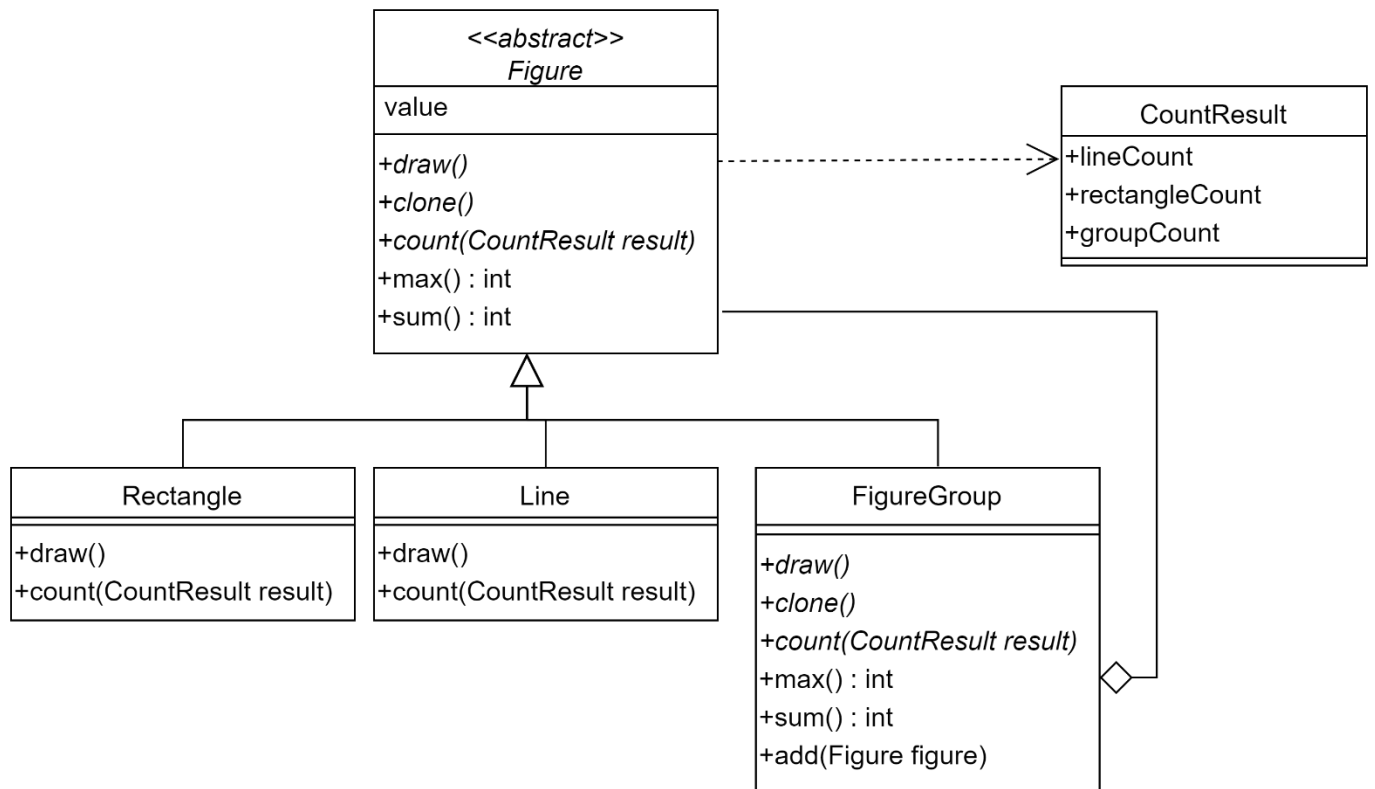


1) 개요



2) CountResult.java

```
1 package visitor.e1;
2
3 public class CountResult {
4     public int lineCount = 0;
5     public int rectangleCount = 0;
6     public int groupCount = 0;
7
8     @Override
9     public String toString() {
10         return String.format("line=%d rectangle=%d group=%d\n",
11                               lineCount, rectangleCount, groupCount);
12     }
13 }
```

3) Figure.java

```
1 package visitor.e1;
2
3 public abstract class Figure implements Cloneable {
4     int value;
5
6     public Figure(int value) {
7         this.value = value;
8     }
9
10    public abstract void draw(int indent);
11
12    @Override
13    public Figure clone() throws CloneNotSupportedException {
14        return (Figure)super.clone();
15    }
16
17    public int max() {
18        return value;
19    }
20
21    public int sum() {
22        return value;
23    }
24
25    public abstract void count(CountResult result);
26 }
```

4) Rectangle.java

```
1 package visitor.e1;
2
3 public class Rectangle extends Figure {
4
5     public Rectangle(int value) {
6         super(value);
7     }
8
9     @Override
10    public void draw(int indent) {
11        String padding = " ".repeat(indent);
12        System.out.printf("%sRectangle(%d)\n", padding, value);
13    }
14
15    @Override
16    public void count(CountResult result) {
17        result.rectangleCount++;
18    }
19 }
```

