

## U07 Entwurfsmuster (I)

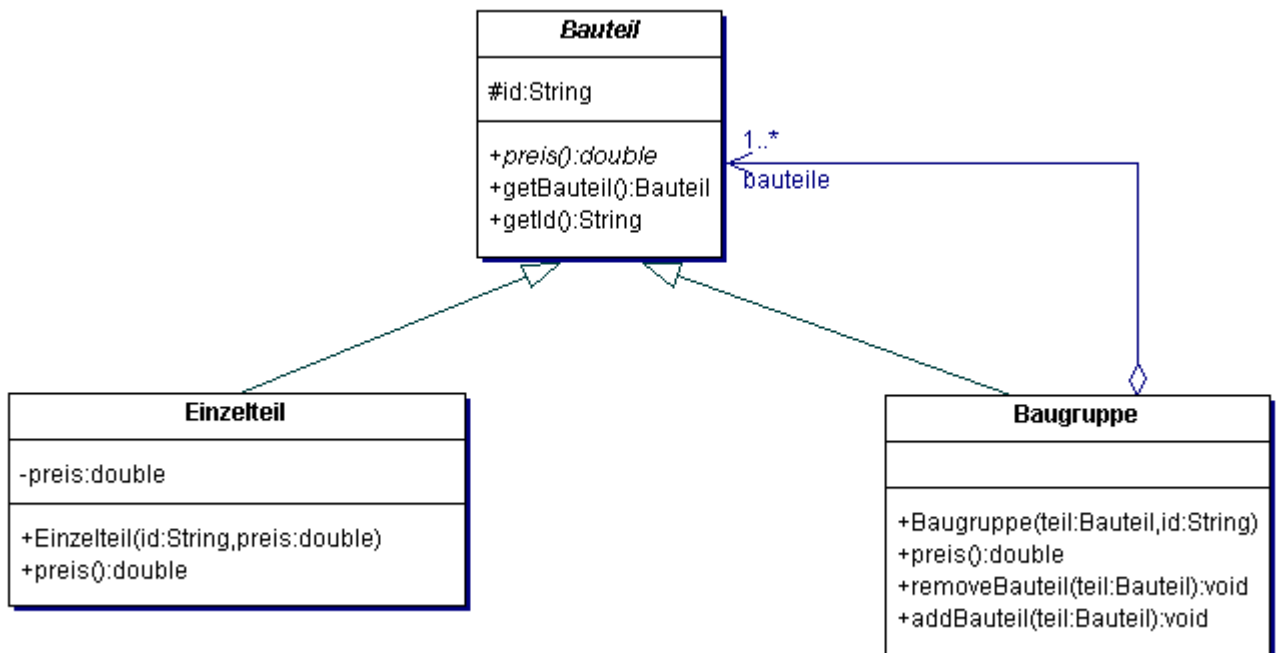
### Inhalt der Übung

- Entwurfsmuster und Implementierung
- Datenstrukturen
- Ausnahmebehandlung

### Übungsaufgaben

#### Aufgabe 1 (Bauteil)

Gegeben ist das nachfolgende UML-Klassendiagramm für eine einfache Stücklistenverwaltung. In diesem Modell bestehen Bauteile entweder aus Einzelteilen oder Baugruppen. Der Preis einer Baugruppe ergibt sich aus der Summe der Preise seiner Bestandteile.



- Welches aus der Vorlesung bekannte Entwurfsmuster wurde in diesem Modell verwendet?
- Zeichnen Sie dieses Entwurfsmuster und die Rollen der daran beteiligten Klassen in das obige Klassendiagramm ein!
- Ergänzen Sie die gegebene Implementierung des Modells (*Stueckliste.zip*), in dem Sie die Klasse *Baugruppe* codieren! Verwenden Sie zur Implementierung das Java-Collection Framework. Beachten Sie, dass die Aggregation zwischen *Baugruppe* und *Bauteil* (Attribut *bauteile*) standardmäßig als `java.util.Set` implementiert wird.

➤ Diskutieren Sie den Entwurf und refaktorisieren Sie ggf. die Klasse Baugruppe!

▪ **Variante 1:**

- a. Diskutieren Sie die `removeBauteil()`-Methode!
- b. Würden Sie weitere Methoden refaktorisieren?

▪ **Variante 2:** Überlegen Sie sich anhand des folgenden Beispiels was passiert, wenn Sie die Aggregation zwischen Baugruppe und Bauteil als Liste implementieren!

```
Einzelteil t1 = new Einzelteil("E001", 2.30);
Einzelteil t2 = new Einzelteil("E002", 4.70);
Baugruppe t3 = new Baugruppe(t1, "B001");
Baugruppe t4 = new Baugruppe(t2, "B002");
t3.addBauteil(t4);
t3.addBauteil(t2);
t3.addBauteil(t4);
```

## Aufgabe 2 (MyCollection)

Gegeben ist die Datei `Bestellung.java`. Dieses Java-Programm arbeitet mit einer selbst definierten Implementierung (`MyCollection`) von `java.util.Collection`. Die Klasse `MyCollection` implementiert die Methode `iterator()` entsprechend dem Iterator-Pattern.

- Veranschaulichen Sie sich das Programm mit Hilfe eines UML-Diagramms.
- Machen Sie sich die Arbeitsweise des implementierungsspezifischen Iterators klar und zeichnen Sie das Iterator-Pattern ein!
- Reimplementieren Sie `MyCollection` als generische Datenstruktur und stellen Sie diese Datenstruktur als eigenständige Klasse (`MyCollection.java`) zur Wiederverwendung für andere Beispiele bereit!

## Empfehlung zum Selbststudium

1. Implementieren Sie die INLOOP-Exam-Aufgaben **Pricing, PredicateIterator, Renovation Project, Part Management, Desktop Search Engine, Project Management** und **Payroll**! Überlegen Sie, welche Entwurfsmuster in den Programmen enthalten sind!
2. Lösen Sie **Aufgabe 3 der Klausur SS 2009** ([1]). Diese Aufgabe erweitert Aufgabe 1 dieser Übung (Bauteil)!

## Referenzen

[1] <ftp://ftp.ifsr.de/klausuren/SWT/SWT-2009-07-%23%23.pdf>

## Literaturempfehlung

Softwaretechnologie für Einsteiger. PEARSON Studium, 2. Geänderte Auflage, 2014

- Rollenmodellierung: S. 101-103
- Entwurfsmuster: S. 199-212, insbesondere S. 210-211