

Übung 8



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Aufgabe 1 (Wissensfrage)

- a) Welche Art von Beziehungen wird durch (a) Attribute und (b) Vererbung abgebildet? Nennen Sie jeweils ein Beispiel.
- b) Welche Vorteile ergeben sich durch Vererbung bei der objektorientierten Programmierung? *Wiederverwendung, schrittweise von Generell zu Speziell-*
- c) Was wird bei der Vererbung von der Oberklasse auf die Subklasse übertragen? Gibt es Elemente der Oberklasse, die nicht auf die Subklasse übertragen werden?
- d) Was sind überschriebene Methoden und wie werden sie implementiert?
- e) Was ist Ihre Meinung zu folgender Aussage: Java ist eine typisierte Sprache. Daher können Objekte nur in solchen Variablen gespeichert werden, bei deren Deklaration genau der Typ des Objekts angegeben wurde.

(Protect, Public, Package) Attribute und Methode können vererbt werden.

*e) Statische Datentyp ist der Typ der Variablen,
Dynamische Datentyp ist der Typ des Objekts.*

Aufgabe 2 (Kochbuch)

- Erstellen Sie eine Klasse `Book`, die das Objektattribut `title` vom Datentyp `String` und das ganzzahlige Objektattribut `pagecount` enthält.
- Erstellen Sie in der Klasse `Book` eine Methode `toString()` mit dem Rückgabewert `String`, die den Buchtitel und die Seitenzahl zurückgibt.
- Erstellen Sie nun eine neue Klasse `SchoolBook`. Diese soll von der Klasse `Book` erben und ist damit Unterklasse von `Book`.
- Geben Sie der Unterklasse `SchoolBook` ein ganzzahliges Objektattribut `gradeLevel`. Das Attribut soll mit dem Wert `-1` initialisiert werden.
- Erstellen Sie nun eine weitere Klasse `CookBook`, die ebenso von der Klasse `Book` erbt und damit zur Unterklasse wird.
- Geben Sie der Klasse `CookBook` ein Objektattribut `vegetarian` vom Typ `boolean`. Implementieren Sie außerdem einen Konstruktor, der einen Parameter enthält, durch den das Attribut `vegetarian` mit dem übergebenen Wert belegt wird. Außerdem soll eine Objektmethode mit dem Bezeichner `isVegetarian` erstellt werden, die `true` zurückgibt, wenn das Kochbuch vegetarisch ist und `false`, wenn nicht.
- Überschreiben Sie nun innerhalb der Klasse `CookBook` die Methode `toString()`. Die überschreibende Methode soll wiederum einen `String` zurückgeben. Dieser soll aus dem Text der `toString()`-Methode aus der Oberklasse bestehen, ergänzt um die Angabe, ob das Kochbuch vegetarisch ist oder nicht. Verwenden Sie hierzu einen Verweis auf die Methode der Oberklasse, anstatt den `String` wiederholt zu schreiben.
- Wie lautet die Ausgabe der folgenden `main`-Methode?

```
1 public class BookManagementSystem {
2     public static void main(String[] args) {
3         SchoolBook myComputerBook = new SchoolBook();
4         myComputerBook.title = "Java ist auch eine Insel";
5         myComputerBook.pageCount = 1308;
6         myComputerBook.gradeLevel = 42;
7         System.out.println(myComputerBook);
8
9         CookBook myCookBook = new CookBook(true);
10        myCookBook.title = "Vegetarisch Kochen für
11        Fortgeschrittene";
12        myCookBook.pageCount = 84;
13        System.out.println(myCookBook);
14
15        Book myBook = myCookBook;
16        System.out.println(myBook);
17    }
18 }
```

Print 新 Object 时会自动调用
toString() 方法!

Java ist auch eine Insel 1308

Vegetarisch Kochen für Fortgeschrittene 84 Kochbuch vegetarisch?
true.

Gleich wie oben.

Aufgabe 3 (Programmieren – Verwaltung von Sparkonten)

Hinweis: Der Programmcode der Musterlösung zur Sparkonto-Aufgabe aus Übung 6 kann in Moodle heruntergeladen werden.

- a) In den vorherigen Übungen haben Sie bereits ein Programm zur Verwaltung von Konten und Sparkonten implementiert. Nachdem Sie sich mit dem Konzept der Vererbung auseinandergesetzt haben, fällt Ihnen auf, dass Ihre Lösung noch nicht optimal ist. Um zukünftig einfacher neue Typen von Konten hinzufügen zu können, möchten Sie alle speziellen Kontotypen von der Klasse `Konto` erben lassen.

