

Übung 5



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Aufgabe 1 (Wissensfrage)

- a) Was ist der Unterschied zwischen Klassen und Objekten?
- b) Was sind Konstruktoren, welchen Zweck erfüllen sie und welche Eigenschaften haben alle Konstruktoren gemein? Wie werden Konstruktoren aufgerufen?
- c) Aus welchen Bestandteilen setzt sich eine Klasse in Java zusammen?
- d) Konstruktoren sind letztlich besondere Methoden. Sie haben gelernt, dass Methoden aus fünf Elementen bestehen. Welches wird bei Konstruktoren **nicht** mit angegeben?

Aufgabe 2 (Datum) *注意if和else if的区别!*

- a) Schreiben Sie eine Klasse mit dem Bezeichner `Datum`. Jedes Objekt dieser Klasse soll über die Attribute `tag`, `monat` und `jahr` vom Datentyp `int` verfügen. Legen Sie hierzu die nötigen Attribute an. (Nutzen Sie zur Deklaration der Attribute `public int ...`).
- b) Erstellen Sie nun einen Konstruktor, der drei Parameter entgegennimmt. Die Datentypen der drei Parameter entsprechen den Datentypen der drei Attribute. Wählen Sie für die Parameter eigene Bezeichner, welche sich von den Bezeichnern der Attribute unterscheiden. Welchen Bezeichner muss der Konstruktor haben?
- c) Im Konstruktor soll zunächst eine einfache Überprüfung vorgenommen werden, ob der Parameter für Monatszahl zwischen 1 und 12 liegt und die Tageszahl zwischen 1 und 31. Ist dies der Fall, so sollen die Attribute entsprechend gesetzt werden. Ansonsten wird eine Fehlermeldung auf der Konsole ausgegeben.
- d) Erweitern Sie die Klasse `Datum` um eine Methode `druckeDatum`, welche das Datum auf der Konsole ausgibt. Nutzen Sie den Modifier `public`. Gibt die Methode einen Wert zurück? Welcher Rückgabotyp muss somit verwendet werden?
- e) Schreiben Sie ein Hauptprogramm (main-Methode) in einer neuen, separaten Klasse mit dem Bezeichner `DatumApp`. In dieser Klasse benötigen Sie keine Attribute und keinen Konstruktor, aber eine main-Methode. Die main-Methode soll die folgenden Aufrufe enthalten:
 - a. Deklarieren Sie eine Variable vom Typ `Datum`. Wählen Sie als Bezeichner `meinDatum`.
 - b. Erzeugen Sie nun ein neues Objekt der Klasse `Datum` und weisen dieses der Variablen `meinDatum` zu. Rufen Sie hierzu den Konstruktor auf, der die Attribute des neu erzeugten Objekts auf das Datum des 1. Januar 1900 setzt.
 - c. Lassen Sie sich das im Objekt gespeicherte Datum des soeben erzeugten `Datum`-Objekts mithilfe eines Aufrufs der Methode `druckeDatum` ausgeben.

Aufgabe 3 (Kontoverwaltung)

Sie sollen eine neue Kontoverwaltungssoftware für eine Bank implementieren. Sie haben sich überlegt, dass folgendes Vorgehen dabei sinnvoll wäre:

- Implementieren Sie zunächst eine Klasse `Konto`. Ein `Konto` soll über ein Attribut `kontonummer` vom Datentyp `int` und ein Attribut `kontostand` vom Datentyp `double` verfügen. Verwenden Sie für beide Attribute den Modifier `public`.
- Schreiben Sie nun einen Konstruktor für die Klasse `Konto`. Dem Konstruktor soll als einziger Parameter ein ganzzahliger Wert übergeben werden können, der dem Attribut `kontonummer` zugewiesen wird. Das Attribut `kontostand` soll beim Anlegen eines neuen Kontos immer auf 0 gesetzt werden.
- Auf die Konten soll natürlich auch Geld eingezahlt werden können. Schreiben Sie dazu in der Klasse `Konto` eine Methode `einzahlen`, die den auf das Konto einzuzahlenden Betrag als Parameter vom Datentyp `double` erhält. Die Einzahlung soll aber nur dann tatsächlich ausgeführt werden, wenn der übergebene Betrag positiv ist. Um dem Aufrufenden zu signalisieren, ob die Transaktion durchgeführt wurde, soll ein `boolean`-Wert zurückgegeben werden. Falls die Transaktion stattgefunden hat soll `true`, falls nicht `false` zurückgegeben werden.
- Um auch wieder Geld auszahlen zu können benötigen Sie eine zweite Methode `auszahlen`. Auch diese erhält den auszuzahlenden Betrag als Parameter (Gleitkommazahl) und soll den Kontostand nur dann verringern, wenn der übergebene Betrag positiv ist und auf dem Konto noch ausreichend Geld vorhanden ist (der Kontostand darf nach der Auszahlung nicht negativ werden). Auch hier soll eine erfolgreiche Transaktion durch Rückgabe des `boolean`-Wertes `true` und eine nicht erfolgreiche durch `false` signalisiert werden.
- Erstellen Sie eine neue Klasse `KontoVerwaltung`, die die `main`-Methode enthält. Dort sollen für fünf Kunden der Bank je ein neues `Konto`-Objekt angelegt werden. Vergeben Sie die Kontonummern 111, 222, 333, 444, 555. Lassen Sie anschließend für alle fünf Konten die Kontonummer sowie den zugehörigen Kontostand auf der Konsole ausgeben.
- Um ihr Programm zu testen, sollen die untenstehenden Buchungen auf den Konten mit den jeweiligen Kontonummern ausgeführt werden. Nutzen Sie hierzu die von Ihnen gewählten Bezeichner für die `Konto`-Objekte. Wenn eine Buchung fehlschlägt, geben Sie eine Fehlermeldung auf der Konsole aus: „Die Buchung konnte nicht ausgeführt werden“.

Kontonummer	Einzahlung	Auszahlung
111	500,00 €	
222	-5,00 €	
333	46,50 €	
444		5,60 €
111		42,00 €
555	4,33 €	

Beispiel: `erstesKonto.einzahlen(500);` // *Einzahlung auf das Konto 111*

- Lassen Sie nun ein zweites Mal für alle Konten die Kontonummer mit den entsprechenden Kontoständen ausgeben.

Aufgabe 1.

a) - Klasse ermöglicht strukturierte Datentype.

Objekte sind Exemplaren der Klasse mit Identitäten.

Sie sind die Werte.

- Jedes Objekt ist Instanz genau einer Klasse, ?
aber eine Klasse kann beliebig viele Instanzen besitzen.

- Alle Objekte einer Klasse besitzen die gleichen Methoden
und haben daher das gleiche Verhalten.

- Alle Objekte einer Klasse besitzen die gleichen Attribute,
allerdings mit unterschiedlichen Werten (Zustand)

b) Konstruktoren sind voreingestellte Java Methode, dadurch
wird eigenes Objekt bei Übergabe der Parameter
erstellt und initialisiert.

Konstruktor sind immer öffentlich, haben keine Rückgabtyp
und heißt wie Klasse. Mittels „new“ werden sie aufgerufen.

c).

Modifizier + ^(class) Schlüsselwort + Bezeichner + Attribute + Methode

- Attribute: Was **haben** Objekte einer Klasse.

- Methode: Was **können** Objekte einer Klasse.

d) Es gibt **kein Rückgabetypp** bei Konstruktoren.

(Aber, return an Object instead of special Datatype?)