

```
1 interface Behavior {
2     void onEnter();
3
4     void observe();
5 }
6
7 class PeacefoulBehavior implements Behavior {
8     private final Mammoth mammoth;
9
10    public PeacefoulBehavior(Mammoth mammoth) {
11        this.mammoth = mammoth;
12    }
13
14    public void observe() {
15        System.out.println(mammoth + " is calm and peaceful.");
16    }
17
18    public void onEnter() {
19        System.out.println(mammoth + " calms down.");
20    }
21 }
22
23 class AngryBehavior implements Behavior {
24     private final Mammoth mammoth;
25
26    public AngryBehavior(Mammoth mammoth) {
27        this.mammoth = mammoth;
28    }
29
30    public void observe() {
31        System.out.println(mammoth + " is furious!");
32    }
33
34    public void onEnter() {
35        System.out.println(mammoth + " get angry!");
36    }
37 }
38
39 class Mammoth {
40     private Behavior behavior;
41
42    public Mammoth() {
43        behavior = new PeacefoulBehavior(this);
44    }
45
46    public void timePasses() {
47        if (behavior.getClass().equals(PeacefoulBehavior.class)) {
48            changeBehaviorTo(new AngryBehavior(this));
49        } else {
50            changeBehaviorTo(new PeacefoulBehavior(this));
51        }
52    }
53
54    public void changeBehaviorTo(Behavior newBehavior) {
55        this.behavior = newBehavior;
56        this.behavior.onEnter();
57    }
58
59    public String toString() {
60        return "The mammoth";
61    }
62
63    public void observe() {
64        this.behavior.observe();
65    }
66 }
67
68
69 class Client {
70     public static void main(String[] args) {
71         Mammoth m = new Mammoth();
72         m.observe();
73         m.timePasses();
74         m.observe();
75         m.timePasses();
76         m.observe();
77     }
78 }
79
```

Pattern <b>STRUTTURALI</b>	Pattern <b>CREAZIONALI</b>	Pattern <b>COMPORTAMENTALI</b>
<input type="checkbox"/> Adapter <input type="checkbox"/> Decorator <input type="checkbox"/> Composite	<input type="checkbox"/> Abstract Factory <input type="checkbox"/> Singleton	<input type="checkbox"/> Observer <input type="checkbox"/> State <input type="checkbox"/> Visitor <input type="checkbox"/> Strategy