게임 프로그래밍

HW2 보고서

박종승 교수님

컴퓨터 공학부

201401414 곽용준

∙ 키 조작

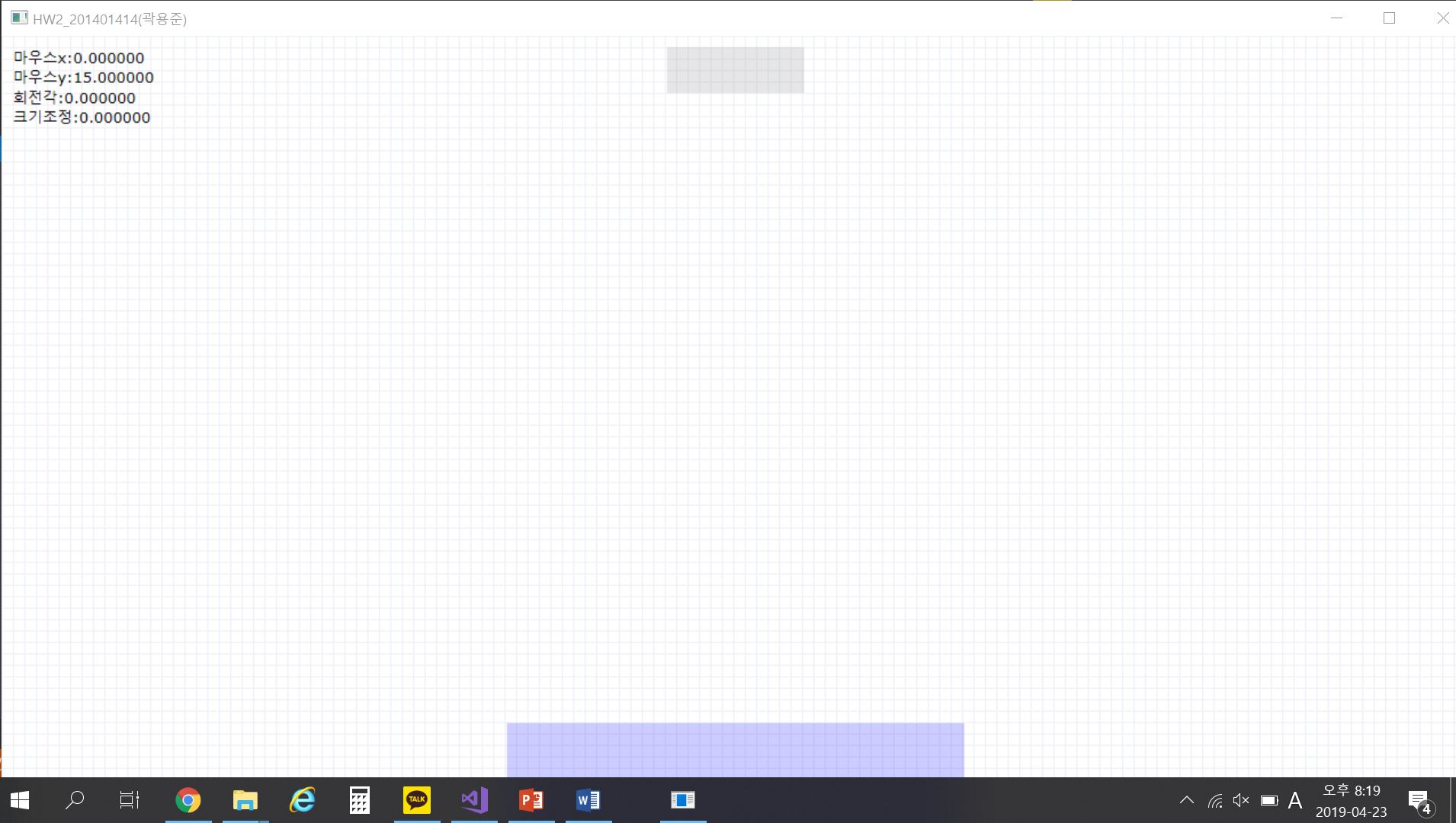
초기 화면은 상단에 임의의 희미한 회색 상자가, 하단에는 보라색 임의의 희미한 상자가 출력된다. 상단의 상자를 클릭하면 임의의 이름과 성적을 가지는 상자가 생성되고 이는 마우스를 따라 움직이며 하단의 상자와 가까워 질수록 크기가 하단 상자와 비슷해지고 회전하게 된다.

삭제 시에는 가장 상단의 상자만 삭제할 수 있으며, 생성과 반대로 드래그하여 상단 상자에 넣으면 삭제가 된다.

