

Worksheet: Composite Pattern 1

Let us assume, that for our graphics editor the two figures `RectangleFigure` and `OvalFigure` have been implemented, and that class `GroupFigure` has been implemented to provide group figures. Method `addFigure` allows to add new figures to an existing group.

```
public class GroupFigure implements Figure {
    private String name;
    public GroupFigure(String name, Figure... figures) {
        this.name = name;
        for(Figure f : figures) { addFigure(f); }
    }
    public void addFigure(Figure f) { ... }
    ....
}
```

Consider the following statement sequences that work with group figures:

```
(1) Figure f1 = new RectangleFigure("f1");
    Figure f2 = new RectangleFigure("f2");
    Figure f3 = new OvalFigure("f3");

    GroupFigure g1 = new GroupFigure("g1", f1, f2);
    GroupFigure g2 = new GroupFigure("g2", f3, f1);

(2) Figure f1 = new RectangleFigure("f1");
    Figure f2 = new RectangleFigure("f2");
    Figure f3 = new OvalFigure("f3");

    GroupFigure g1 = new GroupFigure("g1", f1, f2);
    GroupFigure g2 = new GroupFigure("g2", g1, f3);
    g1.addFigure(g2);
```

Tasks:

- Discuss why the statement sequences (1) and (2) are problematic (to draw the structures of the defined group figures might be helpful).
- Project 06_Composite available in the active directory contains the statement sequences (1) and (2) as shown above. Change the code such that problematic situations are recognized and signaled with runtime exceptions (e.g. with an `IllegalArgumentException`). In order to reach your goal you can declare additional fields (and methods) in all classes and you may (if that is necessary) change the signatures of the existing methods and constructors.