



Programmieren: Einstieg in die Informatik mit Java

WS 2006/2007

Dr. G. Bohlender
Dipl.–Math. techn. M. Richter

08.01.2007

Aufgabenblatt 9

Bearbeitungszeitraum: 10.01.2007 – 24.01.2007

Aufgabe 22: Objektorientiertes Programmieren

Die grundlegende Idee der objektorientierten Programmierung (OOP) besteht darin, Daten und Algorithmen zu sogenannten Objekten zusammen zu fassen. Ziel dieses Konzeptes ist es, dass auf gegebenen Daten nur „sinnvolle“ Operationen ausgeführt werden können. Dem Programmierer soll so der Umgang mit komplexen Datenstrukturen erleichtert werden. Welche Daten ein Objekt speichern, und welche Operationen es auf diesen Daten ausführen kann, wird durch seine zugehörige (Objekt-)Klasse festgelegt.

In dieser Aufgabe sollen die grundlegenden Schritte des objektorientierten Programmierens eingeübt werden. Ziel dieser Aufgabe ist es, eine Objektklasse namens `Person` zu erstellen, die den Vornamen und den Nachnamen einer Person abspeichert und bei Bedarf den vollen Namen oder aber die Initialen dieser Person zurück gibt.

Gehen Sie dabei wie folgt vor:

- (a) Beginnen Sie damit, die Objektklasse zu erzeugen. Erstellen Sie dazu eine Datei mit dem Namen `Person.java`, in der sie folgenden Quelltext speichern

```
public class Person { }
```

- (b) Legen Sie nun fest, welche Daten ein Objekt der Klasse `Person` speichern soll. Definieren Sie dazu zwei sogenannte Instanzvariablen: eine `String`-Variable namens `vorname` sowie eine zweite `String`-Variable namens `nachname`. Ihre Klassendefinition sollte also wie folgt aussehen:

```
public class Person {  
    String vorname;  
    String nachname;  
}
```

- (c) Nun einen Konstruktor für die Objektklasse. Der Konstruktor ist eine spezielle Methode, mit der eine Instanz der Klasse `Person` (d.h. ein Objekt der Klasse `Person`) erzeugt wird.

Folgende Regeln sind zu beachten: Der Konstruktor trägt denselben Namen, wie die Klasse. Konstruktoren besitzen keinen Rückgabety. Ansonsten können Konstruktoren wie gewöhnliche Methoden behandelt werden. Der Konstruktor der Klasse `Person` soll dazu dienen, die Instanzvariablen mit Werten zu belegen. Versehen Sie den Konstruktor daher mit einer formalen Argumentenliste, die zwei `String`-Argumente umfasst. Ihre Klassendefinition könnte also wie folgt aussehen:

```
public class Person {  
    String vorname;  
    String nachname;  
  
    public Person(String vorname, String nachname) {  
        this.vorname = vorname;  
        this.nachname = nachname;  
    }  
}
```

Nun können Sie in Ihrem späteren Hauptprogramm ein Objekt der Klasse `Person` mit dem Befehl `Person p = new Person("Max", "Mustermann");` erstellen und in der Variable `p` abspeichern. Beachten Sie dass innerhalb des Konstruktors der Ausdruck `vorname` das formale Argument und `this.vorname` die Instanzvariable gleichen Namens bezeichnet.

- (d) Nun sollen Objekte der Klasse `Person` mit gewissen Funktionalitäten ausgestattet werden. Definieren Sie dazu zunächst eine sogenannte Instanzmethode namens `vollerName`, die den vollen Namen der Person zurückgibt. Instanzmethoden können wie gewöhnliche Methoden definiert werden. Allerdings darf das Schlüsselwort `static` nicht verwendet werden. Ihre Klassendefinition sollte also wie folgt aussehen:

```
public class Person {  
    String vorname;  
    String nachname;  
  
    public Person(String vorname, String nachname) {  
        this.vorname = vorname;  
        this.nachname = nachname;  
    }  
  
    public String vollerName() {  
        return vorname + " " + nachname;  
    }  
}
```

In Ihrem späteren Hauptprogramm liefert der Befehl `p.vollerName()`; nun die Zeichenkette `Max Mustermann` zurück. Beachten Sie, dass innerhalb der Methode der Ausdruck `vorname` die Instanzvariable bezeichnet.