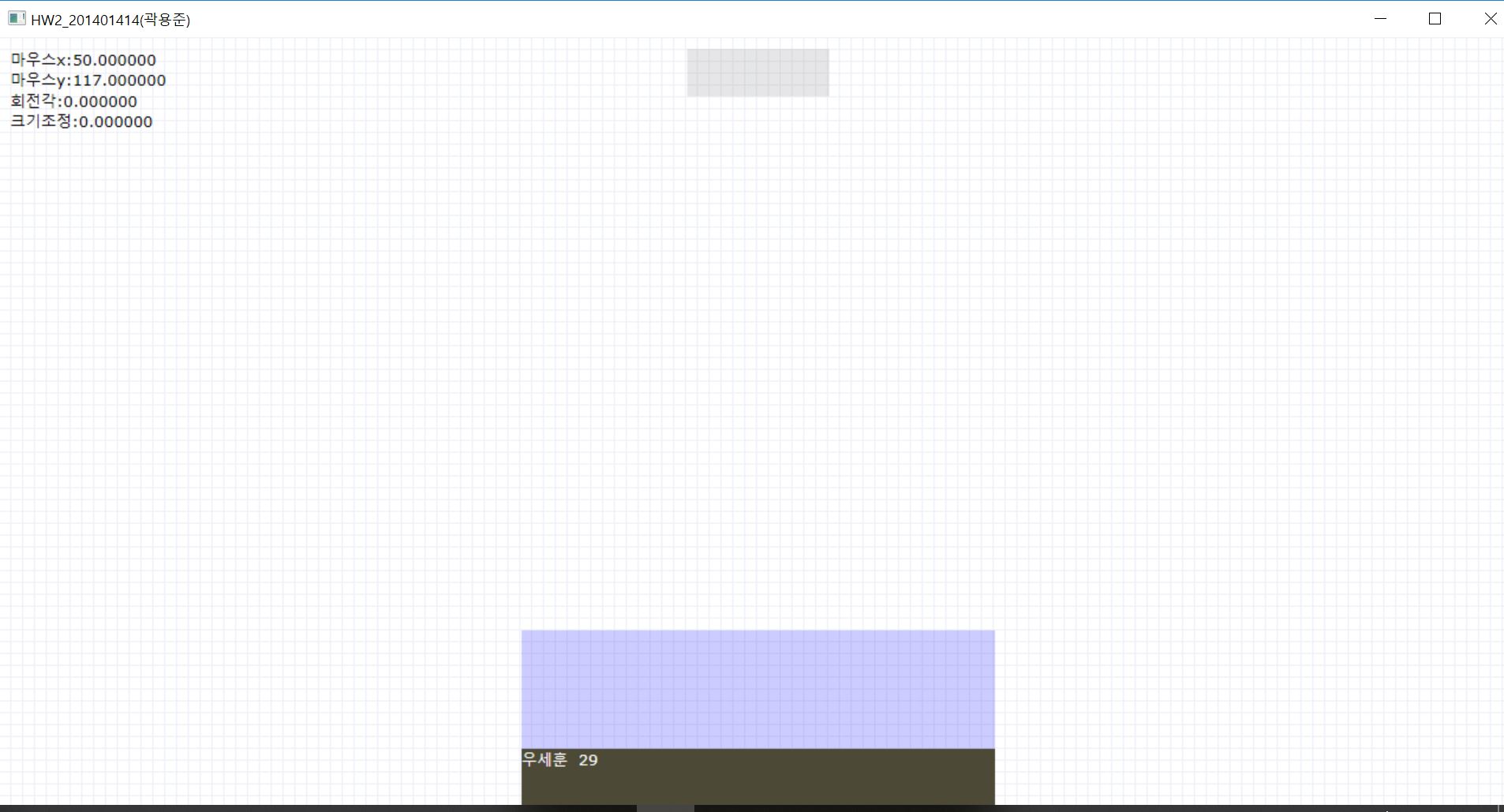
ESC를 누를 시에는 화면이 종료된다.

