

JAVA-ÜBUNGSAUFGABEN VERERBUNG 1

1)

Entwickeln Sie ein Programm, das verschiedene Kontenvorgänge, die für Konten anfällt, abarbeiten kann.

Das Programmdesign und die Lösung soll **objektorientiert** sein.

Das Klassendesign soll wie folgt aufgebaut sein:

(die get- und set-Methoden fehlen, müssen aber trotzdem implementiert werden).

Konto
kontostand: double
- inhaber : String
+ Konto(inhaber : String)
+ einzahlen(wert : double): void
+ abheben(wert : double) : void)
+ scan(eingabe : String) : String
+ print() : void

Bemerkung:

Die Methode scan(...) soll genau dann verwendet werden, wenn Eingaben über Tastatur gemacht werden sollen. es soll genau 3 verschiedene Parameter geben: "n", "e", "a"

Beispiele:

String str;

str = scan("n")

Der Name des Kontoinhabers soll über Tastatur eingegeben werden (z.B. nach einer Namensänderung wie bei einer Verheiratung).

str = scan("e")

Einzahlung eines Geldbetrags über Tastatur.

Der Geldbetrag wird als String zurückgegeben.

str = scan("a")

Auszahlung eines Geldbetrags über Tastatur.

Der Geldbetrag wird als String zurückgegeben.

Die Methode print() gibt den Name des Inhabers mit seinem Kontostand auf dem Bildschirm

a) Entwerfen Sie die Klasse Konto mit ihren Eigenschaften und Methoden.

Bei der Instanzierung soll das Konto immer mit dem Kontostand 0 beginnen.

Die Methode "abheben" muß ein Überziehen des Kontos auf über 5000 Euro Schulden verhindern.

a1) Erzeugen Sie ein Konto für den Kontoinhabers Uwe Mustermann.

a2) Erzeugen Sie die folgenden Kontenbewegungen des Kontoinhabers Uwe Mustermann (mit zugehöriger Bildschirmausgaben) durch entsprechende Methodenaufrufe:

Kontostand und Kontoinhaber ausgeben,

einzahlen 300 Euro (nicht über Tastatur), Kontostand und Kontoinhaber ausgeben,

abheben 400 Euro (nicht über Tastatur), Kontostand und Kontoinhaber ausgeben,

abheben 7000 Euro (nicht über Tastatur), Kontostand und Kontoinhaber ausgeben,

einzahlen 500 Euro (nicht über Tastatur), Kontostand und Kontoinhaber ausgeben

abheben 100 Euro (über Tastatur), Kontostand und Kontoinhaber ausgeben,

einzahlen 200 Euro (über Tastatur), Kontostand und Kontoinhaber ausgeben

Uwe Mustermann hat geheiratet. Er heißt jetzt Oweh Frustmann

Diese Namensänderung mit der Methode `scan(...)` veranlassen. Danach Kontostand und Kontoinhaber ausgeben.

b) Erzeugen Sie eine weitere Klasse Sparkonto. Verwenden Sie (um Schreibarbeit zu sparen) Methoden der Klasse Konto (Wiederverwendung von Software), indem die Klasse Sparkonto von der Klasse Konto erbt.

Der Zinssatz ist ein Tageszinssatz und wird z.B. auf 1% festgelegt. Vor jeder neuen Kontobewegung wird der Zinsgewinn zum Kontostand addiert. Als Parameter wird bei der Methode "zinsberechnung" die Anzahl der Tage (die das Geld schon auf dem Konto liegt) vorgegeben

Der Zinsgewinn berechnet sich wie folgt:

$$\text{Zinsgewinn} = \text{Kontostand} / 100 * \text{Zinssatz} * \text{Tage}$$

Für ein Sparkonto muß außerdem sichergestellt sein, daß das Konto nicht überzogen (nicht negativ) werden darf .

b1) Erzeugen Sie die folgenden Kontenbewegungen des Kontoinhabers Uwe Mustermann (mit zugehöriger Bildschirmausgaben) durch Methodenaufrufe:

Kontostand ausgeben, einzahlen 300 Euro, Kontostand ausgeben,

100 Tage Geld auf der Bank lassen, abheben 400 Euro, Kontostand und Kontoinhaber ausgeben

100 Tage Geld auf der Bank lassen , abheben 7000 Euro, Kontostand und Kontoinhaber ausgeben,

200 Tage Geld auf der Bank lassen , einzahlen 500 Euro, Kontostand und Kontoinhaber ausgeben

b2)

Lösen Sie b1), wenn in der Klasse Konto das Attribut "kontostand" nicht protected, sondern private wäre. Was müßte in den Klassen Konto und Sparkonto geändert werden ?

c) Modellieren Sie das Attribut inhaber durch ein Array von Zeichen:

- inhaber[20]: char

(neues Projekt anlegen)

Schauen Sie dazu in der Java-Dokumentation die Klasse String an, ob diese Methoden liefert, die Strings in char-Arrays umwandelt (und umgekehrt). Zum Beispiel: die Methode

toCharArray() bzw.

String.valueOf(inhaber)

2)

Das Dualsystem und das Hexadezimalsystem sind Zahlensysteme.

Modellieren Sie diesen Sachverhalt, indem Sie dazu ein UML-Diagramm erstellen.

Erzeugen Sie dazu ein Java-Programm.

3)

Ein Fußballverein besteht aus einer Fußballmannschaft, aus Spielern, also Verteidigern und Angreifern und einem Trainer.

Modellieren Sie diesen Sachverhalt, indem Sie dazu ein UML-Diagramm erstellen.

Erzeugen Sie dazu ein Java-Programm.

4)

Modellieren Sie einen Bauernhof bzw. einen Zoo mit Hilfe der objektorientierten Programmierung (siehe Übungen zu Klassen). Benutzen Sie dazu die Vererbung.