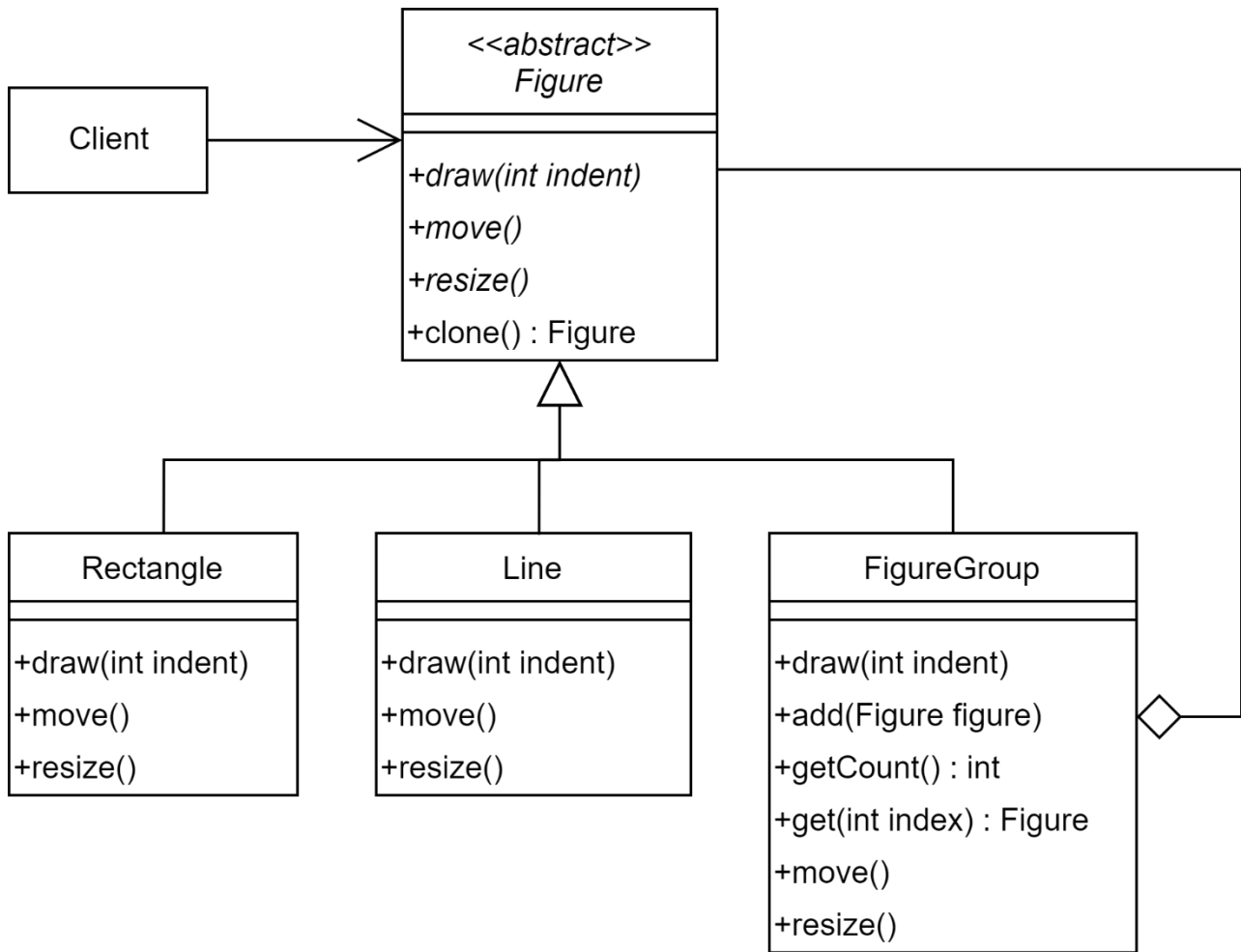


1) 개요



2) Figure.java

```
1 package composite.f2;
2
3 public abstract class Figure implements Cloneable {
4
5     public abstract void draw(int indent);
6     public abstract void move(int dx, int dy);
7
8     @Override
9     public Figure clone() throws CloneNotSupportedException {
10         return (Figure)super.clone();
11     }
12 }
```

int indent

도형을 들여쓰기 위한 값

3) Rectangle.java

```
1 package composite.f2;
2
3 public class Rectangle extends Figure {
4     int x, y;
5     String label;
6
7     public Rectangle(String label) {
8         this.label = label;
9     }
10
11     @Override
12     public void draw(int indent) {
13         String padding = " ".repeat(indent);
14         System.out.printf("%sRectangle(%s %d %d)\n", padding, label, x, y);
15     }
16
17     @Override
18     public void move(int dx, int dy) {
19         x += dx; y += dy;
20     }
21 }
```

