

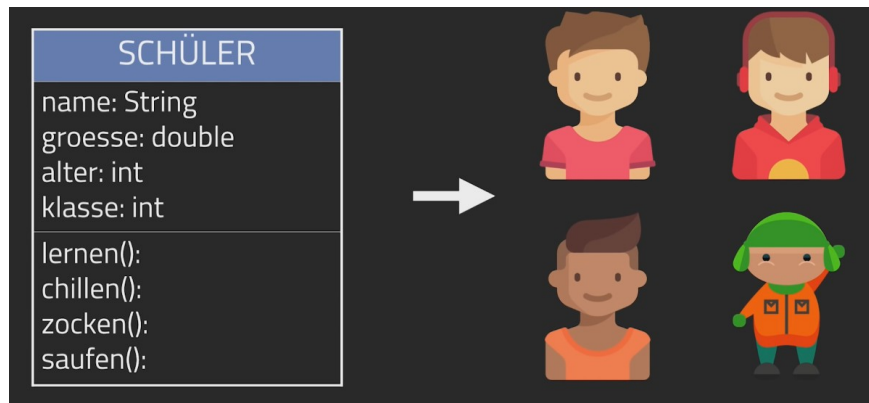
# Aufgabenblatt 1

## Aufgabe 1 – Was ist Objekt-Orientierung?

Sehen Sie sich den Einführungs-Video zur Objekt-Orientierung an und beantworten Sie mit Hilfe der Informationen aus dem Video die folgenden Fragen:

- Beschreiben Sie mit eigenen Worten den Unterschied zwischen einem Objekt und einer Klasse.
- Bei der Objekt-Orientierung besitzen Klassen Attribute und Methoden. Was steckt programmiertechnisch dahinter?
- Was versteht man unter Vererbung?

## Aufgabe 2 - Schüler-Klasse implementieren

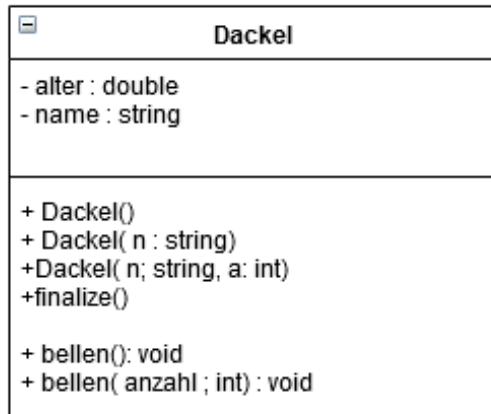


Programmieren Sie das die Klasse `Schueler` aus dem Video (Aufgabe 1) mit Java. Erstellen Sie zusätzlich eine Klasse `SchuelerTest` mit einer `main`-Methode. In der `main`-Methode sollen verschiedene Schüler erstellt werden und etwas machen.

- Speichere das Programm unter dem Namen **aufgabe\_ls3\_01a** ab.

### Fortgeschrittene Schüler:

- Erweitern Sie die Klasse `Schüler` um geeignete Konstruktoren und einen Destruktor. Bereiten Sie sich darauf vor Ihren Mitschülern zu erklären, was Konstruktoren und Destruktoren sind.

**Aufgabe 3 - Dackelklasse**

Bringe den Dackel zum Bellen.

- Implementieren Sie die Klasse `Dackel` und die Testklasse `DackelTest`. Führen Sie das Programm aus.
  - Erweitern Sie die Testfunktion, indem Sie mehrere Dackel erzeugen und bellen lassen.
  - Erweitern Sie die Klasse `Dackel` um die Methode `wieAltBistDu()`. Bei Aufruf dieser Methode soll der Dackel so oft „wau“ sagen, wie der Dackel alt ist.
- Speichere das Programm unter dem Namen **aufgabe\_ls3\_02a** ab.

**Fortgeschrittene Schüler:**

- Erweitern Sie die Klasse `Dackel` um die Funktion `beissen( Dackel d)`. Hierbei ist `d` der Dackel, der gebissen wird. Bei Aufruf dieser Funktion soll der beißende Dackel „Fass!“ und der gebissene Dackel „Aua!“ sagen.
  - Der gebissene Dackel soll zurückbeissen können. Ergänzen Sie hierzu ein Attribut `Lebenspunkte`, welches bei der Erstellung eines Dackels mit einem Zufallswert gesetzt wird und beim gebissen werden reduziert wird. Sollten die Lebenspunkte eine bestimmte Schwelle unterschreiten flieht der Dackel anstatt zurückzubeissen. Ergänzen Sie die Methodenparameter geeignet, so dass der gebissene Dackel weiss, von wem er gebissen wurde.
- Speichere das Programm unter dem Namen **aufgabe\_ls3\_02b** ab.