∙ 실행 결과

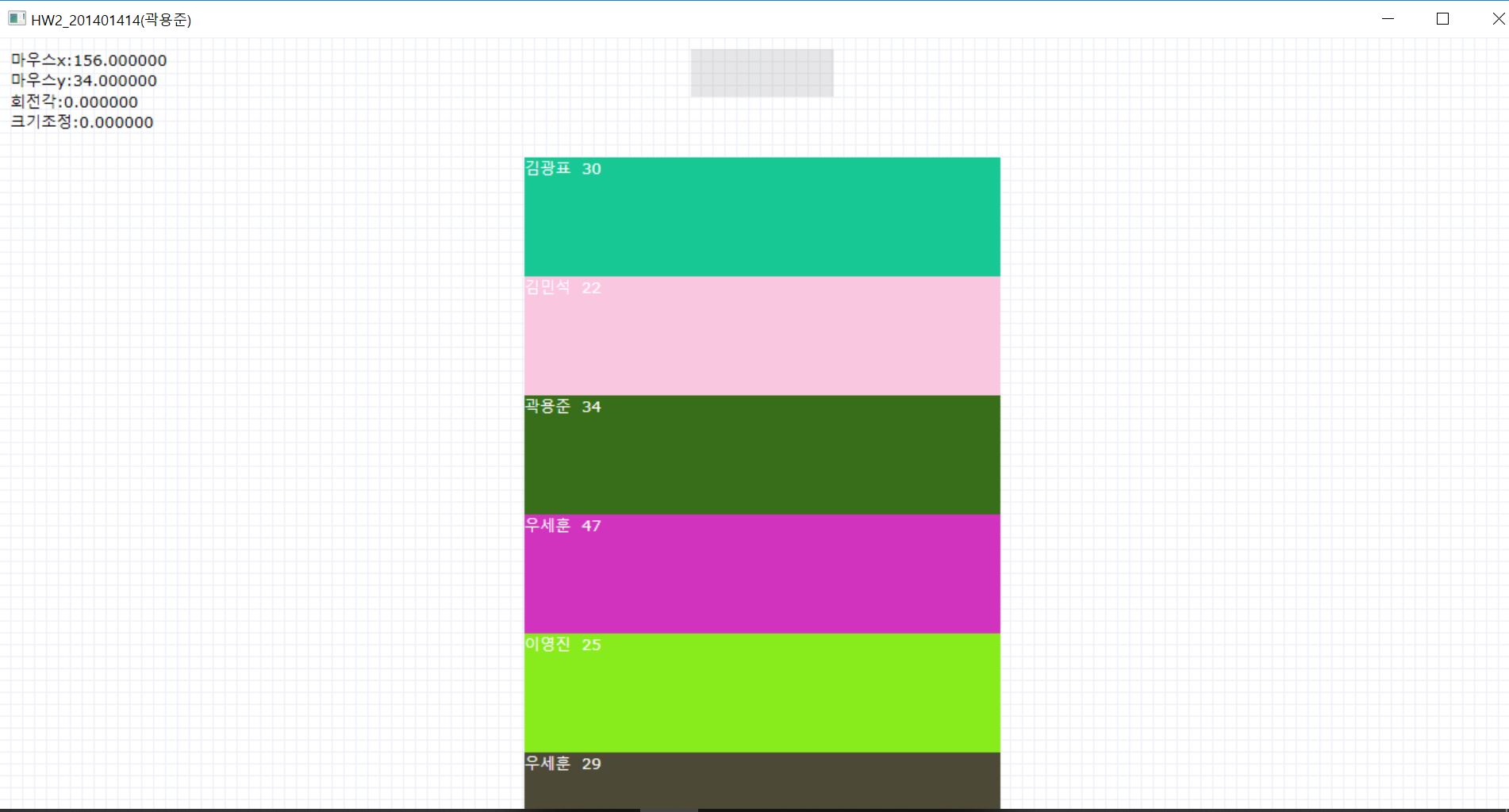
초기화면



