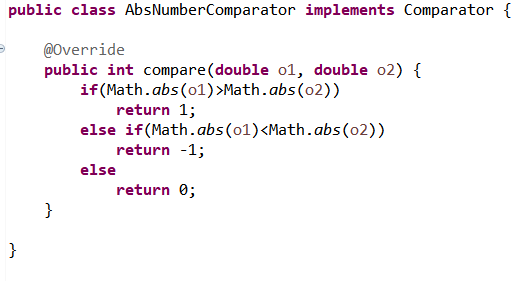
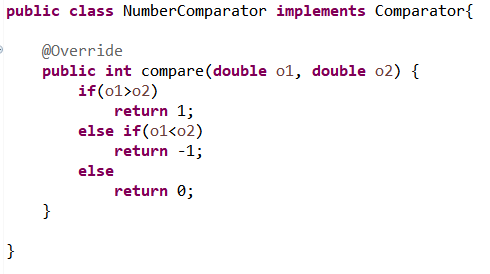
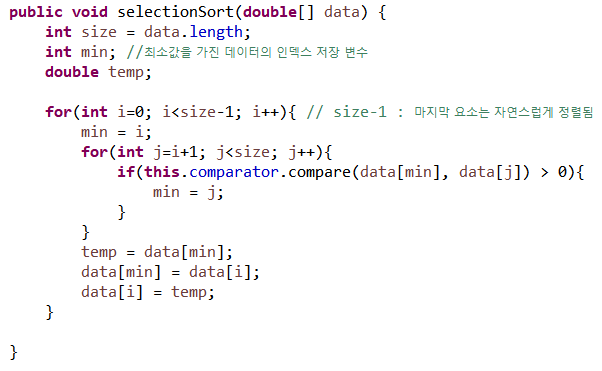
Java20-2 Midterm

32191826 박주희

<Number 1>

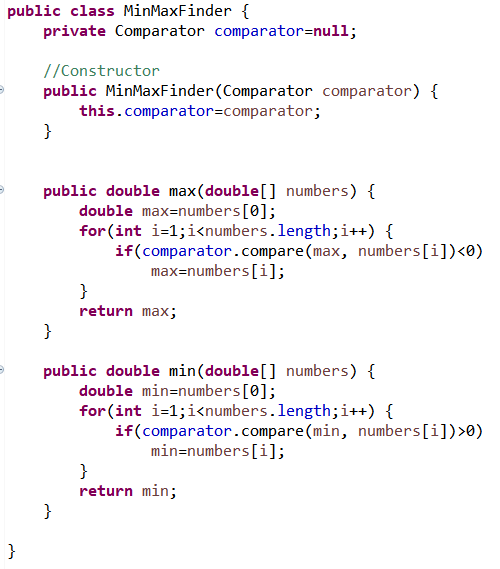
Strategy Pattern을 사용해서 MinMaxFinder 클래스의 double min, double max 함수의 비교 알고리즘을 이용하여 최대 최소 원소를 찾는다. Sorter 클래스에서 selectionSort를 이용하여 숫자로 비교 정렬 또는 절댓값으로 비교 정렬할 수 있는 프로그램이다.





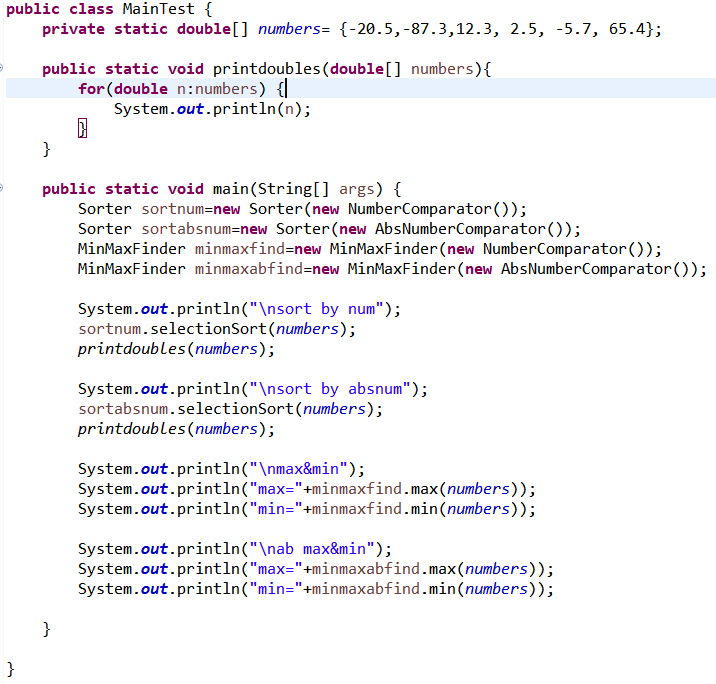
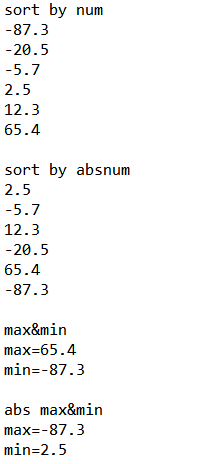
과제로 한 bubblesort과는 다른 sort 방법이다.

