

객체지향프로그래밍 정리노트 (Week6_HWork2)

202004029 김정호, 202004040 노성민

Q1. 간단한 그래픽 편집기를 콘솔 바탕으로 만들어보자. 그래픽 편집기의 기능은 “삽입”, “삭제”, “모두 보기”, “종료”의 4가지이고, 실행과정은 다음과 같다.

```
그래픽 에디터입니다.
삽입:1, 삭제:2, 모두보기:3, 종료:4 >>1
선:1, 원:2, 사각형:3 >>1
삽입:1, 삭제:2, 모두보기:3, 종료:4 >>1
선:1, 원:2, 사각형:3 >>2
삽입:1, 삭제:2, 모두보기:3, 종료:4 >>1
선:1, 원:2, 사각형:3 >>3
삽입:1, 삭제:2, 모두보기:3, 종료:4 >>3
0: Line
1: Circle
2: Rectangle
삽입:1, 삭제:2, 모두보기:3, 종료:4 >>2
삭제하고자 하는 도형의 인덱스 >>1
삽입:1, 삭제:2, 모두보기:3, 종료:4 >>3
0: Line
1: Rectangle
삽입:1, 삭제:2, 모두보기:3, 종료:4 >>4
```

정답:

```
#ifndef LINE_H
#define LINE_H

#include "Shape.h"

class Line : public Shape {
protected:
    virtual void draw();
};

#endif
- Line.h -

#include "Line.h"
#include <iostream>
```

```
using namespace std;
```

```
void Line::draw() {  
    cout << "Line" << endl;  
}
```

- **Line.cpp** -

```
#ifndef CIRCLE_H  
#define CIRCLE_H
```

```
#include "Shape.h"
```

```
class Circle : public Shape {  
protected:  
    virtual void draw();  
};
```

```
#endif
```

- **Circle.h** -

```
#include "Circle.h"  
#include <iostream>
```

```
using namespace std;
```

```
void Circle::draw() {  
    cout << "Circle" << endl;  
}
```

- **Circle.cpp** -

```
#ifndef RECT_H  
#define RECT_H
```

```
#include "Shape.h"
```

```
class Rect : public Shape {  
protected:  
    virtual void draw();  
};
```

```
#endif
```

- **Rect.h** -

```

#include "Rect.h"
#include <iostream>

using namespace std;

void Rect::draw() {
    cout << "Rectangle" << endl;
}

```

- **Rect.cpp** -

```

#ifndef SHAPE_H
#define SHAPE_H

class Shape {
    Shape* next;

protected:
    virtual void draw() = 0;

public:
    Shape();
    virtual ~Shape();
    Shape* add(Shape* p);
    void paint();
    Shape* getNext();
};

#endif

```

-**Shape.h**-

```

#include "Shape.h"
#include <iostream>
using namespace std;
Shape::Shape() {
    next = NULL;
}
Shape::~Shape() {}
Shape* Shape::add(Shape* p) {
    this->next = p;
    return p;
}
void Shape::paint() {
    draw();
}

```

```
}  
Shape* Shape::getNext() {  
    return next;  
}
```

-Shape.cpp-

```
#ifndef UI_H  
#define UI_H  
  
class UI {  
public:  
    static int getMenu();  
    static int getShapeTypeToInsert();  
    static int getShapeIndexToDelete();  
};  
  
#endif
```

-UI.h-

```
#include "UI.h"  
#include <iostream>  
  
using namespace std;  
  
int UI::getMenu() {  
    int key;  
    cout << "삽입:1, 삭제:2, 모두보기:3, 종료:4 >>";  
    cin >> key;  
    return key;  
}  
  
int UI::getShapeTypeToInsert() {  
    int key;  
    cout << "선:1, 원:2, 사각형:3 >>";  
    cin >> key;  
    return key;  
}  
  
int UI::getShapeIndexToDelete() {  
    int key;  
    cout << "삭제하고자 하는 도형의 인덱스 >>";  
    cin >> key;  
    return key;  
}
```

```
}
```

-UI.cpp-

```
#ifndef GRAPHIC_EDITOR_H
#define GRAPHIC_EDITOR_H

#include "Shape.h"

class GraphicEditor {
    Shape* pStart;
    Shape* pLast;

public:
    GraphicEditor();
    void insertItem(int type);
    void deleteItem(int index);
    void show();
    void run();
};

#endif
```

-GraphicEditor.h-

```
#include "GraphicEditor.h"
#include "Line.h"
#include "Circle.h"
#include "Rect.h"
#include "UI.h"
#include <iostream>

using namespace std;

GraphicEditor::GraphicEditor() {
    pStart = pLast = NULL;
}

void GraphicEditor::insertItem(int type) {
    Shape* p = NULL;
    switch (type) {
        case 1:
            p = new Line();
            break;
        case 2:
```

```

        p = new Circle();
        break;
    case 3:
        p = new Rect();
        break;
    default:
        break;
}
if (pStart == NULL) {
    pStart = p;
    pLast = p;
    return;
}
pLast->add(p);
pLast = pLast->getNext();
}

void GraphicEditor::deleteItem(int index) {
    Shape* pre = pStart;
    Shape* tmp = pStart;
    if (pStart == NULL) {
        cout << "도형이 없습니다!" << endl;
        return;
    }
    for (int i = 0; i < index; i++) {
        pre = tmp;
        tmp = tmp->getNext();
    }
    if (tmp == pStart) {
        pStart = tmp->getNext();
        delete tmp;
    }
    else {
        pre->add(tmp->getNext());
        delete tmp;
    }
}

void GraphicEditor::show() {
    Shape* tmp = pStart;
    int i = 0;
    while (tmp != NULL) {
        cout << i++ << ": ";
        tmp->paint();
        tmp = tmp->getNext();
    }
}

```

```
}
```

```
void GraphicEditor::run() {  
    cout << "그래픽 에디터입니다." << endl;  
    int menu, index, type;  
    while (true) {  
        menu = UI::getMenu();  
        switch (menu) {  
            case 1:  
                type = UI::getShapeTypeToInsert();  
                insertItem(type);  
                break;  
            case 2:  
                index = UI::getShapeIndexToDelete();  
                deleteItem(index);  
                break;  
            case 3:  
                show();  
                break;  
            default:  
                return;  
        }  
    }  
}
```

-GraphicEditor.cpp-

```
#include "GraphicEditor.h"
```

```
int main() {  
    GraphicEditor graphicEditor;  
    graphicEditor.run();  
  
    return 0;  
}
```

- main.cpp -

결과:

그래픽 에디터입니다.

삽입:1, 삭제:2, 모두보기:3, 종료:4 >>1

선:1, 원:2, 사각형:3 >>1

삽입:1, 삭제:2, 모두보기:3, 종료:4 >>1

선:1, 원:2, 사각형:3 >>2

삽입:1, 삭제:2, 모두보기:3, 종료:4 >>1

선:1, 원:2, 사각형:3 >>3

삽입:1, 삭제:2, 모두보기:3, 종료:4 >>3

0: Line

1: Circle

2: Rectangle

삽입:1, 삭제:2, 모두보기:3, 종료:4 >>2

삭제하고자 하는 도형의 인덱스 >>1

삽입:1, 삭제:2, 모두보기:3, 종료:4 >>3

0: Line

1: Rectangle

삽입:1, 삭제:2, 모두보기:3, 종료:4 >>4

C:\C++\20231108_Ch9_Hwk\x64\Debug\20231108_Ch9_Hwk.exe(프로세스 25760개)이(가)

종료되었습니다(코드: 0개).

이 창을 닫으려면 아무 키나 누르세요...