

# QT 과제 1

3DOF 2D 로봇 팔 시뮬레이터

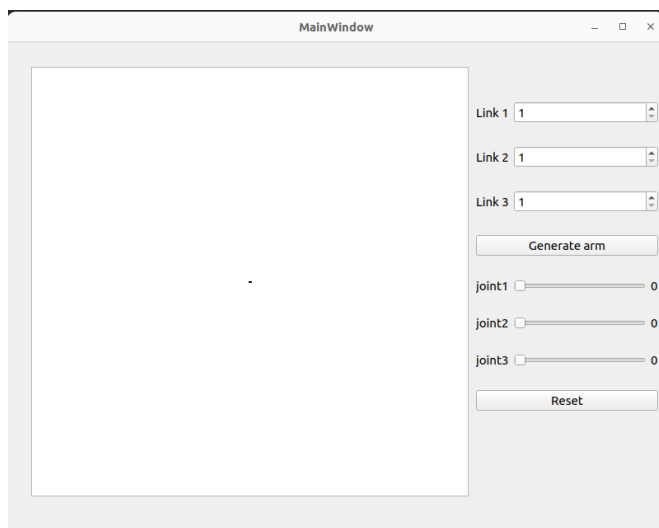
로봇게임단 휴머팀

19기 예비 단원

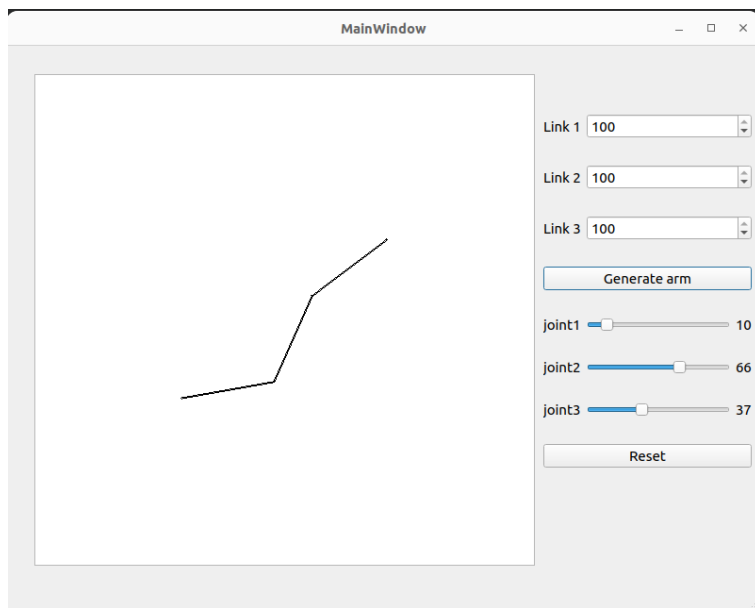
이원준

## 1. UI 사진

- MainWindow.ui



- 위젯 실행 예시



- 우측 스펀박스를 통해 로봇 팔 길이를 입력 받고 버튼을 눌러 UI Visual 영역에서 생성한다.
- 슬라이더를 직접 돌려서 사용자가 직접 joint 각도를 제어할 수 있다.
- 리셋 버튼을 통해 Visual 영역을 초기화 할 수 있다.
- 링크 길이는 1에서 100 사이로 조절할 수 있다.

## 2. 코드 설명

- 이 코드의 핵심적인 요소인 Draw\_frame 멤버 함수에 대해 설명하겠다.

```

MainWindow::joint_pos MainWindow::Draw_frame(float x,float y,int lenght, float angle)
{
    float x1 = x + lenght * cos(angle);
    float y1 = y - lenght * sin(angle);
    QGraphicsLineItem * line1 = new QGraphicsLineItem(x,y,x1,y1);
    line1->setPen(pen);
    scene->addItem(line1);

    float x2 = x + widthsize * cos(angle + M_PI/2.0);
    float y2 = y - widthsize * sin(angle+ M_PI/2.0);
    QGraphicsLineItem * line2 = new QGraphicsLineItem(x,y,x2,y2);
    line2->setPen(pen);
    scene->addItem(line2);

    float x3 = x + widthsize * cos(angle + M_PI/2.0) + lenght * cos(angle);
    float y3 = y - widthsize * sin(angle+ M_PI/2.0) - lenght * sin(angle);
    QGraphicsLineItem * line3 = new QGraphicsLineItem(x2,y2,x3,y3);
    line3->setPen(pen);
    scene->addItem(line3);

    QGraphicsLineItem * line4 = new QGraphicsLineItem(x1,y1,x3,y3);
    line4->setPen(pen);
    scene->addItem(line4);

    joint_pos jointpos;

    jointpos.x = (x1+x3) / 2.0;
    jointpos.y = ((y1+y3) / 2.0) + widthsize / 2.0;

    return jointpos;
}

```

- 이 함수는 멤버 구조체로 선언된 joint\_pos를 리턴으로 받는 함수로 로봇팔을 그리기 위해 필요한 요소인 시작지점의 좌표와 팔의 길이 및 각도를 파라미터로 받는다.
- Ui로 입력받은 각도를 호도법으로 변환하고 삼각함수를 활용해 로봇팔을 작성한다.
- 이때 QGraphicsLineItem으로 포인터 선언을 하고 QGraphicsScene 포인터인 scene의 멤버함수를 활용해 Ui에 그림을 추가한다. 이와 같은 방법으로 ui에 그림을 그린다.
- 다음 로봇 팔의 시작점을 계산하고 구조체 형식으로 리턴 해준다.
- 리턴 받은 좌표를 다시 파라미터로 집어넣어 연속된 로봇팔을 출력할 수 있도록 세팅한다.

- 이러한 작업은 activate\_arm() 멤버함수 내부에서 반복적으로 수행되어 전체 로봇 팔을 즉각적으로 출력할 수 있다.
- 해당 함수는 다음 사진을 참고하자.

```
void MainWindow::activate_arm()
{
    int lenght_frame1 = ui->link1->value();
    int lenght_frame2 = ui->link2->value();
    int lenght_frame3 = ui->link3->value();

    float angle_frame1 = ((float)ui->joint1->value()) * M_PI / 180.0;
    float angle_frame2 = ((float)ui->joint2->value()) * M_PI / 180.0;
    float angle_frame3 = ((float)ui->joint3->value()) * M_PI / 180.0;

    scene->clear();
    joint_pos jointpos2 = Draw_frame(start_x,start_y,lenght_frame1,angle_frame1);
    joint_pos jointpos3 = Draw_frame(jointpos2.x,jointpos2.y,lenght_frame2,angle_frame2);
    Draw_frame(jointpos3.x,jointpos3.y,lenght_frame3,angle_frame3);
}
```