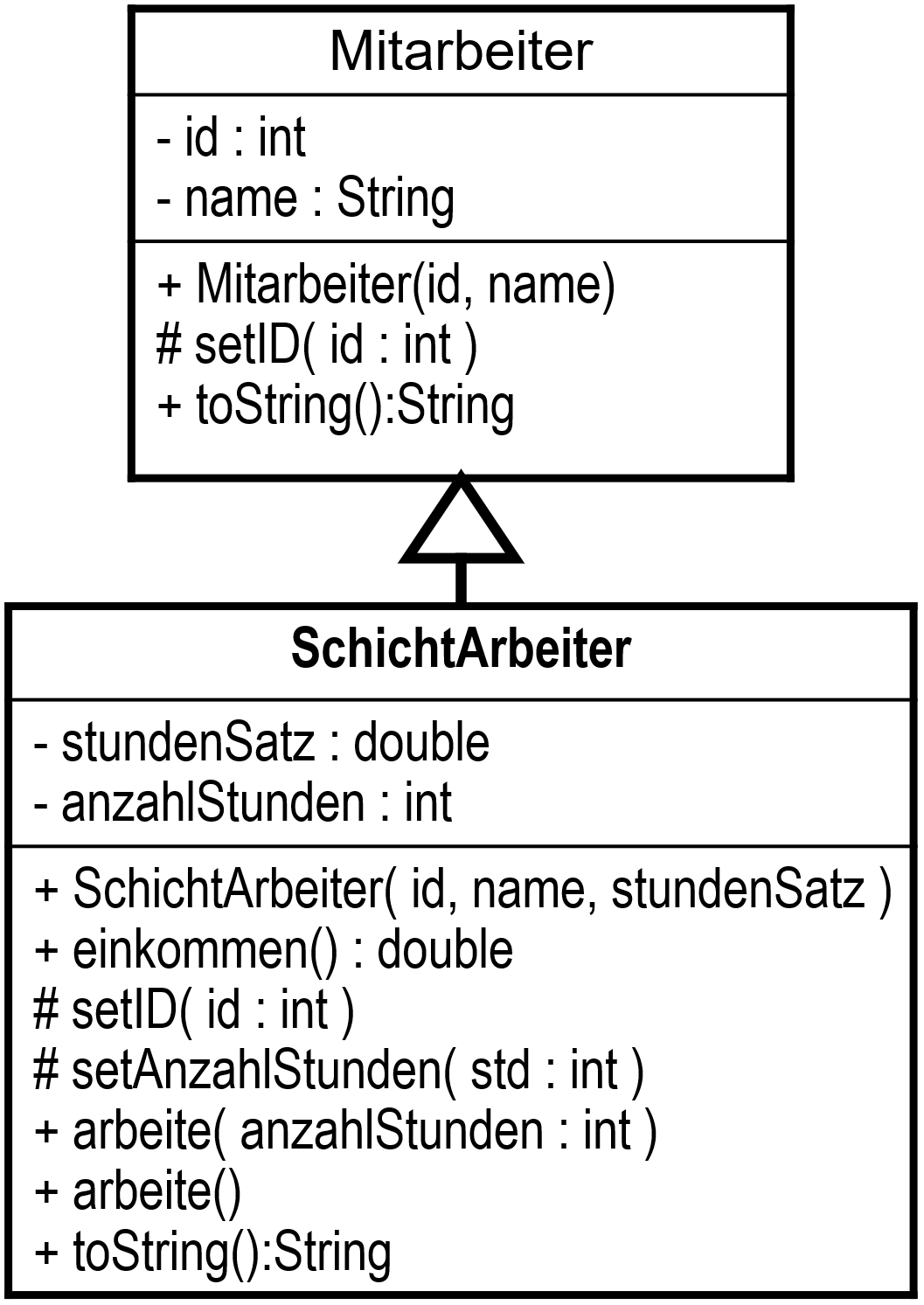
Herr Weiß, der der Projektleiter für individuelle Kundensoftware, möchte, dass Sie ihn bei der Durchführung eines neuen Projekt-Auftrags unterstützen. Die „Aix-Car-Group Ltd.“, war so angetan von der individuellen Software für Mietwagen, dass er Sie darum gebeten hat, eine Software für die Verwaltung von Gehältern ihrer unterschiedlichen Mitarbeiter zu unterstützen.

Schon wieder waren Sie, bedingt durch den Berufsschulunterricht, weder bei den Gesprächen mit dem Kunden noch bei den ersten Design-Entwürfen dabei. Herr Weiß übergibt Ihnen das Gesprächsprotokoll und den unfertigen Entwurf Ihrer Fachkollegen und bittet Sie sich mit den Anforderungen vertraut zu machen.

