

Programmierung 2 - Sommersemester 2023

Prof. Dr.-Ing. Martin Burger

Übungsblatt Nr. 16 Abgabe KW 19

1. Aufgabe

In dieser Aufgabe soll ein Getränkeautomat mit Hilfe von Generics implementiert werden. Setzen Sie dazu folgendes um:

- (a) Implementieren Sie das in Abbildung 1 dargestellte Klassendiagramm. Fügen Sie Standard-konstruktoren und Konstruktoren, welche Werte für die Felder der Klasse entgegennehmen, hinzu. Erweitern Sie die Klasse außerdem um Getter- und Setter-Methoden. Überschreiben Sie in allen Klassen die toString()-Methode sinnvoll.

Beachten Sie: Die Klassen `Getraenk`, `AlkoholischesGetraenk` und `AlkoholfreiesGetraenk` sind abstrakt.

- (b) Implementieren Sie eine generische Klasse `Flasche`. Der Typ-Parameter soll auf `Getraenk` und alle Subtypen eingeschränkt sein. Die Klasse soll eine private Variable `inhalt` mit generischem Typ besitzen. Neben dem Standardkonstruktor soll die Klasse über die Methoden `fuellen` und `leeren` verfügen. Die Methode `fuellen` nimmt ein passendes `Getraenk`-Objekt entgegen und setzt es als `inhalt`. Die Methode `leeren` leert die Flasche. Überschreiben Sie außerdem die `toString`-Methode so, dass der Flascheninhalt als String ausgegeben wird.

- (c) Implementieren Sie eine generische Klasse `Getraenkeautomat`. Diese definiert ebenfalls einen auf `Getraenk` und allen Subklassen eingeschränkten Typ-Parameter.

Die Klasse besitzt zwei private Felder:

- Eine `ArrayList flaschenlager`. Diese `ArrayList` speichert die Flaschen im Getränkeautomaten. Die `ArrayList` kann Objekte vom Typ des Typ-Parameters und aller Subklassen speichern.
- `int kapazitaet`: Die Anzahl der Flaschen, die der Getränkeautomat aufnehmen kann. Die Kapazität soll über den Konstruktor übergeben und gesetzt werden.

Neben dem Konstruktor soll die Klasse über folgende Methoden verfügen:

- `flascheEinlegen`: Diese Methode nimmt ein Objekt vom Typ `Flasche` entgegen und speichert es, falls noch Kapazität vorhanden ist, im `flaschenlager`. Der Inhalt der übergebenen Flasche muss vom Typ des Typ-Parameters oder einer Subklasse sein. Es sollen nur volle Flaschen in den Automaten eingelegt werden können. Dies muss daher geprüft werden.
- `flascheAusgeben`: Diese Methode gibt eine Flasche aus dem Flaschenlager zurück. Der Getränkeautomat soll nach dem "First in, first out"-Prinzip arbeiten (Ja, ein solches Verhalten ist für einen Getränkeautomaten ungewöhnlich, wird aber hier zur Vereinfachung der Aufgabe so implementiert :)).

Überschreiben Sie auch die `toString()`-Methode so, dass der gesamte Inhalt des Automaten ausgegeben wird.

- (d) Implementieren Sie ein Dialogprogramm zum Testen des Getränkeautomaten. Es soll insb. möglich sein verschiedene Typen von Getränkeautomaten zu instanzieren. Z.B. einen Getränkeautomat, welcher nur Alkoholische Getränke beinhaltet. Oder einen Getränkeautomat, welcher nur Wasserflaschen beinhaltet. Außerdem soll es möglich sein dem Getränkeautomaten volle Flaschen hinzuzufügen und wieder ausgeben zu lassen.
- (e) Bewerten und erklären Sie, welche Vorteile Sie durch die generische Programmierung gewinnen.

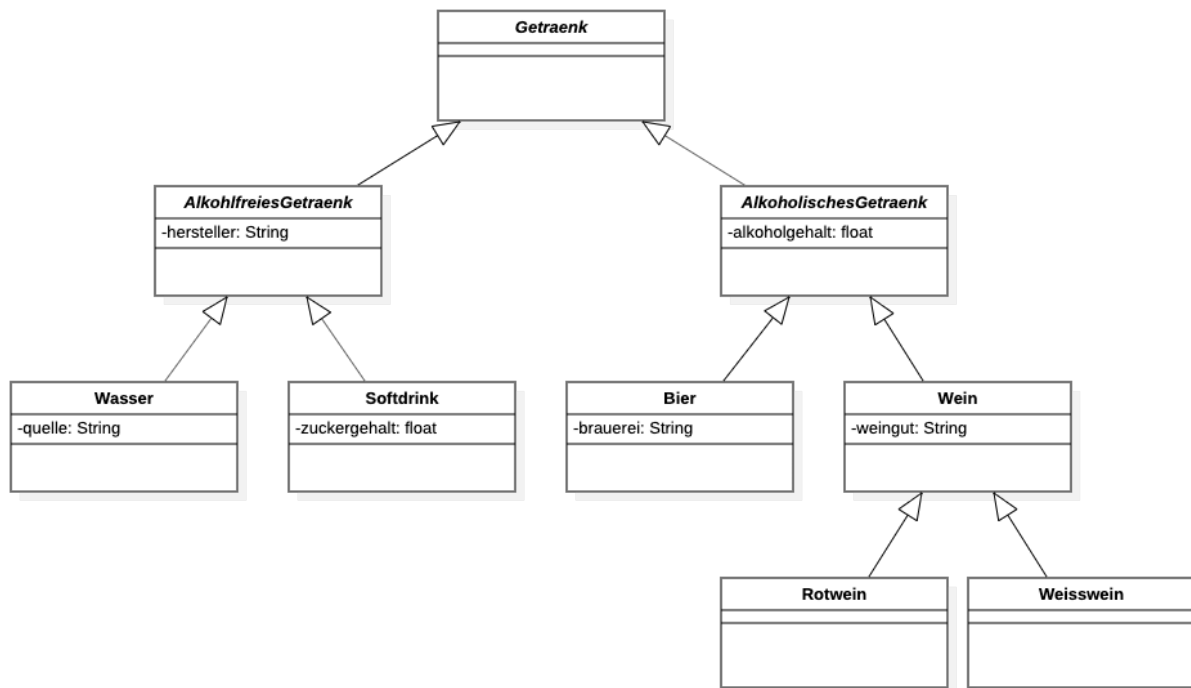


Abbildung 1: Implementieren Sie die in diesem UML-Diagramm dargestellten Klassen.