SelectionSort는 자리는 정해져 있는 상테에서 첫번째 자리에 가장 작은 녀석을 넣고 그 후에 두번째 자리에 그 다음 가장 작은 것을 선택해 넣는다. 이것을 배열 끝까지 반복한다. 배열의 원소를 비교할 때 Comparator를 이용한다.



시험 시간에는 작성하지 못한 코드이다.

max 메소드는 첫번째 원소를 max로 저장하고 compare을 이용해서 배열 전체를 돌면서 이보다 클 경우 max로 저장하는 것을 반복한다. min 메소드는 이와 반대로 첫번째 원소를 min으로 저장하고 compare한다.



<Number 2>

Decorator Pattern: 주어진 상황 및 용도에 따라 어떤 객체에 책임을 덧붙이는 패턴으로, 객체에 추가적인 요건을 동적으로 첨가하며, 기능 확장이 필요할 때 서브클래싱 대신 쓸 수 있는 유연한 대안이 될 수 있다.

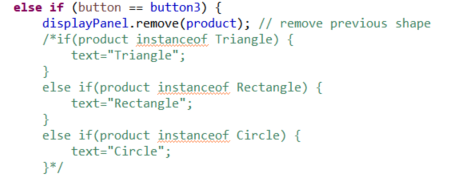
Decorator Pattern을 사용해서 Jcombobox로 Triangle, Rectangle, Circle 중 선택하고 이를 JButton으로 선택하여 red color, thick border, text로 꾸며주는 프로그램이다.

전체적으로 Lab4와 유사했지만 Lab4는 주어진 이미지를 drawImage하는 것이었고 Number2는 java.awt를 이용하여 도형을 그리는 것이다.

Shape->Triangle, Rectangle, Circle

ShapeDecorater -> ColorDecorator, BorderDecorator, TextDecorator

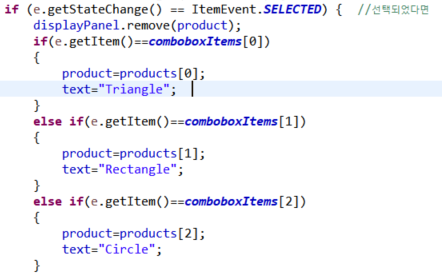
시험시간에도 풀지 못하고 다시 푸는데도 어려움이 있었던 부분이다.



처음에 작성했던 코드이다.

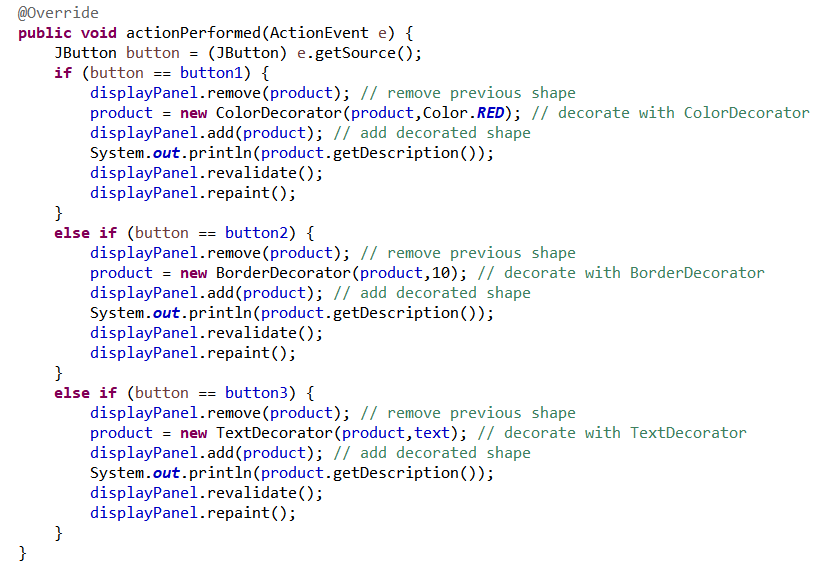
TextDecorator가 추가되면 도형의 이름이 그려져야 한다.

이를 위해서 text 변수를 만들고 product가 Triangle, Rectangle, Circle 중 어느것의 instance인지를 조건문을 이용해서 찾고 그에 따라 text를 저장하였다. 처음에 text=””라고 저장하다보니 계속 text가 출력이 안되서 text paint 부분의 오류인줄 알았다. 그러나 text 초기값을 다르게 저장하니 출려되는 것을 보고 MainFrame 문제임을 알 수 있었다.



이는 수정 후 코드이다.

text를 클래스 변수로 저장한 후 itemStateChanged 메소드에서 combobox 중 무엇을 선택했느냐에 따라 text가 바뀌도록 하였다. 이렇게 수정하니 text가 도형의 이름에 따라서 잘 그려졌다.



MainFrame class의 main method이다. button으로 decorator를 선택하면 기존의 product는 지워지고 선택된 decorator가 추가된 Shape이 그려진다.

Triangle 상태에서 thick border, text, color button를 눌렀을 경우

