



Arbeitsstand

Das Bank-Projekt soll Transaktionen zwischen Konten von Kunden nach den Geschäftsregeln der Wossi-Bank simulieren. In Analogie zur Datenbankentwicklung wurden bislang die Klassen **Kunde** und **Konto** entworfen.

Beim Schließen des Projekts gehen jedoch alle Daten und Transaktionen verloren. In der Realität werden Daten extern in Dateien oder Datenbanken gespeichert. Dies soll im Projekt mithilfe der Klassen **Verwalter** – zum Managen aller Objekte der Klassen **Kunde** und **Konto** – und **Datenzugriff** – zum Speichern und Einlesen der Daten – geschehen.

Klasse Datenzugriff

- 1) Kopieren Sie Klasse **Datenzugriff** sowie die CSV-Dateien **Konten** und **Kunden** in ihren Projektordner.
- 2) Öffnen Sie die CSV-Dateien mit dem Editor oder Notepad. Beschreiben Sie den Inhalt der beiden CSV-Dateien. Vergleichen Sie mit den Eigenschaften eines Kunden bzw. Kontos. Merken Sie sich den Saldo von Konto 1. Schließen Sie die Dateien.
- 3) Öffnen Sie das BlueJ-Projekt. Die Klasse **Datenzugriff** erscheint im Projekt. Beseitigen Sie ggf. vorhandene Fehler wegen fehlender Passung der Bezeichner, des Konstruktors, der Methodenarten oder der Parameterreihenfolge in Ihren Klassen **Kunde** und **Konto**.
- 4) Erstellen Sie ein Objekt der Klasse **Datenzugriff** und lesen sie darüber Daten ein. Inspizieren Sie das Objekt. Beschreiben Sie die Attribute des Objekts. Erstellen Sie eine Klassenkarte der Klasse. Markieren Sie Aufträge/Anfragen sowie Attribute unterschiedlich.
- 5) Schließen Sie das BlueJ-Projekt.

Klasse Verwalter

- 6) Fügen Sie über das Menü „Bearbeiten – Klasse aus Datei hinzufügen ...“ die Klasse **Verwalter** in Ihr Projekt ein.
- 7) In der Klasse **Verwalter** werden – wie bereits in der Klasse **Datenzugriff** – Listen eingesetzt. Wiederholen Sie mithilfe der Tafelwerksergänzung Listen
 - a) die Voraussetzungen für die Verwendung von Listen,
 - b) Möglichkeiten zum effizienten Durchlauf durch Listen.
- 8) Prüfen Sie, ob die Voraussetzungen für die Verwendung von Listen vorliegen und korrigieren Sie gegebenenfalls. Vergleichen Sie ihre Erkenntnis zum Listendurchlauf mit der Realisierung in der Klasse. Beschreiben Sie verbal den Algorithmus zum Listendurchlauf.
- 9) Erzeugen Sie ein neues Verwalterobjekt und inspizieren Sie es. Beschreiben Sie das Objekt. Zahlen Sie etwas Geld auf das Konto mit der Nummer 1 ein. Speichern Sie die Daten über die Methode **sichereDaten**. Kontrollieren Sie die beiden CSV-Dateien. Vergleichen Sie mit dem notierten Saldo von Konto 1.
- 10) Analysieren Sie die Methode zum Einzahlen eines Geldbetrags auf ein beliebiges Konto. Beschreiben Sie die Arbeitsweise der Methode.
- 11) Erläutern Sie Unterschiede zwischen den Beziehungen der Klassen Konto-Kunde und Verwalter-Konto.



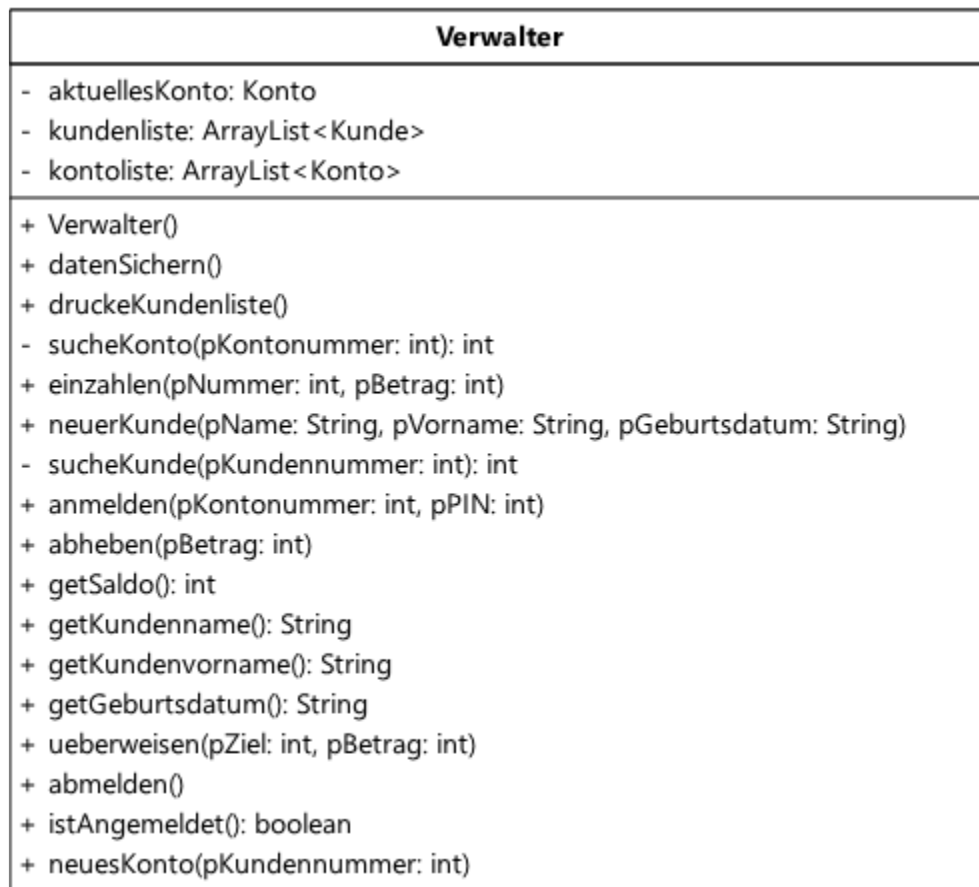
Projekt Bankverwaltung 2025/26

Name:

Vorname:

Klasse:

12) Entwickeln Sie auf Basis der Methoden **einzahlen** und **sucheKonto** die restlichen Methoden gemäß Klassendiagramm.



13) Vervollständigen Sie alle Java-Doc-Kommentare.

14) Entwickeln Sie das neue Klassendiagramm mit allen Beziehungen. Ermitteln Sie die Kardinalität der Beziehungen.

15) Informieren Sie sich unter <http://www.oop-uml.de/drei-schichten-architektur.php> oder/und <http://www.mrknowing.com/2013/11/08/wie-funktioniert-die-3-schichten-architektur/> über die Drei-Schicht-Architektur von Software und fassen Sie die Aussagen geeignet zusammen. Geben Sie an, welche Schichten im Projekt realisiert wurden. Begründen Sie.