

Polymorphismus

Polymorphismus heißt übersetzt „Vielgestaltigkeit“. Man kann dieses Prinzip am besten an einem Beispiel erklären:

```
public class Haus{
    protected String Anschrift;
    public void oeffneTuer(){
        System.out.println("Öffne Haustür")
    }
}

public class Kaufhaus extends Haus{
    protected String Oeffnungszeiten;
    public void oeffneTuer(){
        System.out.println("Öffne Kaufhaustür");
    }
}

public class HaeuserDemo {

    public static void main(String[] args) {
        Haus EinHaus=new Haus();
        EinHaus.oeffneTuer(); // (1)

        Kaufhaus EinKaufhaus=new Kaufhaus();
        EinKaufhaus.oeffneTuer(); // (2)

        Haus WeiteresHaus=new Kaufhaus();
        WeiteresHaus.oeffneTuer(); // (3)
    }
}
```

In der Klasse Kaufhaus wurde die Methode `oeffneTuer()` überschrieben. In der Klasse HaeuserDemo werden die vorher deklarierten Klassen verwendet.

Bei (1) Öffne Haustür wird ausgegeben. Bei (2) wird Öffne Kaufhaustür angezeigt.

Aufgabe: Was wird bei (3) ausgegeben?

Die Ausgabe bei (3) kann man durch Polymorphismus erklären. Dieses Prinzip wird nur bei überschriebenen Methoden verwendet. Man definiert eine Methode in einer Basisklasse und überschreibt diese Methode in abgeleiteten Klassen. Jedoch kann ein Objekt dieser Klasse etwas ganz anderes in dieser Methode ausführen als die Basisklasse, auch wenn dieses Objekt in einer Variablen vom Typ der Basisklasse steckt.

Polymorphismus gibt es z.B. auch bei Tieren. Alle Tiere fressen. Aber ein Zebra frisst ganz anders als ein Löwe.

Applets

Applets sind ein Grund, warum Java so populär geworden sind. Applets sind „kleine“ Programme, die im Webbrowser laufen und z.B. für Animationen genutzt werden. Damit Applets im Webbrowser funktionieren, muss aber eine Java Virtual Machine installiert sein. Bei vie-

len Browser ist nur eine recht alte JVM vorhanden, deshalb sollte man sich von der Webseite von Sun (www.java.sun.com) die aktuelle Version der Java Runtime herunterladen und installieren. Dies ist natürlich nicht nötig, wenn man das Java SDK benutzt, denn dadurch ist automatisch die Runtime aktiviert.

Jedes Applet wird in eine HTML-Seite eingebunden. Eine minimale Webseite sieht so aus:

```
<html>
  <head>
    <title>Applet-Minimalseite</title>
  </head>
  <body>
    <applet code="ErstesApplet" width = "400" height = "300">
    </applet>
  </body>
</html>
```

Interessant ist das Applet-Tag. `code` gibt an, wo sich das Applet befindet und wie es heißt. In diesem Fall heißt es `ErstesApplet` und es befindet sich im gleichen Ordner wie die HTML-Datei. `width` und `height` geben die Breite und Höhe des Applets an. In HTML 4 gibt es noch eine weitere Möglichkeit Applets einzubinden, die wir im Moment aber nicht brauchen.

Der Quelltext des Applets sieht so aus:

```
import java.applet.Applet;
import java.awt.*;
```

```
public class ErstesApplet extends Applet {

    public void paint(Graphics g){
        g.drawString("Hallo vom ersten Applet!", 50, 50);
        g.drawRect(100, 100, 150, 150);
    }
}
```

Die Klasse `ErstesApplet` ist abgeleitet von `Applet` und enthält eine Methode namens `paint` mit einem Parameter vom Typ `Graphics`. Die Methode `paint` ist für das Zeichnen des Applets verantwortlich. In diesem Beispiel wird mit der Methode `drawString` vom Objekt der Klasse `Graphics` ein Text ab der Position (50,50) sowie ein Rechteck mittels `drawRect` ausgegeben. Objekte der Klasse `Graphics` enthalten viele weitere Methoden, um verschiedene Objekte zu zeichnen. Die y-Achse des Koordinatensystem in einem Java-Applet verläuft von oben nach unten und nicht wie in der Mathematik von unten nach oben.

Damit das Applet funktioniert, dürfen die beiden `import`-Anweisungen nicht fehlen!

Nachdem das Applet kompiliert ist, kann man die Webseite in einem Browser anzeigen lassen. Zum Testen von Applets hat sich aber der Appletviewer bewährt, das sich im Java SDK befindet, denn die Webbrowser speichern alte Versionen zwischen und können nicht richtig aktualisiert werden.

In Java gibt es zwei verschiedene Arten von Applets. Klassen, die direkt von der Klasse `Applet` abgeleitet sind und Klassen, die von `JApplet` erben. Für letztere braucht man eine aktuelle Java-Runtime, aber sie bieten dafür auch viel mehr Möglichkeiten.

Aufgaben:

1. Schreibe ein Java-Applet namens Bunt, das verschiedene bunte geometrische Objekte, wie z.B. Kreise, Rechtecke, Linien zeichnet.
2. Ergänze die Klassen vom letzten Blatt jeweils um eine Methode `paint` mit einem Parameter vom Typ `Graphics`, die die geometrische Formen zeichnet.