5) Line.java

```
1 package visitor.e1;
2
3 public class Line extends Figure {
4     String label;
5
6     public Line(int value) {
7         super(value);
8     }
9
10    @Override
11    public void draw(int indent) {
12        String padding = " ".repeat(indent);
13        System.out.printf("%sLine(%d)\n", padding, value);
14    }
15
16    @Override
17    public void count(CountResult count) {
18        count.lineCount++;
19    }
20
21 }
```

6) FigureGroup.java

```
1 package visitor.e1;
2
3 import java.util.ArrayList;
4 import java.util.List;
5
6 public class FigureGroup extends Figure {
7
8     public FigureGroup(int value) {
9         super(value);
10    }
11
12    private List<Figure> figures = new ArrayList<Figure>();
13
14    @Override
15    public Figure clone() throws CloneNotSupportedException {
16        FigureGroup group = new FigureGroup(value);
17        for (Figure figure : figures)
18            group.add(figure.clone());
19        return group;
20    }
21
22    @Override
23    public void draw(int indent) {
24        String padding = " ".repeat(indent);
25        System.out.printf("%sGroup\n", padding);
26        for (Figure figure : figures)
27            figure.draw(indent + 1);
28        System.out.printf("%s\n", padding);
29    }
30
31    public void add(Figure f) {
32        figures.add(f);
33    }
34
35    public int getCount() {
36        return figures.size();
37    }
38
39    public Figure get(int index) {
40        return figures.get(index);
41    }
42
43    public void remove(int index) {
44        figures.remove(index);
45    }
46
47    public void remove(Figure figure) {
48        figures.remove(figure);
49    }
50
51    @Override
52    public int max() {
53        int result = Integer.MIN_VALUE;
54        for (Figure figure : figures)
55            result = Math.max(result, figure.max());
56        return result;
57    }
58
59    @Override
60    public int sum() {
61        int result = 0;
62        for (Figure figure : figures)
63            result += figure.sum();
64        return result;
65    }
66
67    @Override
68    public void count(CountResult result) {
```

```
69     result.groupCount++;  
70     for (Figure figure : figures)  
71         figure.count(result);  
72     }  
73 }
```

7) Example1.java

```
1 package visitor.e1;
2
3 import java.util.Scanner;
4
5 public class Example1 {
6
7     static FigureGroup root = new FigureGroup(0);
8
9     static void drawFigures() {
10         System.out.println();
11         for (int i = 0; i < root.getCount(); ++i) {
12             System.out.printf("%d: ", i);
13             root.get(i).draw(0);
14         }
15         System.out.println();
16     }
17
18     static void execute(String cmd) {
19         try {
20             CountResult countResult;
21             String[] a = cmd.split("[, ]+");
22             switch (a[0].toLowerCase()) {
23                 case "rectangle": root.add(new Rectangle(Integer.valueOf(a[1]))); break;
24                 case "line": root.add(new Line(Integer.valueOf(a[1]))); break;
25                 case "remove":
26                     int index = Integer.valueOf(a[1]);
27                     root.remove(index);
28                     break;
29                 case "duplicate":
30                     Figure figure = root.get(Integer.valueOf(a[1]));
31                     root.add(figure.clone());
32                     break;
33                 case "group":
34                     FigureGroup group = new FigureGroup(0);
35                     for (int i = 1; i < a.length; ++i)
36                         group.add(root.get(Integer.valueOf(a[i])));
37                     root.add(group);
38                     for (int i = 0; i < group.getCount(); ++i)
39                         root.remove(group.get(i));
40                     break;
41                 case "max":
42                     System.out.printf("max = %d\n", root.max());
43                     break;
44                 case "sum":
45                     System.out.printf("sum = %d\n", root.sum());
46                     break;
47                 case "count":
48                     countResult = new CountResult();
49                     root.count(countResult);
50                     System.out.println(countResult);
51                     break;
52                 case "quit": System.exit(0); break;
53                 default:
54                     System.out.println(a[0]);
55             }
56         } catch (Exception e) {
57             e.printStackTrace();
58         }
59     }
60
61     static void prompt() {
62         System.out.printf(" 사각형 : rectangle 정수\n");
63         System.out.printf(" 선 : line 정수\n");
64         System.out.printf(" 삭제 : remove 번호\n");
65         System.out.printf(" 복제 : duplicate 번호\n");
66         System.out.printf(" 그룹 : group 번호1, 번호2,...\n");
67         System.out.printf(" 최대값 : max\n");
68         System.out.printf(" 합계 : sum\n");
```

```

69         System.out.printf(" 도형 수: countWn");
70         System.out.printf(" 종료   : quitWn");
71         System.out.printf(" ? ");
72     }
73
74     public static void main(String[] args) {
75         try (Scanner scanner = new Scanner(System.in)) {
76             while (true) {
77                 prompt();
78                 String cmd = scanner.nextLine();
79                 execute(cmd);
80                 drawFigures();
81             }
82         }
83     }
84 }

```

도형에 대한 작업을 추가할 때 마다, 도형 클래스들을 수정해야 한다.