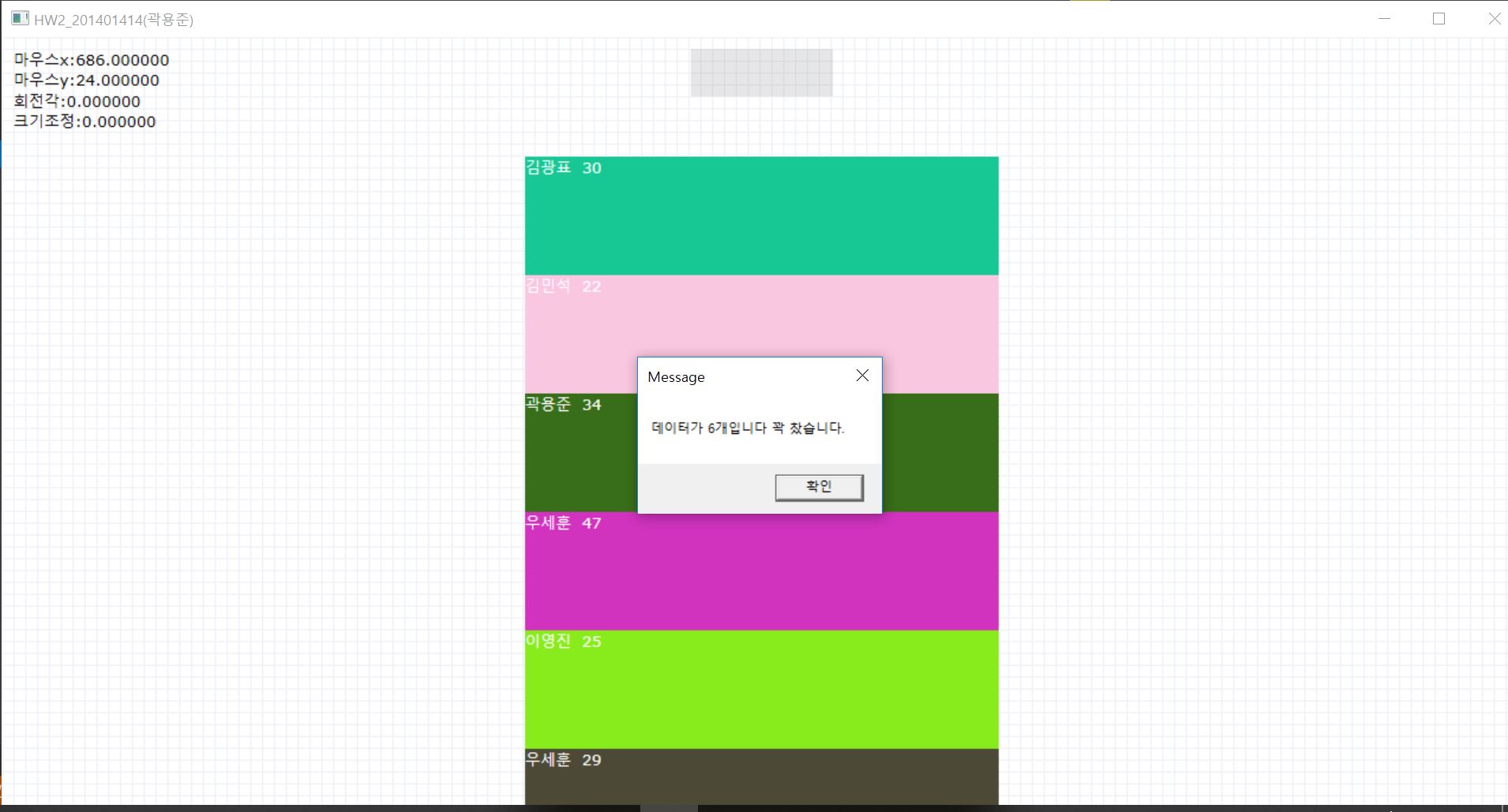
상자 입력(첫번째)



