

ROS2 day2 hw3결과보고서

2025407012/로봇학부/송연우

목차

1. Hpp

(1) Mainwindow

(2) qnode

2. Cpp

(1) mainwindow

(2) qnode

3. 실행 화면

1. Hpp

(1) Mainwindow

헤더파일에서의 선언부입니다. 버튼 2개에 대한 슬롯 함수만을 추가했습니다.

```
20      /
27  ✓ class MainWindow : public QMainWindow
28      {
29          Q_OBJECT
30
31      public:
32          Ui::MainWindowDesign* ui;
33          MainWindow(QWidget* parent = nullptr);
34          ~MainWindow();
35          QNode* qnode;
36          void label_2(QString *msg);
37
38      private slots:
39          void on_pushButton_clicked();
40
41          void on_pushButton_2_clicked();
42
43      private:
44          void closeEvent(QCloseEvent* event);
45      };
```

(2) Qnode

헤더파일 선언부입니다. 발행자, 구독자, 시그널 함수만을 추가했습니다.

```
//  
40  class QNode : public QThread  
41  {  
42      Q_OBJECT  
43  public:  
44      QNode();  
45      ~QNode();  
46      void topic_callback(const std_msgs::msg::String::SharedPtr msg);  
47      void publishing(QString text);  
48      //void setMessage(QString str);  
49  
50  protected:  
51      void run();  
52  
53  private:  
54      std::shared_ptr<rcclcpp::Node> node;  
55      rcclcpp::Publisher<std_msgs::msg::String>::SharedPtr publisher;  
56      rcclcpp::Subscription<std_msgs::msg::String>::SharedPtr subscriber;  
57  
58      Q_SIGNALS:  
59          void rosShutDown();  
60          void receivedMessage(QString msg);  
61  };
```

2. cpp파일

(1) mainwindow

```

14  ✓ MainWindow::MainWindow(QWidget* parent) : QMainWindow(parent), ui(new Ui::MainWindowDesign)
15  {
16      ui->setupUi(this);
17
18      QIcon icon(":/ros-icon.png");
19      this->setWindowIcon(icon);
20
21      qnode = new QNode();
22
23      QObject::connect(qnode, SIGNAL(rosShutDown()), this, SLOT(close()));
24      connect(qnode, &QNode::receivedMessage, this, [this](QString msg){
25          ui->label_2->setText(msg);
26      });
27      qnode->start();
28  }
29
30  void MainWindow::closeEvent(QCloseEvent* event)
31  {
32      QMainWindow::closeEvent(event);
33  }
34
35  MainWindow::~MainWindow()
36  {
37      delete ui;
38  }
39
40
41  ✓ void MainWindow::on_pushButton_clicked()
42  {
43      QString text = ui->textEdit->toPlainText();
44      // qnode->setMessage(text);
45      qnode->publishing(text);
46      ui->label->setText(text);
47
48  }

```

Mainwindow의 소스파일 중 생성자에서는 메시지를 수신했을 때의 시그널과 label값을 변경시키는 슬롯함수를 묶는 코드를 추가했습니다. 나머지 버튼은 함수는 있으나 사용하지 않았습니다.

(2) qnode

```
20  ✓ QNode::QNode()
21  {
22      int count=0;
23      int argc = 0;
24      char** argv = NULL;
25      rclcpp::init(argc, argv);
26      node = std::make_shared<rclcpp::Node>("qnode");
27      publisher = node->create_publisher<std_msgs::msg::String>("topicname", 10);
28      subscriber = node->create_subscription<std_msgs::msg::String>(
29          "topicname",
30          10,
31          std::bind(&QNode::topic_callback, this, std::placeholders::_1));
32  }
33
```