- (e) Erweitern Sie ihre Objektklasse um eine weitere Instanzmethode namens `namensInitialen`, welche die ersten Buchstaben des Vor- und des Nachnamens zurückgibt. Erstellen Sie

dann **in einer neuen Datei** ein Java-Programm mit dem Namen `Personen`. Erzeugen Sie in der `main`-Methode zwei Objekte der Klasse `Person` (z.B. mit den Namen „Max Mustermann“ und „Natalie Niemand“) und geben Sie den vollen Namen, sowie die Initialen der Personen aus, die durch diese Objekte repräsentiert werden.

Sie sollten folgendes Ergebnis erhalten

Jahr	Summe der Auszahlungen
1	0.00
2	-100.00
3	0.00
4	-200.00
5	109.27
6	-150.00
7	0.00
8	220.76
9	157.78
10	0.00

Aufgabe 23 (Pflichtaufgabe): *Finanzplanung*

Im Finanzwesen versteht man unter einer Geldanlage eine Investition von Geldbeträgen, die nach einer gewissen Zeitspanne einen bestimmten Ertrag erzielen soll. Werden mehrere Anlagen getätigt, so spricht man von einem Anlagenportfolio. Ziel dieser Aufgabe ist es, ein objektorientiertes Java-Programm zu erstellen, mit dem die Auswirkung eines Anlagenportfolios über einen Zeitraum von 10 Jahren simuliert werden kann. Dabei wird von stark vereinfachten Annahmen ausgegangen.

- (a) Erstellen Sie eine Objektklasse mit dem Namen `Geldanlage`, die eine Geldanlage abstrakt beschreiben soll. Wir gehen davon aus, dass eine Anlage über einen bestimmten Geldbetrag B in einem sogenannten Anfangsjahr t_0 getätigt wird und danach bis zu einem sogenannten Endjahr t_{Ende} läuft. Der Ertrag E der Geldanlage wird durch den sogenannten Zinssatz z bestimmt, d.h. es gilt

$$E = (1 + z)^{t_{\text{Ende}} - t_0} B.$$

Definieren Sie in der Klasse `Geldanlage` für das Anfangsjahr t_0 und für das Endjahr t_{Ende} jeweils eine Instanzvariable vom Typ `int`. Definieren Sie weiterhin für den Betrag B und für den Zinssatz z jeweils eine Instanzvariable vom Typ `double`. Erstellen Sie einen Konstruktor, dem Werte für t_0 , t_{Ende} , B und z übergeben werden können.

- (b) Die Auswirkung einer Geldanlage kann durch eine sogenannte Auszahlungsfunktion a beschrieben werden, die für jedes Jahr t die Kosten bzw. Auszahlung $a(t)$ der Geldanlage zurückliefert. Im Anfangsjahr gilt $a(t_0) = -B$, im Endjahr gilt $a(t_{\text{Ende}}) = E$. Für alle übrigen Jahre, d.h. für $t \neq t_0$ und $t \neq t_{\text{Ende}}$, gilt $a(t) = 0$. Erweitern Sie Ihre Objektklasse `Geldanlage` um eine Instanzmethode namens `auszahlung`, die zu einem gegebenen Jahr t (repräsentiert durch einen `int`-Wert) den Wert der Auszahlungsfunktion $a(t)$ zurückgibt.
- (c) Erstellen Sie ein Java-Programm mit dem Namen `Finanzplanung`. Erstellen Sie in der `main`-Methode ein Feld von `Geldanlage`-Objekten namens `portfolio` mit drei Komponenten. Erzeugen Sie drei Objekte vom Typ `Geldanlage` mit folgenden Parametern:

Anfangsjahr	Endjahr	Betrag	Zinssatz
2	5	100	3.0 %
4	8	200	2.5 %
6	9	150	1.7 %

Speichern Sie diese Objekte in den Feldkomponenten von `portfolio` ab. Berechnen Sie anschließend für die Jahre 1, 2, 3, ..., 10 die jeweilige Summe der Auszahlungsfunktionen der einzelnen Geldanlagen und geben Sie diese auf dem Bildschirm aus.