



Java-OOP - Mehrere Klassen - Vererbung

Aufgabe 1

Die Konten sollen weiter spezifiziert werden, indem sie in „Girokonto“ und „Sparkonto“ unterschieden werden.

- Welche Gemeinsamkeiten (Attribute und Methoden) haben beide Kontenarten? Worin unterscheiden sie sich?
- Die Gemeinsamkeiten werden zunächst in der Klasse „Konto“ abgebildet, diese wird Oberklasse. Müssen Änderungen vorgenommen werden?
- Die Unterschiede sind denn in den jeweiligen Unterklassen „Girokonto“ und „Sparkonto“ zu implementieren.

Vererbung – extends und protected

Damit eine (Unter)Klasse als Spezifizierung einer anderen Klasse deklariert werden kann, muss sie über `extends` als Unterklasse definiert werden: `public class Girokonto extends Konto{...}`
Sind die Attribute der Oberklasse mit dem Zugriffsmodifizierer `private` deklariert, so kann die Unterklasse nicht darauf zugreifen. Dies erscheint wenig sinnvoll, in diesem Fall sollten die Attribute in der Oberklasse statt mit `private` mit `protected` deklariert werden, das bedeutet, dass die Klasse und alle Unterklassen darauf zugreifen können.

Schaue nach, welches Symbol nun im UML-Diagramm vor den geänderten Attributen erscheint.
Der Konstruktor einer Klasse kann leider nicht vererbt werden, daher müssen die Unterklassen ggf. eigene Konstruktoren erhalten. Mit einem kleinen Trick, kann man dann aber doch auf den Konstruktor der Oberklasse zugreifen:

`super()`

Damit ist jetzt leider kein Lob für deine Lösung der ersten Aufgabe gemeint, auch wenn du es sicher verdient hättest. Mit dem Befehl `super ()` kann eine Unterklasse auf die Oberklasse zugreifen. Soll z.B. der Konstruktor der Oberklasse verwendet werden, kann dies mit Hilfe von `super ()` erreicht werden:

```
public class Girokonto extends Konto {  
  
    private [...];  
  
    public Girokonto(int a, double b, double d)  
    {  
        super(a,b); //Aufruf Konstruktor  
        this.[...]=d;  
    }  
}
```



Aufgabe 2

Die SchülerInnen unserer Schule sind teilweise Vollzeit da, teilweise in Teilzeit (Berufsschule). Außerdem werden sie von Lehrern unterrichtet und von den Sekretärinnen verwaltet.

- a) Analysiere zunächst, welche Gemeinsamkeiten die Personengruppen haben und stelle diese in einer Klasse `Person` dar.
- b) Unterscheide dann zunächst in Unterklassen Schüler, Lehrer und Sekretärin die weitere Spezifizierung.
- c) Schüler werden dann weiter in Vollzeit und Teilzeit spezifiziert. Auch hierfür werden entsprechende Klasse deklariert.

Hinweis: Bei dieser Aufgabe reicht das Erstellen der Klassen im UML-Modul des Java-Editors, mit geeigneten Beziehungen, eine Implementierung findet nicht statt.