

**Präsenzübungen zur Vorlesung**  
**Objektorientierte Programmierung: Wintersemester 2021/2022**  
Nr. 11

---

**Aufgabe 11.1:** Bottleneck

In dieser Aufgabe wollen wir zunächst Getränke in Flaschen abfüllen und diese später generalisieren. Nehmen Sie zunächst an, dass es eine abstrakte Klasse `Beverage`, sowie die Unterklassen `Water`, `Cola`, `Juice` und `Beer` gibt. Es soll nun die Klasse `Bottle` entwickelt werden, die Methoden wie z.B. `break`, `fill` und `empty` besitzt.

- a) Konzipieren Sie einen Entwurf für die Klasse `Bottle`, die Getränke beinhalten kann.
- b) Überlegen Sie nun, wie man von dem Getränk abstrahieren kann, so dass die Flasche einen beliebigen Inhalt haben kann (z.B. Benzin, Sand, Pommes Frites). Überarbeiten Sie Ihr Konzept aus Aufgabenteil a) entsprechend. Dabei sollen Flaschen, die für Getränke gedacht sind nicht mit anderen Inhalten befüllt werden dürfen. Dies gilt auch umgekehrt.
- c) Geben Sie Anweisungen zum Erzeugen einer Flasche Cola sowie einer Flasche Pommes Frites an.

**Aufgabe 11.2:** Tom Bronson

Geben Sie für jede Zeile (ab Zeile 2) in den untenstehenden Code-blöcken an, ob diese übersetzt vom Compiler werden kann, oder nicht. Im Fehlerfall sollten Sie einen Grund für den Fehler nennen.

- a) 

```
1 List<? super Integer> d = new LinkedList<Number>();  
2 Number i = d.get(0);  
3 Object o3 = d.get(0);  
4 d.add(new Integer(5));  
5 Number n = new Long(5);  
6 d.add(n);
```
- b) 

```
1 List<? extends Number> d = new LinkedList<Integer>();  
2 d.add(Integer.valueOf(5));  
3 d.add(null);  
4 Number n = d.get(0);  
5 Integer i = d.get(0);
```