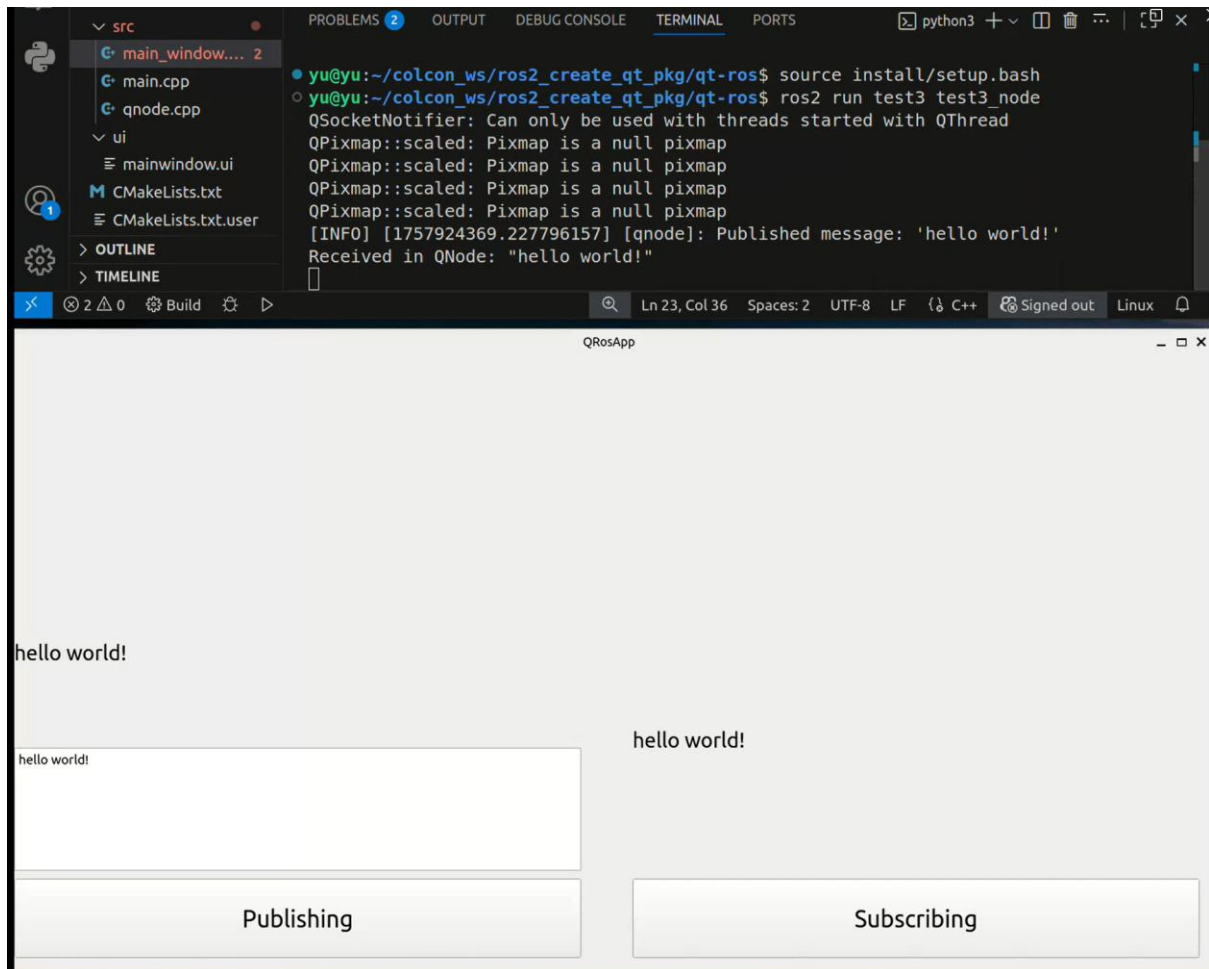
위 사진은 Qnode 소스파일에서의 생성자 정의 부분입니다.

```
55  void QNode::topic_callback(const std_msgs::msg::String::SharedPtr msg) {
56      qDebug() <<"Received in QNode:" << QString::fromStdString(msg->data);
57      emit receivedMessage(QString::fromStdString(msg->data));
58  }
59
60  ✓ void QNode::publishing(QString text){
61      auto msg = std_msgs::msg::String();
62      msg.data = text.toStdString();
63      publisher->publish(msg);
64      RCLCPP_INFO(node->get_logger(), "Published message: '%s'", msg.data.c_str());
65  }
```

소멸자와 run함수를 지나 있는 topic_callback()함수와 publishing()함수입니다.

Topic_callback()함수에서는 메시지를 수신하고 나서 시그널을 방출하는 코드를 추가해 Mainwindow클래스에서 label값을 변경할 수 있도록 했습니다.

3. 실행 화면



실행화면입니다. Subscribing 버튼에는 다른 기능은 없으며 publishing 버튼을 누르면 textedit박스의 문자열이 자동으로 발행되고, 이를 수신한 뒤 subscribing버튼 위 label에 표시됩니다.