

## Übung 7: Design Pattern

### Aufgabe A: Design Pattern

Nennen Sie für die in der Vorlesung vorgestellten Entwurfsmuster (Design Pattern) **Template Method**, **Strategy**, **Bridge** und **Decorator** jeweils ein weiteres Beispiel. Skizzieren Sie es mit einem UML-Klassendiagramm und erläutern Sie warum das Entwurfsmuster in diesem Beispiel sinnvoll ist.

*Suchen, wenn möglich, in eigenen, bzw. Ihnen vertrauten Projekten nach den Beispielen.*

### Aufgabe B: Flexibilität und Wiederverwendung

Das Zugbeispiel wird um eine Klasse erweitert, welche direkt fertige Züge mit Waggon erstellt:

```
public class PersonenZug {
    Lokomotive erstelleZug() {
        Lokomotive lok = new Lokomotive(80000);
        for (int i = 0; i < 10; i++) {
            Personenwaggon p = new Personenwaggon(2000, 3);
            lok.ankopplelnPerso(p);
        }
        return lok;
    }
}
```

Anschließend soll das Erstellen von Zügen flexibler gestaltet werden können, so dass auch reine Güterzüge oder Züge mit Kippwaggon erstellt werden können. Zudem soll auch die Waggonanzahl variiert werden können.

Ändern Sie die Implementierung um diese Funktionalität auch anzubieten. Welche Design Pattern können helfen?

### Aufgabe C: Unabhängige Erweiterungen

Das Zug-Beispiel soll ein weiteres Mal erweitert werden. Diesmal soll es für die Güterwaggon vier Erweiterungsmöglichkeiten geben.

- Güterwaggon können auch als „bewachte“ Versionen zur Verfügung stehen. Wenn ein Waggon bewacht ist, ist er um 1000 kg schwerer.
- Des Weiteren können die Waren einzeln verpackt eingeladen werden. Dadurch erhöht sich das Gewicht (beim Einladen) um 10 kg pro Gut.
- Der Waggon kann versichert werden. Für die Versicherung sind pro geladenem Gut 10 EUR Prämie fällig.

Alle diese Erweiterungen sollen sich auch mit der Erweiterung als Kühlwaggon (Übung 4.C) kombinieren lassen. Alle drei Erweiterungen sollen unabhängig kombiniert werden können, z.B. sowohl ein bewachter, versicherter Güterwaggon, ein bewachter, verpackter Kühlwaggon, wie auch ein Güterwaggon ohne Erweiterungen sollen möglich sein.

Diskutieren Sie verschiedene Möglichkeiten diese Erweiterungen zu Implementieren (Nutzen Sie UML-Notation zur Illustration). Welche Grenzen gibt es bei der Vererbung? Welche Design Pattern können helfen? Welche Nachteile hat die Lösung mit Design-Pattern?

### **Aufgabe D: Observer Pattern**

Neben den vier in der Vorlesung vorgestellten Design Pattern gibt es viele weitere. Recherchieren Sie das Observer Pattern und erläutern Sie dessen Motivation und Funktionsweise am Beispiel des Lok- oder Bank-Szenarios.

Kennen Sie Beispiele in denen das Observer Pattern benutzt wird?

*Ihre Lösung dieses Übungszettels geben Sie bitte wie gewohnt bis zum 02.12.2012 23:59 Uhr per Email an [selecture@mathematik.uni-marburg.de](mailto:selecture@mathematik.uni-marburg.de) ab.*