

1. 소스 파일

깃허브 업로드 완료

2. 소스 수행 화면

```
Microsoft Visual Studio 디버그 콘솔
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:\Users\seo\Downloads\수업\2-2\1. 월_객체지향프로그래밍\실습\homework5_1\Debug\homework5_1.exe(프로세스 2108개)이(가) 종료되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요...
```

3. 소스 구현 설명

a. 문제 정의

해당 문제는 그래픽 에디터 프로그램을 사용하여 선택한 메뉴에 따라 도형의 삽입, 삭제, 보기, 프로그램 종료의 작업을 진행하는 문제이다.

b. 문제 해결 방법

문제	해결 방법
삽입	UI::insert() 메서드를 사용하여 원하는 도형 선택. 선택한 도형의 클래스를 동적 생성하여 g[index]에 저장.
삭제	UI::del() 메서드를 사용하여 삭제할 도형의 인덱스 선택. 선택한 인덱스 이후의 모든 배열을 한 칸씩 당겨 조정.
보기	배열 g[]에 저장된 도형 객체의 show() 메서드를 호출하여 각 도형의 인덱스와 종류 출력.

c. 아이디어 평가

문제	아이디어 평가
삽입	직관적이고 간단. 또한 제시된 도형이 아닌 새로운 도형이 추가 되더라도 기존 코드 수정 없이 GraphicEditor를 상속받는 새로운 클래스와 선택지를 추가하면 되므로 코드의 확장이 유연함.
삭제	배열의 특징을 활용하여 도형을 직관적으로 관리 및 삭제 후 인덱스를 효율적으로 재조정. 각 도형이 동적으로 생성, 삭제되고 있어 동적 메모리 관리가 유연함.
보기	show() 메서드를 사용하여 단순하고 효과적인 출력 방식. 도형의 수와 관계없이 같은 방식으로 출력할 수 있으며, 도형이 추가되더라도 show() 메서드를 추가하면 되므로 코드의 확장이 유연함.

d. 알고리즘 설명

순서	설명
메인 루프 반복	무한 루프.
메뉴 출력	UI::menu()를 호출하여 사용자가 선택할 수 있는 메뉴 출력.
메뉴 선택	삽입:1 UI::insert()를 호출하여 삽입할 도형 선택. 사용자가 선택한 도형에 따라 해당 도형 객체가 동적으로 생성. 생성된 객체는 g[index] 배열에 저장되며, 도형이 생성될 때마다 index 값이 증가함.
	삭제:2 UI::del()를 호출하여 삭제할 도형 선택. 선택된 인덱스를 기준으로 삭제된 도형 이후의 모든 도형을 g[i]=g[i+1]을 사용하여 배열에서 한 칸씩 앞으로 이동. 각 도형의 index 값은 subIndex() 메서드를 사용하여 재조정.
	모두보기:3 g[i] -> show()를 통해 각 도형 객체의 show() 메서드 호출. 도형의 인덱스와 이름이 출력.
	종료:4 break 문을 사용하여 반복문 종료.