

ROS2 2일차 보고서

목차

1. 코드 분석
2. 실행 결과
3. 참고 문헌

1. 코드 분석

****publisher****

class MinimalPublisher : public rclcpp::Node

- MinimalPublisher 클래스를 만들고 rclcpp::Node 를 상속

Node("minimal_publisher"), count_(0)

- 노드 이름을 설정해주고 클래스 안 변수를 초기화

publisher_ = this->create_publisher<custom_interfaces::msg::AddTwoInts>

("add_two_ints", 10);

- custom_interfaces::msg::AddTwoInts 로 메시지 타입을 정해주고 토픽이름과 버퍼크기를 정해줌

timer_ = this->create_wall_timer(500ms, std::bind(&MinimalPublisher::timer_callback, this));

- 0.5 초마다 timer_callback 함수를 호출

auto message = custom_interfaces::msg::AddTwoInts();

- custom msg 를 저장할 변수 선언 auto 로 type 이 자동으로 저장됨

message.a = this->count_++;, message.b = {1, 2, 3};

- 정수형 데이터와 벡터 데이터

RCLCPP_INFO(this->get_logger(), "Publishing: a=%ld", message.a);

- a 를 보내고 있음을 로그 출력

```
publisher_->publish(message);
```

- 메시지를 보냄

```
rclcpp::spin(std::make_shared<MinimalPublisher>());
```

- 노드 실행 유지

```
**subscriber**
```

```
class MySubscriber : public rclcpp::Node
```

- MySubscriber 클래스를 만들고 rclcpp::Node 를 상속

```
MySubscriber() : Node("my_subscriber")
```

- 노드 이름 설정

```
sub_ = this->create_subscription<custom_interfaces::msg::AddTwoInts>
```

- 받을 메시지 타입을 custom_interfaces::msg::AddTwoInts 로 정해줌

```
("add_two_ints", 10, std::bind(&MySubscriber::msg_callback, this,  
std::placeholders::_1));
```

- 토픽이름과 버퍼크기를 publisher 와 맞춰주고 callback 함수 설정

```
for (const auto& val : msg.b) {b_str += std::to_string(val) + " "};
```

- 벡터 b 에 있는 정보를 가져와 b_str 에 추가

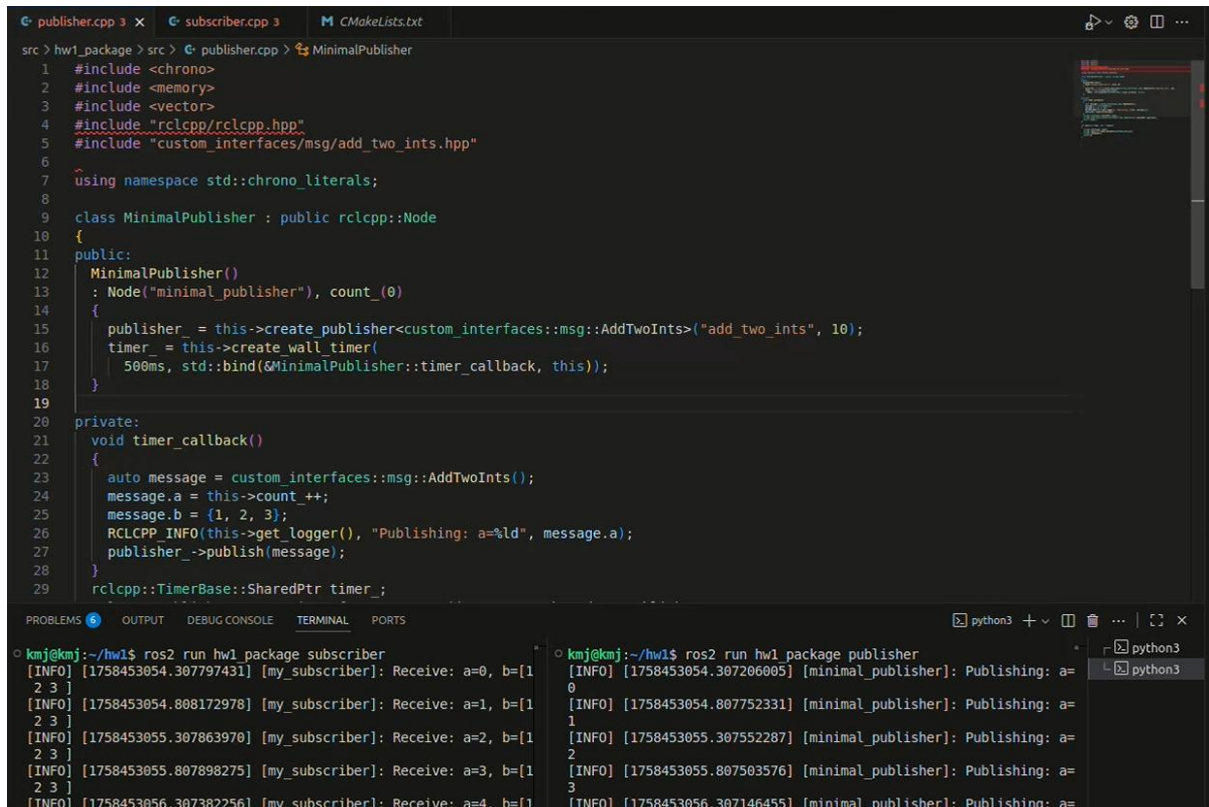
```
RCLCPP_INFO(this->get_logger(), "Receive: a=%ld, b=[%s]", msg.a, b_str.c_str());
```