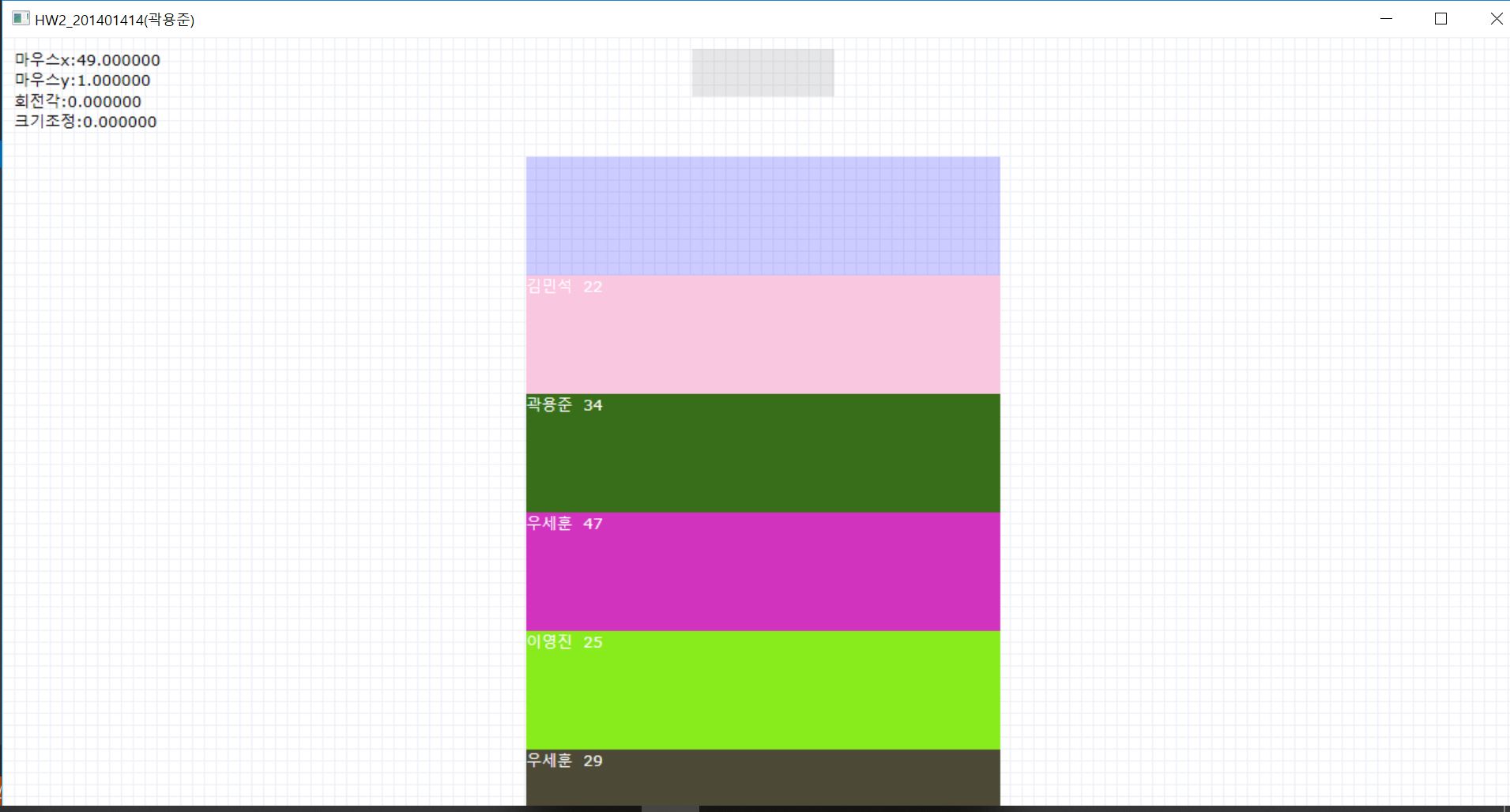
상자 입력(6개 ; Full)



상자 입력(오류)



상자 삭제



∙ 주요 코드

**드래깅 시에 회전 및 상자 배율 조정**

//드래깅 상자 move\_rate 핵심 만들어질 상가자 세로가 가로의 1/4

if (Moving\_box) {

D2D1\_RECT\_F temprectangle = D2D1::RectF(

current\_p.x\*width\_rate - move\_rate \* width\_rate, current\_p.y\*height\_rate - move\_rate \* 0.25\*height\_rate,

current\_p.x\*width\_rate + move\_rate \* width\_rate, current\_p.y\*height\_rate + move\_rate \* 0.25\*height\_rate);

//드래깅되는 상자의 현재 마우스 위치에 따른 위치 계산

if (Make\_box) {

Angle = - ((current\_p.y \* height\_rate - 30 \* height\_rate)

/ (30 \* height\_rate - (rtSize.height - ((0.5 \* HEIGHT \* height\_rate) \* (2\*sp + 3))))) \* 360;

} //상자 생성시에 회전각 계산

else if (Remove\_box) {

Angle = - ((current\_p.y \* height\_rate - 30 \* height\_rate)

/ (30 \* height\_rate - (rtSize.height - ((0.5 \* HEIGHT \* height\_rate) \* (2\*sp + 1))))) \* 360;

} //상자 삭제시에 회전각 계산

m\_pRenderTarget->SetTransform(D2D1::Matrix3x2F::Rotation(Angle, current\_p)); //상자 회전

m\_pRenderTarget->FillRectangle(&temprectangle, m\_pOriginalShapeBrush[boxs]); //상자 그리기

여기서 Moving\_Box는 상자 생성과 삭제시에 모두 True값을 가진다.

**마우스 위치 및 변환값 출력**

/\* 상자 인자 정보들 \*/

m\_pRenderTarget->SetTransform(D2D1::Matrix3x2F::Identity());

WCHAR szText[100];

swprintf\_s(szText, L"마우스x:%f\n마우스y:%f\n회전각:%f\n크기조정:%f\n", current\_p.x, current\_p.y, Angle, move\_rate);

m\_pRenderTarget->DrawText(

szText,

wcslen(szText),

m\_pTextFormat,

D2D1::RectF(10.0f, 10.0f, 150.0f, 150.0f),

m\_pTextBrush

); //좌상단에 쓰여지는 정보 쓰기