

-문제 정의

5번째 과제에서 만든 Shape 클래스의 포인터를 이용한 객체들의 연결 대신에 vector를 사용해 객체들을 저장하고 활용 하도록 바꾸는 것

-문제 해결 방법

1. GraphicEditor 클래스 insertShape 함수에 기존의 포인터를 이용한 객체들의 연결 대신에 vector 클래스의 연산자 함수인 push_back 함수를 이용해 객체들을 Shape 클래스의 v에 삽입 한다.
2. wb-아이디어 평 GraphicEditor 클래스 deleteShape 함수에 기존의 포인터를 이용한 객체들의 삭제 대신에 vector 클래스의 연산자 함수인 erase와 STL 컨테이너 원소에 대한 포인터인 iterator을 이용해 객체들을 삭제한다.
3. GraphicEditor 클래스 runEditor 함수에 기존의 포인터를 이용한 객체들의 검색에서 vector 타입인 객체 v를 배열처럼 v[2] 이런 식으로 원소 값에 접근한다.
4. GraphicEditor 클래스 runEditor 함수에 반복문과 vector 클래스의 연산자 함수인 size를 통해 v의 원소 개수를 알아낸다.

-아이디어 평가

문제 해결 방법 1번 결과 : vector 클래스의 연산자 함수인 push_back 함수를 이용해 v 맨 마지막에 객체들을 삽입 할 수 있었다.

문제 해결 방법 2번 결과 : vector 클래스의 연산자 함수인 erase와 STL 컨테이너 원소에 대한 포인터인 iterator을 이용해 원소를 삭제 할 수 있었다.

문제 해결 방법 3번 결과 : v는 vector 타입이지만 [] 연산자가 내장되어 있어서 배열처럼 접근 가능했다.

문제 해결 방법 4번 결과 : vector 클래스의 연산자 함수인 size를 통해 v의 원소의 개수를 알아내 for 문에 사용 가능했다.

-알고리즘

vector 클래스와 멤버 함수들을 이용해 가능하게 객체들을 한 곳으로 모으고 삭제하는 등의 기능을 활용하는 것