Gesprächsprotokoll des Kundengesprächs

Anwesend: Herr Weiß und Frau Kallen

* Die Software soll je nach Mitarbeitertyp die Gehälter unterschiedlich bestimmen können. Schichtarbeiter werden nach Stunden und Stundensatz bezahlt. **Büroarbeiter** erhalten ein Festgehalt. **Manager** erhalten neben einem Festgehalt auch einen Bonus, abhängig von einem individuell vereinbarten Prozentsatz. Der Bonus soll auch unabhängig vom Gehalt berechnet werden können.
* Da **Schichtarbeiter** erst bezahlt werden, nachdem Sie gearbeitet haben, soll es eine Methode arbeite() geben, welche die Anzahl der Stunden erhöht. Es soll dafür zwei Varianten geben. Ohne Parameter erhöht die Methode die Anzahl Stunden um 8. Die zweite Variante erhält die Anzahl gearbeiteter Stunden.
* IDs werden einmalig erstellt und unterliegen bestimmten Regeln:
  + Mitarbeiter IDs sind immer positiv und vierstellig.
  + Schichtarbeiter IDs sind positiv, vierstellig und beginnen mit der Ziffer 3.
  + Büroarbeiter IDs sind positiv, vierstellig und beginnen mit 5.
* Eine Verwaltung der Mitarbeiter soll in der Lage sein,
  + Eine Gehaltsliste mit abschließender Gehaltssumme zu erstellen,
  + Neue Mitarbeiter einzustellen und
  + Mitarbeiter zu entlassen.



**Aufgabe 1**

Informieren Sie sich über Vererbung und halten Sie Fragen schriftlich fest. Nutzen Sie dafür die **beigelegten Kontrollfragen.**

1. Recherchieren Sie diese merkwürdigen **Pfeile**, die in die falsche Richtung zu zeigen scheinen, das **#-**Symbol und was das mit dieser **Methodenwiederholung** auf sich hat. Über **Konstruktoren** musst Du auch noch mehr wissen.   
   Alternativ – Lasse Dich am BWV fortbilden.   
     
   Ergänzen Sie anschließend Ihr **Klassendiagramm** um Büroarbeiter und Manager.
2. **Recherchier die Pfeile.**
3. **Müssen alle Methoden in Schichtarbeiter wiederholt werden?**
4. **Kann man den Konstruktor von Mitarbeiter auch in Schichtarbeiter verwenden?**
5. **Was soll das # Zeichen vor dem setID() ?**
6. Implementieren Sie anschließend alle neuen Mitarbeiter-Klassen. Legen Sie alle Mitarbeiterklassen in ein Package mit Namen mitarbeiter. Die Main-Klasse (Verwaltung) sollte im default-Package bleiben.😉

**Aufgabe 2**

Bevor wir den Mitarbeiter-Teil des Programms weiterbearbeiten– die Leute von Aix-Car wollen auch **unterschiedliche Kfz** in ihrer Software berücksichtigen.

1. Ergänzen Sie zuerst das **Klassendiagramm** für Kfz gemäß des folgenden Gesprächsprotokolls:

Gesprächsprotokoll des Kundengesprächs

Anwesend: Herr Weiß und Frau Kallen

Ein **Kfz** hat einen maximalen und einen tatsächlichen Tankinhalt. Ein Kfz kann mit Angabe eines Benzinvolumens aufgetankt werden – aber nie mehr als die Tankgröße. Wird zu viel getankt, läuft der Tank über. Das überlaufende Benzinvolumen wird zurückgegeben. Es kann nur gefahren werden, wenn der Tank nicht leer ist. Wenn der Motor nicht an ist, muss er angestellt werden.

Ein **Personentransportfahrzeug** ist ein Kfz, hat eine Anzahl Sitze (maximale Passagierzahl) und eine Anzahl Passagiere. Beim **Einsteigen der Passagiere** wird überwacht, ob sie noch reinpassen. Passen sie nicht rein schlägt das Einsteigen fehl. Personen können auch aussteigen.

Ein **Bus** ist ein Personentransportfahrzeug. Ein Bus benötigt einen Fahrer, wobei erstmal nur der Name des Fahrers gespeichert wird.

**Lkws** haben eine Ladefläche mit einer bestimmten Größe in Quadratmetern. Lkws können beladen und entladen werden, wobei immer berücksichtigt wird, dass die maximale Ladefläche nicht überschritten werden darf und dass nicht mehr entladen werden kann als auf der Ladefläche steht.

1. Implementieren Sie jetzt Ihr Modell in Java.