4) Line.java

```
1 package composite.f2;
2
3 public class Line extends Figure {
4     int x, y;
5     String label;
6
7     public Line(String label) {
8         this.label = label;
9     }
10
11     @Override
12     public void draw(int indent) {
13         String padding = " ".repeat(indent);
14         System.out.printf("%sLine(%s, %d, %d)\n", padding, label, x, y);
15     }
16
17     @Override
18     public void move(int dx, int dy) {
19         x += dx; y += dy;
20     }
21 }
```

5) FigureGroup.java

```
1 package composite.f2;
2
3 import java.util.ArrayList;
4 import java.util.List;
5
6 public class FigureGroup extends Figure {
7     private List<Figure> figures = new ArrayList<Figure>();
8
9     @Override
10    public Figure clone() throws CloneNotSupportedException {
11        FigureGroup group = new FigureGroup();
12        for (Figure figure : figures)
13            group.add(figure.clone());
14        return group;
15    }
16
17    @Override
18    public void draw(int indent) {
19        String padding = " ".repeat(indent);
20        System.out.printf("%sGroup\n", padding);
21        for (Figure figure : figures)
22            figure.draw(indent + 1); // 그룹 내부 도형 1 칸 들여쓰기
23        System.out.printf("%s\n", padding);
24    }
25
26    public void add(Figure f) {
27        figures.add(f);
28    }
29
30    public int getCount() {
31        return figures.size();
32    }
33
34    public Figure get(int index) {
35        return figures.get(index);
36    }
37
38    @Override
39    public void move(int dx, int dy) {
40        for (Figure figure : figures)
41            figure.move(dx, dy);
42    }
43 }
```

clone 메소드 구현: deep copy

6) Example2.java

```
1 package composite.f2;
2
3 import java.util.ArrayList;
4 import java.util.List;
5 import java.util.Scanner;
6
7 public class Example2 {
8
9     static List<Figure> figures = new ArrayList<Figure>();
10
11     static void drawFigures() {
12         System.out.println();
13         for (int i = 0; i < figures.size(); ++i) {
14             System.out.printf("%d: ", i);
15             figures.get(i).draw();
16         }
17         System.out.println();
18     }
19
20     static void execute(String cmd) {
21         try {
22             String[] a = cmd.split("[, ]+");
23             switch (a[0].toLowerCase()) {
24                 case "rectangle": figures.add(new Rectangle(a[1])); break;
25                 case "line": figures.add(new Line(a[1])); break;
26                 case "remove":
27                     int index = Integer.valueOf(a[1]);
28                     figures.remove(index);
29                     break;
30                 case "move":
31                     index = Integer.valueOf(a[1]);
32                     figures.get(index).move(2, 3);
33                     break;
34                 case "duplicate":
35                     Figure figure = figures.get(Integer.valueOf(a[1]));
36                     figures.add(figure.clone());
37                     break;
38                 case "group":
39                     FigureGroup group = new FigureGroup();
40                     for (int i = 1; i < a.length; ++i)
41                         group.add(figures.get(Integer.valueOf(a[i])));
42                     figures.add(group);
43                     for (int i = 0; i < group.getCount(); ++i)
44                         figures.remove(group.get(i));
45                     break;
46                 case "quit": System.exit(0); break;
47             }
48         } catch (Exception e) {
49         }
50     }
51
52     static void prompt() {
53         System.out.printf(" 사각형 : rectangle 레이블\n");
54         System.out.printf(" 선 : line 레이블\n");
55         System.out.printf(" 삭제 : remove 번호\n");
56         System.out.printf(" 복제 : duplicate 번호\n");
57         System.out.printf(" 이동 : move 번호\n");
58         System.out.printf(" 그룹 : group 번호1, 번호2,...\n");
59         System.out.printf(" 종료 : quit\n");
60         System.out.printf(" ? ");
61     }
62
63     public static void main(String[] args) {
64         try (Scanner scanner = new Scanner(System.in)) {
65             while (true) {
66                 prompt();
67                 String cmd = scanner.nextLine();
68                 execute(cmd);
69             }
69         }
70     }
71 }
```

69	drawFigures();
70	}
71	}
72	}
73	}

실행 사례

사각형 : rectangle 레이블
 선 : line 레이블
 삭제 : remove 번호
 복제 : duplicate 번호
 이동 : move 번호
 그룹 : group 번호1, 번호2,...
 종료 : quit
 ? **rectangle A**