Alle `Konto`-Objekte sollen über die Objektattribute `kontonummer` und `kontostand` verfügen. Das Attribut `kontonummer` soll bei der Erzeugung eines `Konto`-Objekts mit einem dem Konstruktor übergebenen Wert belegt werden, während der `kontostand` während der Erzeugung mit 0 initialisiert wird.

Die Klasse `Konto` soll außerdem drei Methoden definieren:

- `public boolean einzahlen(double)` – bereits in vorheriger Übung implementiert
- `public boolean auszahlen(double)` – bereits in vorheriger Übung implementiert
- `public void druckeKontoauszug()`

Zur Erinnerung: Die Methoden `einzahlen(double)` und `auszahlen(double)` erhalten jeweils den Betrag als Parameter übergeben und liefern einen Wert zurück, der angibt, ob die Transaktion ausgeführt wurde oder nicht. Diese Transaktionen können nicht ausgeführt werden, falls der übergebene Parameter einen negativen Wert hat. Eine Auszahlung kann nur erfolgen, wenn der Auszahlungsbetrag nicht größer ist als der Kontostand.

Die Methode `druckeKontoauszug()` soll Angaben über Kontonummer und Kontostand auf der Konsole ausgeben.

- b) Die (in der Musterlösung aus Übung 6 bereits existierende) Klasse `Sparkonto` soll nun von der Klasse `Konto` erben, sodass nur das Attribut `zinssatz` sowie die Methoden `verzinsen` und `jahresabschluss` in der Klasse `Sparkonto` implementiert sind.
- c) Mittlerweile hat ihr Auftraggeber entdeckt, dass sich mit Dispozinsen sehr gutes Geld verdienen lässt. Dazu möchte die Bank nun auch ein `Girokonto` anbieten. Die Klasse `Girokonto` verfügt über ein Attribut `dispoLimit` vom Datentyp `double`, das bei der Erzeugung eines `Girokontos` mit einem Wert belegt werden soll. Überschreiben Sie die Methode `auszahlen(double)` in der Klasse `Girokonto`, sodass auch negative Kontostände bis zum Erreichen des Dispolimits möglich sind.

Hinweis: Die Methode `auszahlen(double)` ist bereits durch die Ober-Klasse `Konto` implementiert, welche überprüft, dass das Konto ausreichend gedeckt ist.

- d) Außer diesen drei Klassen benötigen Sie ein Hauptprogramm (main-Methode) in der Klasse `Bank`. Zunächst soll der Zinssatz für Sparkonten auf 0,6% (0,006) gesetzt werden. Danach werden zwei `Girokonten` und ein `Sparkonto` erzeugt, die alle in einem Array gespeichert werden sollen. Die Werte für `Kontonummer` und `Dispolimit` (bei `Girokonten`) können Sie nach Belieben initialisieren. Auf diesen Konten sollen die folgenden Transaktionen ausgeführt werden. Nach jeder Transaktion soll für das betroffene Konto ein Kontoauszug gedruckt werden.

Hinweis: Machen Sie sich klar, unter welchen Umständen die Methode `verzinsen()`, auch auf einem Objekt der Klasse `Konto` (z.B. dem Konto „Spar“) aufgerufen werden kann.

Konto	Einzahlung	Auszahlung
Giro1	1000	
Giro1		1500
Giro2	200	
Giro2		700
Spar	5000	
Spar		verzinsen()

Aufgabe 1.

a) Attribute der Ober-Klasse wird auf Sub-Klasse vererbt.

Methode der Ober-Klasse ist automatisch auch eine Methode der abgeleiteten Klasse.

b) Wiederverwendung der Code.

Schrittweise Entwicklung vom Generellen zum Speziellen. (Abstrakt \rightarrow Konkret)

c) Methode und Attribute werden auf die Subklassen übertragen, aber Konstruktor nicht!

d) Im Rahmen der Subklasse wird geerbte Methode neu konzipiert und implementiert. Dieser Vorgang heißt „Überschreiben“.

- Beim Aufruf einer überschriebenen Methode der Subklasse wird die neu implementierte Methode aufgerufen.
- Bei Referenz „super“ wird die überschriebene Methode der Ober-Klasse zugegriffen.

e) Die Aussage ist falsch. Es ist bei Vererbung möglich, ein Subklasse relevante Objekt in Variable, deren Datentyp Oberklasse ist, zu speichern.

(Polymorphismus)