

# Übungsblatt 3

Abgabe: 27.04.2009

---

## Aufgabe 1 Manchmal klappt es, manchmal nicht

Bei der Ausführung eines Programms kommt es manchmal zu unvorhergesehenen Ereignissen, vor allem wenn ungeschulte Nutzer die Software mit falschen Eingaben füttern oder im laufenden Betrieb die Festplatte herausnehmen. Auch bei Autohändler Hans-Werner K. ist sowas schon vorgekommen, und er möchte, dass seine neue Software mit solchen Störungen umgehen kann.

### Aufgabe 1.1 Text lesen (60%)

Ab und zu müssen im Autohaus neue Artikel in den Bestand aufgenommen werden. Die neuen Artikel stehen zeilenweise in einer Textdatei. Die Attribute (Artikelnummer, Artikelbezeichnung, Menge, Einkaufspreis) sind durch Kommata getrennt:

```
00453221,Daihatsu Cuore,4,8400.00
00332448,Opel Astra,1,13500.00
```

Schreibt eine Methode, die den Namen einer Textdatei als Eingabe bekommt und für jeden gelesenen Artikel ein Objekt erzeugt, das die Daten aufnimmt. Geht zunächst davon aus, dass die Datei immer korrekt formatiert ist. Das Datenbank-Programm soll alle Artikel in einer (z.B. einfach verketteten) Liste speichern. Achtung: Bei der Leseoperation kann eine *IOException* auftreten. Fangt den Fehler an einer geeigneten Stelle auf und reagiert mit einer sinnvollen Fehlermeldung. Schreibt auch eine Methode, die alle Artikel ausgibt. Ergänzt euer Programm durch eine *main*-Methode, dokumentiert es und testet seine Funktionalität.

**Tip:** Der Zugriff auf eine Textdatei gelingt durch die Java-Klassen *FileReader* und *BufferedReader*. Eine Textzeile lässt sich durch die String-Methode *split(",")* in ein Array aus Strings zerlegen. Durch *Integer.parseInt()* kann ein String in eine Zahl umgewandelt werden.

### Aufgabe 1.2 Noch mehr Ausnahmen (40%)

Auch beim Einlesen einer Datenzeile kann einiges schief gehen. Zum Beispiel stimmt die Anzahl der Attribute nicht, oder statt der erwarteten Artikelnummer tauchen Buchstaben auf. Schreibt eine Methode für die Umwandlung einer Textzeile in ein Objekt. Prüft darin die Syntax der gelesenen Zeile und definiert mindestens zwei eigene Exceptions, die auf konkrete Fehler in der Formatierung hinweisen. Fangt die Exceptions während des Einlesens ab, so dass das Programm das Lesen in der nächsten Zeile fortsetzen kann. Testet das erweiterte Programm auch mit geeigneten fehlerhaften Textdateien.