

# Übungsblatt 3

(10. April 2008)

**Aufgabe 1** Legen Sie die Klassen aus Ihrer Lösung des letzten Übungsblattes in ein Paket, das Sie als Entwickler eindeutig identifiziert.

**Aufgabe 2** Schreiben Sie für Ihre Klassen aus dem zweiten Übungsblatt eine Methode `public boolean equals(Object that)`.

**Aufgabe 3** Lassen Sie jetzt zusätzlich die Klasse `GeometricObject` folgende Schnittstelle implementieren:

```
Paintable.java
1 package name.panitz.javaMI;
2 import java.awt.Graphics;
3 public interface Paintable{
4     public void paintMeTo(Graphics g);
5 }
```

Implementieren Sie in allen Unterklassen von `GeometricObject` die Methode `paintMe`.

**Aufgabe 4** Erzeugen Sie in einer Testmethode Objekte der folgenden Klasse:

```
ShowPaintable.java
1 package name.panitz.javaMI;
2 import java.awt.Graphics;
3 import java.awt.Dimension;
4 import javax.swing.*;
5
6 public class ShowPaintable extends JPanel{
7     Paintable p;
8
9     @Override public Dimension getPreferredSize(){
10         return new Dimension(400,300);
11     }
12     @Override public void paintComponent(Graphics g){
13         super.paintComponent(g);
14         p.paintMeTo(g);
15     }
}
```

```
16 public ShowPaintable(Paintable p){
17     this.p=p;
18     JFrame f = new JFrame();
19     f.add(this);
20     f.pack();
21     f.setVisible(true);
22 }
23 }
```

Übergeben Sie dem Konstruktor dabei jeweils unterschiedliche Objekte der Klassen, die die Schnittstelle `Paintable` implementieren. Also z.B.: `new ShowPaintable(new Circle(40,new Vertex(100,40)))`;