

Die Kaffeemaschine

Wir programmieren eine simple Kaffeemaschine. Diese ist zusammengesetzt aus Behältern für die Zutaten (Wasser, Kaffee, Kakao, Zucker und Milch) und einem Abfallbehälter.

1 Behaelter

Die abstrakte Klasse `Behaelter` soll den aktuellen Füllstand und den maximalen Inhalt des Behälters in Liter beinhalten. Außerdem soll sie eine abstrakte Methode `public String toString()` beinhalten, die von den Subklassen überschrieben werden müsste.

2 ZutatenBehaelter

Die Klasse `ZutatenBehaelter` ist eine Konkretisierung der Klasse `Behaelter`. Sie soll den Zutaten-Bezeichner und die Methode zum Entnehmen von einer bestimmten Menge beinhalten.

3 Rezept

Die Klasse `Rezept` soll den Namen des Getränks und die dafür notwendige Menge von Wasser, Kaffee, Kakao, Zucker und Milch in Liter beinhalten, z.B.

Name	Wasser	Kaffee	Kakao	Zucker	Milch
Kaffee schwarz:	0.2	0.02			
Kaffee Zucker:	0.2	0.02		0.02	
Kaffee Milch:	0.2	0.02			0.02
Kaffee Milch/Zucker:	0.2	0.02		0.02	0.02
Kakao:	0.2		0.02	0.02	0.02

4 KaffeeMaschine

Die Klasse `KaffeeMaschine` sollte die Hauptklasse Ihrer Anwendung sein. Sie enthält eine `main()`-Methode und kann somit ausgeführt werden. Die Kaffeemaschine gibt in einer endlosen Schleife eine nummerierte Liste der angebotenen Getränke (bzw. vorhandenen Rezepte) aus, von denen der Benutzer eines auswählen kann.

5 Getränk erstellen

In der `getraenkErstellen(Rezept r)` Methode soll nun von Ihnen das gewünschte Getränk erstellt werden. Dabei soll die entsprechenden Menge der Zutaten aus den Behältern entnommen werden.

6 AbfallBehaelter

Bei dem Brühen von Kaffee fällt der Kaffeesatz an, der in einer Klasse `AbfallBehaelter` gesammelt werden soll. Dieser `AbfallBehaelter` soll (wie schon die Klasse `ZutatenBehaelter`) eine Konkretisierung der Klasse `Behaelter` sein. Nach dem Zubereiten von Kaffee soll der Kaffeesatz in diesem Behälter gesammelt werden. Dafür soll die Methode zum Nachfüllen des Behälters implementiert werden.

7 Wartung

Alle Komponenten sollen nun gewartet werden können. Eine Wartung heißt, dass der `AbfallBehaelter` geleert wird, und die `ZutatenBehaelter` nachgefüllt werden.

Definieren Sie dazu ein Interface `Wartbar` mit einer Methode `void wartung()`, das von allen wartbaren Komponenten implementiert werden soll.

Erweitern Sie Ihr Benutzerinterface um das Kommando **"Wartung"**, das die Wartung auslösen soll.

8 Fehlerbehandlung

Erweitern Sie die Kaffeemaschine um eine Fehlerbehandlung mittels Exceptions. Definieren Sie zwei Exception-Klassen:

- `BehaelterVollException`: wird geworfen, falls der `AbfallBehaelter` voll ist.
- `ZutatLeerException`: wird geworfen, falls ein `ZutatenBehaelter` leer ist.

Erweitern Sie die Methode `getraenkErstellen` und die Hauptschleife um geeignete Fehlerbehandlung:

- Eine `BehaelterVollException` soll dazu führen, dass nur noch eine Wartung möglich ist
- Eine `ZutatLeerException` soll eine Warnung an den Benutzer ausgeben, dass seine Wahl zur Zeit nicht möglich ist.

9 Status

Schreiben Sie nach jeder Benutzeraktion den Maschinen-Status (Füllstand der Zutaten und des `AbfallBehaelters`) auf `stdout`.

Verwenden Sie dazu in geeigneter Weise die Methode `public String toString()` aller beteiligten Klassen (`KaffeeMaschine`, `AbfallBehaelter` und `ZutatenBehaelter`).

10 Dokumentation

Dokumentieren Sie die Klassen und Methoden mit sinnvollen Kommentaren.

Rufen Sie `javadoc` auf Ihren Klassen auf und schauen Sie sich die erzeugte Dokumentation an.