

## 9. Übungsaufgabe

### Themen:

Entwurfsmuster, Untertypbeziehungen, gesamter Stoff

### Termine:

Ausgabe: 12.12.2012  
reguläre Abgabe: 19.12.2012, 12:00 Uhr  
nachträgliche Abgabe: 09.01.2013, 12:00 Uhr

### Abgabeverzeichnis:

Gruppe/Aufgabe9

### Programmaufruf:

java Test

### Grundlage:

[Skriptum](#), Schwerpunkt auf den Abschnitten 5.1, 5.2.1 und 5.2.2

## Aufgabe

### Welche Aufgabe zu lösen ist:

In einer Bäckerei werden vor Weihnachten Kekse gebacken. Jedes Keks hat eine bestimmte Form (rund, in der Form eines Mondes und in der Form eines Weihnachtsmannes) und besteht aus einer bestimmten Teigart (Mürbteig, Zimtsternteig oder Schokoladenteig). Neben einfachen Keksen gibt es Doppelkekse, die aus zwei einfachen Keksen und einer Füllung (Schokolade oder Marmelade) bestehen. Es gibt Keksbackmaschinen, die nur Kekse mit einer bestimmten Form, aber mit unterschiedlichen Teigarten backen können. Weiters gibt es eine Backmaschine für Doppelkekse. Diese Maschine wird mit einem Keks beladen, erkennt automatisch die Teigart und Form dieses Kekses, bäckt ein äquivalentes Keks und erzeugt daraus ein Doppelkeks, das mit einer bestimmten Füllung gefüllt ist. Kunden geben Bestellungen an die Bäckerei ab.

Eine Bestellung ist eine Liste von Positionen, die jeweils die Anzahl, Form, Teigart und Fülle (wenn Doppelkekse gewünscht sind, sonst keine Fülle) angeben. Als Ergebnis erhält man eine Keksdose, die mit allen entsprechenden Keksen gefüllt ist.

Schreiben Sie ein Javaprogramm, das die Bäckerei simuliert. Entwickeln Sie Klassen(hierarchien) um Kekse, Keksbackmaschinen (für einfache Kekse und Doppelkekse), Bestellungen und Keksdosen darzustellen.

Folgende Methoden und Funktionalitäten sollen unterstützt werden:

- Eine Methode *inhalt* für eine Keksdose, die Beschreibungen der in der Dose enthaltenen Kekse auf die Standardausgabe ausgibt.
- Eine Methode *drucke* für eine Bestellung, die alle Positionen in der Standardausgabe auflistet.

Schreiben Sie eine Klasse Test, die mehrere Bestellungen erstellt, die Bestellungen auf die Standardausgabe ausgibt, entsprechende Keksdosen erzeugt und befüllt, und die Inhalte der Keksdosen anzeigt.

### **Warum die Aufgabe diese Form hat:**

Diese Aufgabe enthält Details, die den Einsatz bestimmter Entwurfsmuster nahelegen. Durch mehrere Arten von Backmaschinen, Keksformen, Füllungen, etc. ergibt sich eine umfangreiche Typstruktur, die auch parallele Teilstrukturen enthält. Alle Schwierigkeiten in der Aufgabe sollten mit bekannten Mitteln leicht zu lösen sein. Das gibt Ihnen zum Abschluss der Laborübung Gelegenheit zu zeigen, dass Sie schöne, einfache und gut wartbare Programme schreiben können.

### **Wie die Aufgabe zu lösen ist:**

Es ist wichtig, alle Untertypbeziehungen zu finden und die Ersetzbarkeit auf geeignete Weise (durch durchdachte Zusicherungen, aber auch durch Testfälle) sicherzustellen. Vererbungsbeziehungen, die nicht gleichzeitig auch Untertypbeziehungen sind, müssen unbedingt vermieden werden. Andererseits sollen überall Untertypbeziehungen hergestellt werden, wo das möglich ist. Überlegen Sie sich, wo Sie (abstrakte) Klassen und Interfaces verwenden. Beachten Sie die Sichtbarkeit. Benutzen Sie Generizität und vorgefertigte Containerklassen, wo dies sinnvoll erscheint.

Diese Aufgabe enthält wahrscheinlich weniger inhaltliche Schwierigkeiten als vorangegangene Aufgaben. Gerade deswegen sollten Sie ein schönes, einfaches und gut wartbares Programm abliefern, da solche Qualitätskriterien stärker in die Beurteilung einfließen als gewohnt.

### **Was im Hinblick auf die Beurteilung zu beachten ist:**

Schwerpunkte bei der Beurteilung liegen auf Untertypbeziehungen und Entwurfsmustern. Kräftige Punkteabzüge gibt es für

- Verletzungen des Ersetzbarkeitsprinzips bei Verwendung von Vererbungsbeziehungen,
- den Verzicht auf mögliche Untertypbeziehungen,
- fehlende, falsche, nichtssagende und unverständliche Zusicherungen,
- unnötigen Code und mehrfache Vorkommen gleicher oder ähnlicher Code-Stücke,
- komplizierte und schwer verständliche Programmstrukturen, die sich leicht aus nicht oder falsch eingesetzten Entwurfsmustern ergeben,
- unpassende Sichtbarkeit.

**Was im Hinblick auf die Abgabe zu beachten ist:**

Verzichten Sie auch bei der Lösung dieser Aufgabe auf die Verwendung von packages und Verzeichnissen innerhalb des Abgabeverzeichnisses *Gruppe/Aufgabe9*. Schreiben Sie (abgesehen von geschachtelten Klassen) nicht mehr als eine Klasse in jede Datei.