0: Rectangle(A 0 0)

사각형 : rectangle 레이블
 선 : line 레이블
 삭제 : remove 번호
 복제 : duplicate 번호
 이동 : move 번호
 그룹 : group 번호1, 번호2,...
 종료 : quit
 ? **move 0**

0: Rectangle(A 2 3)

사각형 : rectangle 레이블
 선 : line 레이블
 삭제 : remove 번호
 복제 : duplicate 번호
 이동 : move 번호
 그룹 : group 번호1, 번호2,...
 종료 : quit
 ? **line B**

0: Rectangle(A 2 3)

1: Line(B, 0, 0)

사각형 : rectangle 레이블
 선 : line 레이블
 삭제 : remove 번호
 복제 : duplicate 번호
 이동 : move 번호
 그룹 : group 번호1, 번호2,...
 종료 : quit
 ? **move 1**

0: Rectangle(A 2 3)

1: Line(B, 2, 3)

사각형 : rectangle 레이블
 선 : line 레이블
 삭제 : remove 번호
 복제 : duplicate 번호
 이동 : move 번호
 그룹 : group 번호1, 번호2,...
 종료 : quit
 ? **move 1**

0: Rectangle(A 2 3)

1: Line(B, 4, 6)

사각형 : rectangle 레이블
 선 : line 레이블
 삭제 : remove 번호

복제 : duplicate 번호
이동 : move 번호
그룹 : group 번호1, 번호2,...
종료 : quit
? duplicate 0

```
0: Rectangle(A 2 3)
1: Line(B, 4, 6)
2: Rectangle(A 2 3)
```

사각형 : rectangle 레이블
선 : line 레이블
삭제 : remove 번호
복제 : duplicate 번호
이동 : move 번호
그룹 : group 번호1, 번호2,...
종료 : quit
? group 0 1

```
0: Rectangle(A 2 3)
1: Group(
  Rectangle(A 2 3)
  Line(B, 4, 6)
)
```

사각형 : rectangle 레이블
선 : line 레이블
삭제 : remove 번호
복제 : duplicate 번호
이동 : move 번호
그룹 : group 번호1, 번호2,...
종료 : quit
? duplicate 1

```
0: Rectangle(A 2 3)
1: Group(
  Rectangle(A 2 3)
  Line(B, 4, 6)
)
2: Group(
  Rectangle(A 2 3)
  Line(B, 4, 6)
)
```

사각형 : rectangle 레이블
선 : line 레이블
삭제 : remove 번호
복제 : duplicate 번호
이동 : move 번호
그룹 : group 번호1, 번호2,...
종료 : quit
? group 0 1

```
0: Group(
  Rectangle(A 2 3)
  Line(B, 4, 6)
)
1: Group(
  Rectangle(A 2 3)
  Group(
    Rectangle(A 2 3)
    Line(B, 4, 6)
  )
)
```

사각형 : rectangle 레이블
선 : line 레이블
삭제 : remove 번호
복제 : duplicate 번호

이동 : move 번호
그룹 : group 번호1, 번호2,...
종료 : quit
? duplicate 1

```
0: Group(  
  Rectangle(A 2 3)  
  Line(B, 4, 6)  
)  
1: Group(  
  Rectangle(A 2 3)  
  Group(  
    Rectangle(A 2 3)  
    Line(B, 4, 6)  
  )  
)  
2: Group(  
  Rectangle(A 2 3)  
  Group(  
    Rectangle(A 2 3)  
    Line(B, 4, 6)  
  )  
)  
)
```

사각형 : rectangle 레이블
선 : line 레이블
삭제 : remove 번호
복제 : duplicate 번호
이동 : move 번호
그룹 : group 번호1, 번호2,...
종료 : quit
? move 2

```
0: Group(  
  Rectangle(A 2 3)  
  Line(B, 4, 6)  
)  
1: Group(  
  Rectangle(A 2 3)  
  Group(  
    Rectangle(A 2 3)  
    Line(B, 4, 6)  
  )  
)  
2: Group(  
  Rectangle(A 4 6)  
  Group(  
    Rectangle(A 4 6)  
    Line(B, 6, 9)  
  )  
)  
)
```

사각형 : rectangle 레이블
선 : line 레이블
삭제 : remove 번호
복제 : duplicate 번호
이동 : move 번호
그룹 : group 번호1, 번호2,...
종료 : quit
? quit

노란색으로 칠한 부분은, 사용자가 입력한 명령
하늘색으로 칠한 부분은, 사용자의 명령에 의해서 생성된 도형 객체