

**Information**

Alle in diesem Dokument enthaltenen Aufgaben gehören zu einer komplexen Aufgabenstellung und sind zum Teil voneinander abhängig. Ziel ist die schrittweise Entwicklung einer Software.

**1. Aufgabe****Anforderungsbeschreibung**

Eine Bank verwaltet viele Konten (nur Girokonten) und Kunden. Zu jedem Konto gehören eine Kontonummer (vom Typ IBAN) und ein Kontostand. Zu jedem Kunden gehören eine Kundennummer (z. B. 4711), ein Vor- und ein Nachname. Jeder Kunde besitzt mindestens ein Konto.

Ein Kunde kann seinen Kontostand abfragen und auch Geld auf sein Konto einzahlen. Es ist auch möglich, dass er von seinem Konto Geld abhebt, wenn dabei ein Kontostand von mindestens 1 € bestehen bleibt. Von seinem Konto kann der Kunde Geld auf ein anderes Konto überweisen. Eine Überweisung beinhaltet immer das Abheben eines Geldbetrages vom eigenen Konto und das Einzahlen auf ein anderes Konto.

**Aufgabenstellung**

- a) Erstellen Sie anhand der Anforderungsbeschreibung ein Klassendiagramm unter Beachtung folgender Vorgaben:
- Für die Verwaltung von Konten und Kunden muss eine Klasse `Bank` existieren. Für diese Klasse müssen hier noch keine Attribute, Konstruktoren und Methoden angegeben werden. Alle anderen Klassen müssen Attribute, die zugehörigen get-Methoden und einen Konstruktor besitzen. Dieser Konstruktor muss allen Attributen einen Wert zuweisen können. Weitere Methoden sind hier noch nicht erforderlich.
  - Das Klassendiagramm muss eine maximale Aussagekraft besitzen, das betrifft insbesondere die Angabe von Datentypen und Sichtbarkeiten sowie die Beschriftung von Assoziationen. Falls diesbezügliche Informationen in der Anforderungsbeschreibung fehlen, so sind diese selbständig zu ergänzen.
- b) Erstellen Sie anhand des Klassendiagramms von a) den zugehörigen Javacode.
- c) Die Bank verwaltet zwei Konten und einen Kunden. Dieser Kunde ist der Eigentümer von den beiden Konten. Erstellen Sie ein entsprechendes Objektdiagramm. Allen Attributen müssen sinnvolle Werte zugeordnet werden.
- d) Erstellen Sie anhand der Anforderungsbeschreibung ein Anwendungsfalldiagramm.
- e) Wie sollen die in d) beschriebenen Anwendungsfälle im Programmcode realisiert werden? Diskutieren Sie diese Frage zunächst untereinander und anschließend mit der Lehrkraft.
- f) Ergänzen Sie anhand der Ergebnisse von e) das in a) erstellte Klassendiagramm und den zugehörigen Code.
- g) Der Anwendungsfall `überweisen` soll mittels Sequenzdiagramm dargestellt werden. Es existieren zwei Konten (Konto-Objekte), `k1` und `k2`. Auf beiden Konten befinden sich je 300 €. Erstellen Sie für die nachfolgend genannten Abläufe jeweils ein Sequenzdiagramm.

Hinweis/Vereinfachung:

Der Kunde startet die Abläufe von der Main-Methode aus, es wird kein Kunden-Objekt benötigt.

I Der Kunde möchte von seinem Konto `k1` 200 € auf das Konto `k2` überweisen.

II Der Kunde möchte von seinem Konto `k1` 500 € auf das Konto `k2` überweisen.

III Der Kunde möchte von seinem Konto `k1` einen beliebigen Betrag auf das Konto `k2` überweisen.

## **2. Aufgabe**

Jetzt soll die Verwaltung von Kunden und Konten durch die Bankangestellten betrachtet werden. Eine diesbezügliche Anforderungsbeschreibung gibt es nicht. Vielmehr sollen mögliche Anwendungsfälle für Bankangestellte im Unterrichtsgespräch ermittelt und festgelegt werden.

### **Teilaufgaben:**

- a) Nennen Sie typische Anwendungsfälle aus der Sicht der Bankangestellten.

Hinweis:

Nachdem eine Vielzahl von Anwendungsfällen im Unterrichtsgespräch zusammengetragen wurde, erfolgt eventuell eine Präzisierung derselben. Möglicherweise werden auch Anwendungsfälle gestrichen, damit der Umfang der zu erstellenden Software nicht zu groß wird. Beachten Sie das bitte bei der Bearbeitung der nachfolgenden Teilaufgaben.

- b) Erstellen Sie für den Akteur "Bankangestellter" ein Anwendungsfalldiagramm.

- c) Aus den in b) dargestellten Anwendungsfällen sind passende Methoden für die Klasse `Bank` zu erstellen. Vervollständigen Sie zunächst das in den Aufgaben 1 und 2 entwickelte Klassendiagramm. Implementieren Sie anschließend die o. g. Methoden und testen Sie diese.

d) **Zusatzaufgabe:**

Manchmal müssen Kunden anhand von Vor- und/oder Nachnamen gesucht werden. Implementieren Sie für diesen Anwendungsfall eine Methode in der Klasse `Bank` und testen Sie diese. Die Methode soll eine Liste mit allen Kundenobjekten zurückgeben, welche der Suche entsprechen.

e) **Zusatzaufgabe:**

Erweiterung der Aufgabe d): In der zurückgegebenen Liste müssen die Kundenobjekte bzgl. der Attribute Nachname und Vorname alphabetisch sortiert gespeichert sein.