- a 와 b 값을 문자열로 변환하여 로그 출력(벡터는 바로 출력 X)

```
rclcpp::Subscription<custom_interfaces::msg::AddTwoInts>::SharedPtr sub_;
```

- 클래스 안 멤버변수 sub_ 선언

2. 실행 결과



```
src > hw1_package > src > publisher.cpp > MinimalPublisher
1 #include <chrono>
2 #include <memory>
3 #include <vector>
4 #include "rclcpp/rclcpp.hpp"
5 #include "custom_interfaces/msg/add_two_ints.hpp"
6
7 using namespace std::chrono_literals;
8
9 class MinimalPublisher : public rclcpp::Node
10 {
11 public:
12     MinimalPublisher()
13     : Node("minimal_publisher"), count_(0)
14     {
15         publisher_ = this->create_publisher<custom_interfaces::msg::AddTwoInts>("add_two_ints", 10);
16         timer_ = this->create_wall_timer(
17             500ms, std::bind(&MinimalPublisher::timer_callback, this));
18     }
19
20 private:
21     void timer_callback()
22     {
23         auto message = custom_interfaces::msg::AddTwoInts();
24         message.a = this->count_++;
25         message.b = {1, 2, 3};
26         RCLCPP_INFO(this->get_logger(), "Publishing: a=%ld", message.a);
27         publisher_->publish(message);
28     }
29     rclcpp::TimerBase::SharedPtr timer_;
30
31     int count_;
32     rclcpp::Publisher<custom_interfaces::msg::AddTwoInts> publisher_;
33     rclcpp::TimerBase::SharedPtr timer_;
34 }
```

```
kmj@kmj:~/hw1$ ros2 run hw1_package subscriber
[INFO] [1758453054.307797431] [my_subscriber]: Receive: a=0, b=[1
2 3]
[INFO] [1758453054.808172978] [my_subscriber]: Receive: a=1, b=[1
2 3]
[INFO] [1758453055.307863970] [my_subscriber]: Receive: a=2, b=[1
2 3]
[INFO] [1758453055.807898275] [my_subscriber]: Receive: a=3, b=[1
2 3]
[INFO] [1758453056.307382256] [my_subscriber]: Receive: a=4, b=[1
2 3]

kmj@kmj:~/hw1$ ros2 run hw1_package publisher
[INFO] [1758453054.307206005] [minimal_publisher]: Publishing: a=
0
[INFO] [1758453054.807752331] [minimal_publisher]: Publishing: a=
1
[INFO] [1758453055.307552287] [minimal_publisher]: Publishing: a=
2
[INFO] [1758453055.807503576] [minimal_publisher]: Publishing: a=
3
[INFO] [1758453056.307146455] [minimal_publisher]: Publishing: a=
4
```

- custom msg 에 있는 정수 a 와 vector b 가 토픽으로 데이터가 전송되는 모습

3. 참고 문헌

[Creating custom msg and srv files — ROS 2 Documentation: Humble documentation](#)