

ROS\_1일차\_과제1번\_보고서\_20기\_인턴\_2025402040\_컴퓨터정보공학부\_  
노기문

1. 토픽 설명
2. 통신방법

## 1. 토픽 발행

직진 :

```
ros2 topic pub --once /turtle1/cmd_vel geometry_msgs/msg/Twist W  
"{linear: {x: 2.0, y: 0.0, z: 0.0}, angular: {x: 0.0, y: 0.0, z: 0.0}}"
```

=> x축 방향으로 (거북이가 바라보는 방향이 x) 2.0만큼 이동 하라  
선속도  $x=2.0 \rightarrow$  앞으로 일정 거리 직진.  
각속도  $z=0.0 \rightarrow$  회전 없음.

제자리 90도 회전 :

```
ros2 topic pub --once /turtle1/cmd_vel geometry_msgs/msg/Twist W  
"{linear: {x: 0.0, y: 0.0, z: 0.0}, angular: {x: 0.0, y: 0.0, z: 1.57}}"
```

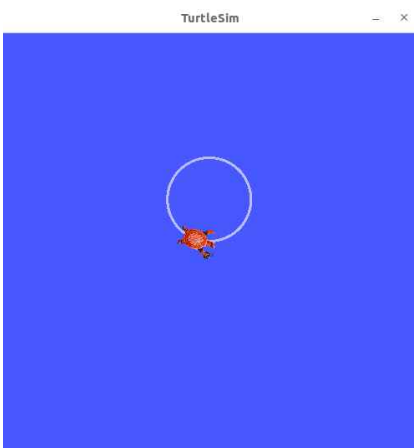
=> 선속도 0.0  $\rightarrow$  제자리 정지.  
각속도  $z=1.57 \text{ rad/s} (\approx 90^\circ/\text{s}) \rightarrow$  초당 90도 회전.

120도 회전 :

```
# 120도 회전  
ros2 topic pub --once /turtle1/cmd_vel geometry_msgs/msg/Twist W  
"{linear: {x: 0.0, y: 0.0, z: 0.0}, angular: {x: 0.0, y: 0.0, z: 2.09}}"
```

위와 동일하지만 120도의 라디안은 2.09

원



```
ros2 topic pub -r 30 /turtle1/cmd_vel geometry_msgs/msg/Twist W  
"{linear: {x: 2.0, y: 0.0, z: 0.0}, angular: {x: 0.0, y: 0.0, z: 1.8}}"
```

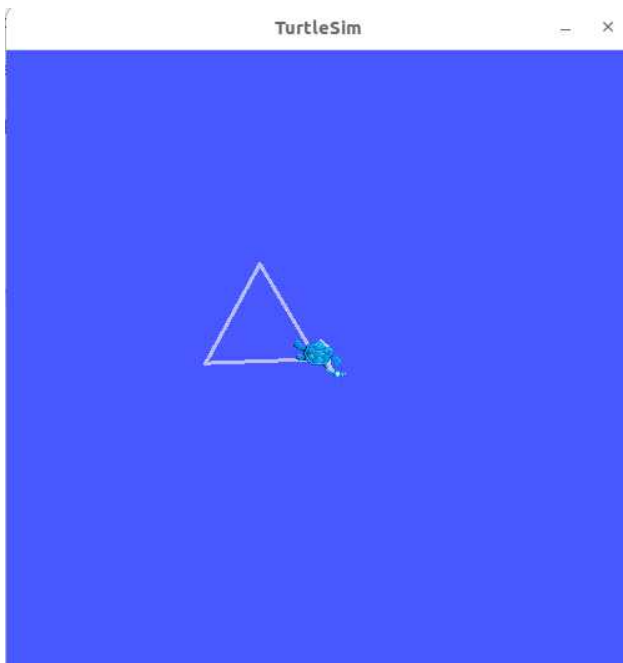
-r 30  $\rightarrow$  초당 30Hz로 반복 발행.  
 $\text{linear.x}=2.0 \rightarrow$  전진 속도.  
 $\text{angular.z}=1.8 \rightarrow$  동시에 회전.

### 사각형



직진 + 제자리 90도 회전 => 4회 반복

### 삼각형



직진 + 제자리 120도 회전 => 3회 반복

## 2. ROS2의 통신 방법

사용자 측 내가 토픽을 입력하면 turtlesim\_node 내부의 구독 콜백이 /turtle1/cmd\_vel 메시지를 받는다.

linear.x, angular.z 값을 읽어 거북이의 속도 상태를 갱신하고

물리 시뮬레이션이 작용하여

매 주기마다 내부에서 (x,y,theta) 위치를 적분한다음

linear.x는 전진 속도 → x, y 좌표 변화와

angular.z는 각속도 → 방향(theta) 변화한다.

마지막으로 그래픽 업데이트하며

새 좌표에 맞게 화면에 거북이 위치와 궤적을 그려준다.

- 끝 -