

<HW1>

과제 : QT GUI ros 패키지로 turtlesim 조종

1. 과제 조건

- 1 일차 1 번, 2 번 과제 기능 전부 포함
- QPushButton 사용
- cmd_vel 값 gui 출력

2. 개발 과정 및 프로그램 구조

2 - 1 ui



ui 구성은 다음과 같다. 각 버튼과 스핀 박스의 기능들은 qlabel 로 표시했다.

2 - 2 위젯 연결 및 통신

- 생성자 (Mainwindow)

먼저 생성자에서 ui 설정들(spinBox 범위 설정, pushButten, label 이름 설정, icon 설정)을 모두 초기화 하고 connect 함수를 이용해 발생할 시그널을 적절한 슬롯과 연결해 주었다. 또한 1 초마다 PrintCmd_vel 슬롯 함수를 호출하도록 했다.

- 생성자 (Qnode)

이 생성자에서는 퍼블리셔와 클라이언트, 노드를 생성하고, 변수들을 초기화한다.

- 버튼, 스핀박스

각각의 버튼과 스핀박스는 `qnode.cpp` 파일에 있는 함수를 호출해 명령을 처리한다.

먼저 `setBackgroundColor` 함수이다 이 함수는 `r, g, b` 값을 매개변수로 받아서 배경색 파라미터를 설정한다. 설정한 파라미터를 비동시방식으로 요청을 전송해 변경사항을 반영한다.

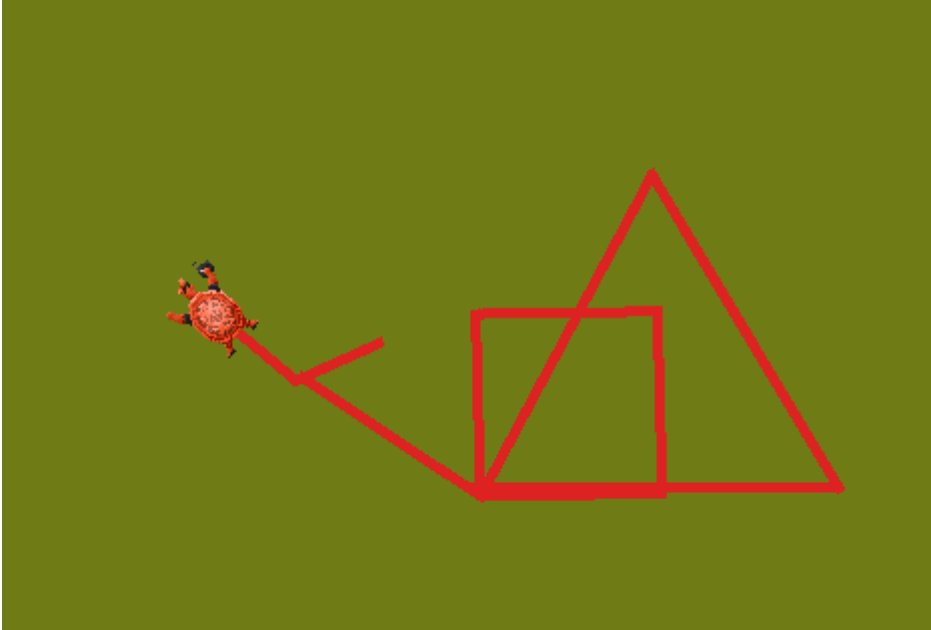
다음은 `setPenColorLength` 함수이다 이 함수는 `r, g, b, width` 값을 받아와 값을 설정하고 비동기 방식으로 서비스를 요청해 설정된 펜 속성에 따라 거북이의 경로 색상과 굵기가 설정된다.

다음은 `publishVelocity` 함수이다. 이 함수는 `linear_x, angular_z` 값을 매개변수로 받아오고 (각각은 전진/후진속도, 회전속도를 의미함) 그 값을 퍼블리셔를 통해 토픽에 발행하여 적절한 움직임을 만들 수 있다.

3.실행

실행화면은 아래와 같다.





4. 배운점

qt ui 와 ros2 를 결합해서 qt 애플리케이션이 ros2 노드와 통신하는 방법을 배울 수 있었다.

<HW2>

과제 : Qt 로 string 입력 받아 버튼 눌러 talker/listener 노드 제작

1. 과제 조건

- 버튼 눌러서 퍼블리시
- 서브스크라이브 하면 라벨값 변경

2. 개발 과정 및 프로그램 구조

2 – 1 생성자

생성자에서는 라벨의 이름을 정의해주고 connect 함수를 통해 시그널을 처리할 슬롯을 정해준다.

2 – 2 퍼블릭

버튼을 클릭하면 qnode 에 정의되어있는 publishCounter 함수로 이동하고 init 함수에서 생성한 퍼블리셔를 활용해 메시지를 퍼블리쉬한다. 이때 퍼블릭 여부를 확인하기 위해 count 변수를 선언해두고 publish 를 할때마다 1 씩 증가하도록 설정한다.

2 – 3 서브스크라이버

서브스크라이버에서는 수신한 메시지를 QLabel 을 통해 ui 에 나타나게 한다.

3.실행 결과

