

DAY2

과제 2번 보고서

2023741024
로봇학부 박건후

목차

- (1) cupview.cpp
- (2) mainwindow.cpp
- (3) 시연 사진

(1) cupview.cpp

```
public slots:
    void refill();           // 버튼에서 호출

private slots:
    void startSuck();        // 2초마다 빨기
    void onAnimationTick(); // 슬러그 이동
    void onRefillTick();     // 리필 애니메이션

private:
    // 상태
    double level = 0.8;      // 0.0~1.0 : 컵 안의 수위
    bool sucking = false;    // 빨기 중 여부
    double progress = 1.0;   // 슬러그 이동 진행도(0~1)
    double refillTarget = 1.0; // 끝까지 리필 되도록 설정
    void paintEvent(QPaintEvent*) override;
    // 타이머
    QTimer pulseTimer;       // 2초 주기 빨기
    QTimer animTimer;        // 슬러그 이동 30ms
    QTimer refillTimer;      // 리필 20ms
};
```

개략적으로 설명드리기 위해 cupview 헤더 파일 부분을 가져왔습니다. 제가 이번에 만든 것은 큐를 활용하여 간단한 ui를 만들었습니다. 컵에 있는 음료를 빨대로 빨면 밑에 있는 음료가 올라가고, 음료 윗부분이 내려가고 이 때 refill버튼을 누르면 리필이 되는 것입니다. 그래서 먼저 들어간 것이 빨대를 통해 나오게 되고 리필 버튼을 누르면 새 음료가 컵으로 들어가는 것으로 큐의 구조를 사용하여 만든 것입니다.

버튼을 누르면 리필을 해야 하기에 refill()슬롯 메서드를 만들었고 timeout에 따라 슬롯 메서드가 호출되어 그림이 작동되게 하는 private slots 메서드들 그리고 음료의 상태, 리필의 채워지는 정도 등의 상태값을 저장하는 변수들, 그리고 타이머 객체를 선언한 것입니다.

```

CupView::CupView(QWidget *parent) : QWidget(parent) {
    setMinimumSize(480, 320);

    pulseTimer.setInterval(2000);
    connect(&pulseTimer, &QTimer::timeout, this, &CupView::startSuck);
    pulseTimer.start(); //2초마다 빨기 시작

    //설정만 하고 대기
    animTimer.setInterval(30);
    connect(&animTimer, &QTimer::timeout, this, &CupView::onAnimTick);

    refillTimer.setInterval(20);
    connect(&refillTimer, &QTimer::timeout, this, &CupView::onRefillTick);
}

```

이제 cpp파일의 메서드 구현부를 보겠습니다. 이 부분은 생성자 부분으로 2초마다 음료가 빨대를 통해 올라오는 것과 타이머 설정을 하여 몇 ms주기로 움직이게 할지를 설정한 부분입니다.

```

void CupView::startSuck() {
    if (sucking || refillTimer.isActive()) return;
    if (level <= 0.05) return; // 거의 바닥이면 빨지 않음
    level = std::max(0.0, level - 0.05); // 한 번 빨 때 줄어드는 양
    progress = 0.0;
    sucking = true;
    animTimer.start();
    update();
}

void CupView::onAnimTick() {
    if (!sucking) { animTimer.stop(); return; }
    progress += 0.04; // 속도(값 키우면 빨라짐)
    if (progress >= 1.0) {
        progress = 1.0;
        sucking = false;
        animTimer.stop();
    }
    update();
}

```

빨대를 통해 음료 일부가 올라갈 때는 2초마다 이루어집니다. 그 2초가 지나면 타이머가 발생하며 빨대를 통해 음료 일부가 올라가야 합니다. 이에 대한 타이머를 설정하였고 그 타이머로 호출되는 슬롯 메서드가 startSuck이라는 함수입니다. 이 때 리필하는 동안에 빨 수는 없고 음료가 없을 때 또한 빨아도 음료가 없어 빨 수 없을 것입니다.

이에 대한 예외처리를 진행하였습니다. 그 이후에는 빠는 것을 할 텐데 빠는 도중인 것을 보여줄 메서드를 만들고 연속적으로 음료가 올라가는 모습을 보여주고자 타이머를 사용한 것입니다. 이에 대한 타이머는 animTimer입니다.

```
void CupView::onAnimTick() {
    if (!sucking) { animTimer.stop(); return; }
    progress += 0.04; // 속도(값 키우면 빨라짐)
    if (progress >= 1.0) {
        progress = 1.0;
        sucking = false;
        animTimer.stop();
    }
    update();
}
```

이 때 progress라는 변수를 사용한 이유를 보실 수 있습니다. 음료 일부가 이동하는 것을 끝내는 시점을 명확히 설정하기 힘들었기 때문입니다. 그래서 progress라는 변수를 두고 끝나는 임계점을 1로 설정하여 30ms당 올라가며 올라가는 속도를 조절하며 명확히 끝나는 시점을 잡을 수 있었습니다.

```
void CupView::refill() {
    // 현재에서 30% 채우기(최대 가득)
    refillTarget = std::min(1.0, level + 0.3);
    if (refillTarget <= level) return;
    refillTimer.start();
}

void CupView::onRefillTick() {
    level += 0.01; // 채우는 속도
    if (level >= refillTarget) { //다 채우면 타이머 멈춰서 채우기 멈춤
        level = refillTarget;
        refillTimer.stop();
    }
    update();
}
```

그리고 빨대로 올라가는 것을 구현했으니 리필하는 부분을 보실 차례입니다. 리필할 때는 현재 레벨을 현재 상황에 따라 적당히 올리는 것이 필요했습니다. 무조건 끝까지 채우는 것보다는 적당히 채우는 것이 좀 더 큐 상황에 적합해 보였기 때문입니다.

리필할 때 또한 연속적인 모습을 담고자 타이머를 사용했습니다. 빨대로 일부 음료가 올라가는 것을 구현하는 것과 같이 20ms당 level의 속도를 뒤서 목표 수위 기준값에 맞춰 원하는 속도로 올라가는 것을 구현하였습니다.

(2) mainwindow.cpp

```
: QMainWindow(parent), ui(new Ui::MainWindow)
{
    ui->setupUi(this);

    cupView = new CupView(this);
    auto *lay = new QVBoxLayout(ui->cupHost);
    lay->setContentsMargins(0,0,0,0);
    lay->addWidget(cupView);

    // 버튼 누르면 리필되는 메서드 호출
    connect(ui->refillButton, &QPushButton::clicked, cupView, &CupView::refill);
}

MainWindow::~MainWindow() { delete ui; }
```

이 부분은 MainWindow클래스의 생성자 구현 부분으로써 과제 1번과 같이 위젯 위에 객체를 올려 놓고 그림을 그릴 수 있도록 올려놓는 부분입니다. 그리고 리필 버튼을 누르면 리필하는 슬롯 메서드를 호출하도록 만들기로 설계했기 때문에 이에 대한 connect 함수를 작성하여 버튼을 누르면 리필이 되도록 만들었습니다.

(3) 시연 사진



이렇게 왼쪽 사진처럼 음료가 처음에 존재하다가 오른쪽 사진처럼 시간에 따라 음료 일부가 위로 올라가며 용량이 주는 것을 보실 수 있습니다.