**Aufgabe 4 - Dackelklasse (II)****Anfänger:**

Erweitern Sie Ihre Klasse `Dackel` wie folgt:

1. Ergänzen Sie das Attribut `energievorrat` vom Typ `int`. Ergänzen Sie auch eine entsprechende `get`-Methode und eine `set`-Methode.
  2. Erweitern Sie die Konstruktoren, so dass der `energievorrat` eines neuen Dackels 50 beträgt, wenn nichts anderes als Parameter mitgegeben wird. Wird bei einem Konstruktor der `energievorrat` als Parameter mitgegeben, so bekommt der Dackel diesen Wert. Ist der mitgegebene Wert größer als 100, so wird der Energievorrat auf 100 gesetzt.
  3. Sofern noch nicht vorhanden, erweitern Sie die Konstruktoren, so dass als Konsolenmeldung mitgeteilt wird, dass ein neuer Dackel geboren wurde und welche Startwerte er besitzt.
  4. Ergänzen Sie eine Methode `fressen( int menge)`. Wenn die `menge` größer gleich 10 ist, so wird der `energievorrat` um 10 vergrößert, ansonsten um die `menge`. Geben Sie auf der Konsole aus, welche `menge` der Dackel frisst und wie groß der `energievorrat` vor und nach dem fressen ist. Sorgen Sie dafür, dass der `energievorrat` nie größer als 100 wird.
  5. Ergänzen Sie eine Methode `int laufen( int entfernung)`. Ist der `energievorrat` bereits kleiner gleich 5, so ist der Dackel zu müde zum Laufen und sagt dies auch. Um eine Entfernungseinheit zu laufen braucht der Dackel 2 Energieeinheiten. Ist genug Energie vorhanden, so läuft der Dackel die volle Entfernung, ansonsten nur soviel, wie der `energievorrat` erlaubt. Geben Sie auf der Konsole aus, wie weit der Dackel läuft und wie groß der `energievorrat` danach ist. Wie weit tatsächlich gelaufen wurde, wird als Rückgabewert von der Methode zurückgegeben.
- Speichere das Programm unter dem Namen **aufgabe\_ls3\_03a** ab.

**Fortgeschrittene Schüler:**

- Erweitern Sie Ihre Methoden zur Beißerei geeignet. Ein Dackel mit viel Energie soll seine lebenspunkte langsamer abbauen, als eine Dackel mit wenig Energie. Sowohl das Beißen, als auch das Fliehen soll Energie kosten.
- Speichere das Programm unter dem Namen **aufgabe\_ls3\_03b** ab.

**Aufgabe 5 - Kampf- und Schmuse-Dackel****Anfänger:**

Erweitern Sie Ihre Klasse `Dackel` wie folgt:

1. Ergänzen Sie eine Methode `void beissen()`. Beim Beissen reduziert sich der Energievorrat um 5 Einheiten. Ist der Energievorrat bereits kleiner als 5, so ist kein beissen mehr möglich und dies wird auch gesagt. Geben Sie auf der Konsole aus, welcher Dackel beisst. Eine komplette Beisserei mit Zurückbeissen(siehe Fortgeschrittene) ist nicht notwendig.
2. Ergänzen Sie eine Methode `void wedeln()`. Geben Sie auf der Konsole aus, dass der Dackel mit dem Schwanz wedelt.

Erstellen Sie eine neue Klasse `KampfDackel`, die folgendes leistet:

3. Ein `KampfDackel` kann zusätzlich noch zerfleischen. Hier wird angegeben, welcher Dackel zerfleischt werden soll. Diesem sind alle Energiepunkte zu entziehen. Die entsprechende Methode lautet: `void zerfleischen(Dackel opfer)`. Es erfolgt eine geeignete Ausgabe auf der Konsole.
4. Überlagern Sie die Methode `beissen`, indem Sie die Konsolenausgabe etwas heftiger gestalten.



Erstellen Sie eine neue Klasse `SchmuseDackel`, die folgendes leistet:

5. Ein `SchmuseDackel` kann zusätzlich noch schmusen. Es erfolgt eine geeignete Ausgabe auf der Konsole. Einmal Schmusen verbraucht einen Energievorrat.
6. Überlagern Sie die Methode `beissen`, so dass eine `SchmuseDackel` anstatt zu beissen einmal mit dem Schwanz wedelt und einmal schnust.



- Speichere das Programm unter dem Namen **aufgabe\_ls3\_04a** ab.

**Fortgeschrittene Schüler:**

- Führe Sie die Erweiterungen der Anfänger analog aus.
- Beim Zerfleischen geben Sie mit, welcher Dackel zerfleischt werden soll. Auch dies soll letztlich in einer wüsten Beisserei enden. Zerfleischen besitzt einen etwas höheren Zerstörungsgrad des Gegners. Ob die Gegenreaktion des angegriffenen Dackels in zerfleischen oder beissen besteht, soll durch Zufall bestimmt werden, wobei Sie daran denken müssen, dass nur `KampfDackel` zerfleischen können, normale aber nicht.
- Auch `SchmuseDackel` können gebissen oder zerfleischt werden. Aus political-correctness und um die Spielvielfalt zu erhöhen, müssen auch sie sich behaupten können. Ergänzen Sie hierzu eine Methode `beschwichtigen`, die die Kampfeslust des Gegners deutlich abkühlt.
- Der Schwerpunkt Ihrer Überlegungen soll auf einer ausgewogenen Strategie liegen, so dass jeder Dackel eine Chance hat. Damits aber auch Spass macht, soll hin- und wieder einer ins Gras beissen.
- Speichere das Programm unter dem Namen **aufgabe\_ls3\_04b** ab.