtle*0:acceleration.x').d < -5000

# 터틀이 왼쪽으로 기울었는가?
__('Turtle*0:acceleration.y').d > 5000

# 터틀이 오른쪽으로 기울었는가?
__('Turtle*0:acceleration.y').d < -5000

# 터틀이 거꾸로 뒤집어졌는가?
__('Turtle*0:acceleration.z').d > 0

# 터틀이 뒤집어지지 않았는가?
__('Turtle*0:acceleration.z').d < -3000
```