

Themen:

Stoff der Vorlesung bis einschließlich Verzweigung

#### Aufgabe 4.1

Was ist der Unterschied zwischen folgenden Codesegmenten?

Geben Sie in der Tabelle den Wert von  $x$  nach Ausführung des Codestücks für die Anfangsbelegungen  $x = 3$ ,  $x = 5$  und  $x = 9$  an.

- a) `if (x > 4) if (x < 8) x = 0; else x = 1;`
- b) `if (x > 4) {if (x < 8) x = 0;} else x = 1;`
- c) `if (x > 4) if (x < 8) {x = 0; x = 1;}`
- d) `if (x > 4) {x = 0; if (x < 8) x = 1;}`

Anfangsbelegung von $x$	3	5	9
Wert von $x$ nach Ausführung von a)			
... b)			

#### Aufgabe 4.2

Ein Spieler des Spiels ToF (True or False) hat einen Namen und kann ein Spiel spielen, das er entweder gewinnt oder verliert.

Definieren Sie eine Klasse `Spieler`, die die Spielbilanz eines Spielers führt.

Diese Klasse soll über eine (Instanz-)Methode `spiel(boolean gewonnen)` verfügen. Der Eingabeparameter gibt an, ob der Spieler gewonnen oder verloren hat. Es sollen die Anzahl der insgesamt gespielten Spiele, der insgesamt gewonnenen und der insgesamt verlorenen Spiele und die Gewinnquote (in Prozent) des Spielers ermittelt werden können.

Falls Sie die Klasse mit dieser (↓) Anweisungsfolge testen,

sollte die Ausgabe so aussehen:

```
Spieler a = new Spieler("Anna");
Spieler b = new Spieler("Otto");
b.spiel(true);
a.spiel(true);
a.spiel(true);
b.spiel(false);
a.spiel(true);
a.spiel(true);
b.spiel(true);
a.spiel(false);
b.spiel(false);
a.spiel(false);
a.spiel(true);
a.spiel(true);
a.zeigeBilanz();
System.out.println();
b.zeigeBilanz();
```

```
Spielbilanz fuer Anna:
gewonnen: 6
verloren: 2
Quote: 75.0%
```

```
Spielbilanz fuer Otto:
gewonnen: 2
verloren: 2
Quote: 50.0%
```

### Aufgabe 4.3

Wir entwickeln ein Raumbuchungssystem (denken Sie zB an unsere Stillarbeitsräume): Ein Raum

- hat eine Raumnummer (automatisch durch den Konstruktor vergeben)
- eine Anzahl an Plätzen (wird dem Konstruktor übergeben)
- kann „gebucht“ sein oder nicht.
- Wenn der Raum gebucht werden soll, muss eine Person als verantwortlich eingetragen werden. Außerdem muss angegeben werden, mit wievielen Personen der Raum belegt werden soll. Wenn diese Anzahl größer ist als die Anzahl der Plätze im Raum, wird der Raum nicht vergeben. In diesem Fall soll eine Meldung „Raum bietet nicht genug Platz für ... Personen“ auf dem Monitor ausgegeben werden.
- Falls der Raum vergeben wird, wird nachgehalten, an welche Person der Raum vergeben wurde.
- Der Raum kann wieder „freigeben“ werden und könnte dann an eine andere Person vergeben werden.
- Wenn ein Raum aktuell vergeben ist, kann er nicht von einer anderen Person gebucht werden. Bei dem Versuch, einen schon vergebenen Raum erneut zu vergeben, soll die Meldung „Raum schon vergeben“ auf dem Monitor ausgegeben werden.

a) Implementieren Sie eine Klasse `Raum`. Nutzen Sie die Klasse `Person`.

b) Sehen Sie die Möglichkeit vor, die Anzahl der freien Räume zu erfragen.

c) Sehen Sie die Möglichkeit vor, die „Auslastungsquote“ der insgesamt verfügbaren Plätze zu erfragen. Damit ist der Anteil der belegten Plätze (nicht der Räume) an der Gesamtzahl der Plätze aller Räume gemeint.

Beispiel:

Wenn alle Räume über 4 Plätze verfügen und alle Räume belegt sind, aber jeweils nur mit 2 Personen, dann ist die Auslastungsquote 50 %.