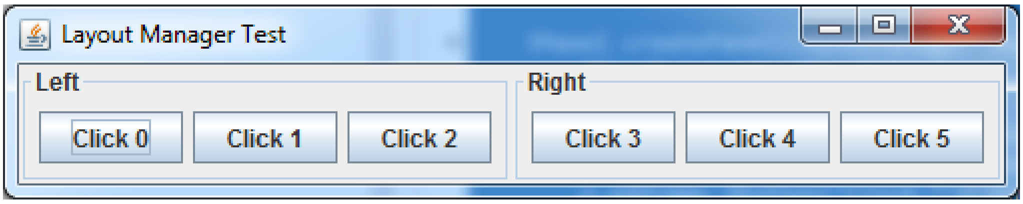
Module Design Patterns
Prof. Dr. Dominik Gruntz
Prof. Dr. Wolfgang Weck

Worksheet: Strategy Pattern

In class you have seen that a strategy object can either be used by one context or by several contexts. With this worksheet you learn under which conditions a strategy object can either be used by several contexts or by one context only.

We will investigate different strategy implementations with the help of the program `LayoutComparer` which you find in this week's project `05_Strategy.zip` stored on the Active Directory (and on the back side).

This program opens a window which contains two panels "Left" and "Right" side by side. In each of these panels three buttons are added. These buttons are placed with a `FlowLayout` manager which is ignoring the constraints "West", "Center" and "East".



Now change the code so that a `BorderLayout` manager is used instead of the `FlowLayout` manager. You only have to remove the comment characters "`//`" in the third to last line.

Explain why the buttons are no longer visible in the left window.

Hints:

- How many layout manager instances are used in the whole application?
- The situation can (supposedly) be corrected if the layouting algorithm is explicitly triggered by invoking method `pack()` between the two invocations of `add(createPanel(…))`. The problem described above seems to be fixed, but if you now change the size of the window, then only one of the two panels is adjusted. Why?
- In order to answer these questions it might be helpful if you study the sources of class `BorderLayout`.
- How would the implementation of the `BorderLayout` manager class have to be changed so that an instance can be used by several contexts?

```
package patterns.strategy.stateful;

import java.awt.BorderLayout;
import java.awt.FlowLayout;
import java.awt.GridLayout;
import java.awt.LayoutManager;

import javax.swing.BorderFactory;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JPanel;

public class LayoutComparer extends JFrame {

   public static void main (String[] args) {
      JFrame f = new LayoutComparer();
      f.setDefaultCloseOperation(EXIT_ON_CLOSE);
      f.pack();
      f.setVisible(true);
   }

   static int counter = 0;

   JPanel createPanel(LayoutManager layout, String title) {
      JPanel p = new JPanel();
      p.setLayout(layout);
      p.add(new JButton("Click "+counter++), "West");
      p.add(new JButton("Click "+counter++), "Center");
      p.add(new JButton("Click "+counter++), "East");
      p.setBorder(BorderFactory.createTitledBorder(title));
      return p;
   }

   LayoutComparer() {
      setTitle("Layout Manager Test");
      setLayout(new GridLayout(1, 2));
      LayoutManager m;
      m = new FlowLayout();
//    m = new BorderLayout();

      add(createPanel(m, "Left"));
      add(createPanel(m, "Right"));
   }
}
```