

1. Übung zu Softwaretechnik 2

In einem Lager soll der Bestand von verschiedenen Artikeln verwaltet werden. Entwerfen Sie dazu eine Klasse *Lager* und eine Klasse *Artikel*., d.h. im *Lager* werden *Artikel* benutzt.

Fachklasse* Artikel

Entwerfen Sie die Klasse *Artikel* mit folgenden Attributen und Methoden:

Attribute:

int	artikelNr	Artikelnummer
String	bezeichnung	Artikelbezeichnung
int	bestand	aktueller Artikelbestand
double	preis	aktueller Preis

Methoden:

void	bucheZugang(int menge)	Zugang aufbuchen
void	bucheAbgang(int menge)	Abgang abbuchen
String	toString()	Artikel-Objekt als Zeichenkette aufbereiten

Beispielausgabe von toString()

Artikel: 4711 Bezeichnung: Staubsauger Bestand: 100 Preis: 12.34

get-Methoden	zu allen Attributen
set-Methoden	nur zu den Attributen, für die eine set-Methode Sinn macht

Konstruktoren:

- Es werden zwei Konstruktoren für das Konstruieren eines Artikel-Objekts benötigt: einer mit 3 und einer mit 4 Argumenten. Warum macht ein Standardkonstruktor keinen Sinn ?

Folgende Bedingungen sind zu beachten:

- die Artikelnummer muss eine 4-stellige positive Zahl sein; einmal gesetzt darf sie nicht mehr verändert werden;
- die Bezeichnung darf nicht die Null-Referenz sein; leer darf sie auch nicht sein bzw. sie darf nicht allein aus *Whitespace* (Leerzeichen, Tabulatorzeichen, Zeilen-Ende-Zeichen) bestehen;
- der Bestand darf nie kleiner als 0 werden;
- der Preis darf nie kleiner als 0 werden;
- die bei *bucheAbgang* und *bucheZugang* übergebenen Werte müssen auf > 0 überprüft werden, evtl sind dort weitere Prüfungen nötig.

In dieser Übung ist die Plausibilität von übergebenen Werten mit Hilfe von *assert* durchzuführen, um entweder das Erzeugen von Objekten vollständig oder Wertänderungen von einzelnen Attributen eines bestehenden Objekts zu verhindern.

Alle Eingaben, die beim Arbeiten mit der Klasse gebraucht werden, müssen ihren Methoden in Form von Parametern übergeben werden. Die Klasse darf keine Eingaben von der Standard-eingabe lesen, d.h. die Klasse *Stdin* NICHT benutzen !

Fachklasse* Lager

Entwerfen Sie die Klasse *Lager*, die eine vorzugebende maximale Menge an Artikelobjekten mittels Array verwalten kann. Das Array enthält Referenzen auf die Artikel.

Attribute:

`String standort` Standort des Lagers

Weitere evtl. benötigte bzw. sinnvolle Attribute legen Sie selbst fest.

Benötigte Methoden:

- Artikel im Lager anlegen, d.h. Artikel-Objekt erzeugen und Referenz im Array des Lagers ablegen
- Artikel aus dem Lager entfernen, d.h. Referenz auf ein Artikel-Objekt auf *Null* setzen
- Zugang buchen für einen Artikel
- Abgang buchen für einen Artikel
- Preis für alle Artikel um einen bestimmten Prozentsatz verändern (positiv oder negativ möglich, auf sinnvolle Werte beschränken)
- Ausgabe des kompletten Lagerbestands in lesbarer Form

Anmerkungen:

- Das Array ist selbst zu programmieren, d.h. es ist keine von Java vorgefertigte Klasse zugelassen.
- Die Plausibilität von übergebenen Werten ist gegebenenfalls mit *assert* durchzuführen, um Inkonsistenzen in den Objekten zu verhindern.
- Alle Eingaben, die beim Arbeiten mit der Klasse gebraucht werden, müssen ihren Methoden in Form von Parametern mit gegeben werden. Die Klasse darf keine Eingaben von der Standardeingabe lesen, d.h. die Klasse *Stdin* NICHT benutzen!

Grundsätzliche Vorgehensweise

- Erstellen Sie die Klassen vorzugsweise mit dem Entwicklungswerkzeug BlueJ (oder mit einem Werkzeug Ihrer Wahl, wenn Sie es besser können/kennen) und bearbeiten Sie dort die Klassen.
BlueJ ist vorinstalliert, bei einem anderen Entwicklungssystem sind Sie selbst gefordert !
- Zu einem professionellen Quellcode gehören natürlich sinnvolle Kommentare, eine selbst-erklärende Namensgebung für Attribute, Variablen und Konstante (Namen in Binnenmajuskeln bzw. "*lowerCamelCase*"), sowie eine saubere Strukturierung des Programmtextes.
- Kommentieren Sie im Quellcode ihre Klassen, Attribute, Methoden und wichtige Verarbeitungsschritte prägnant und sinnvoll.
- Testen Sie ihre Klasse, d.h. überlegen Sie sich Testfälle für positives und negatives Testen !

* Begriff: *Fachklasse*

Mit einer Fachklasse werden geforderte Funktionalitäten realisiert, die keinen Dialog mit dem Benutzer benötigen. Die variierende Verarbeitung wird durch Übergabeparameter bei den verwendeten Methoden gesteuert.

Als Schnittstelle zum Benutzer dient in dieser Übung das Entwicklungssystem BlueJ mit seinen Bearbeitungsmöglichkeiten. Siehe hierzu auch folgende Einführung

<http://www-stl.htw-saarland.de/doc/bluej.html>

oder die JAVA-Lernhilfen

<http://www-stl.htw-saarland.de/doc/java.html>

Abgabe

Wird noch bekannt gegeben.