

LU07.L03 - Refactoring oder nicht?

Nr.	Einordnung	Begründung
1	Refactoring	Verhalten identisch, nur der Name ändert. Mit dem Rename-Werkzeug der IDE risikoarm
2	Bugfix	Das Verhalten ändert sich absichtlich: Bestellungen über genau 100 bekommen neu Rabatt
3	Optimierung	Ergebnis bleibt gleich, nur die Laufzeit ändert. Achtung: Reihenfolge und Duplikate ändern sich - falls davon etwas abhängt, ist es doch eine Verhaltensänderung
4	Refactoring	Dieselbe Response, andere Struktur. Der klassische Fall für D1I
5	Neues Feature	Die Ausgabe enthält etwas, das vorher nicht da war
6	Refactoring	Gleiche Liste als Ergebnis, deklarativer geschrieben
7	Rewrite	Die Implementierung wird ersetzt. Verhalten meist ähnlich, aber nicht garantiert gleich - hier braucht es Characterization Tests <i>vorher</i>
8	Grenzfall	Für den Aufrufer, der den Rückgabewert nutzt, ändert sich nichts. Für den Aufrufer, der sich auf die mutierte Eingabeliste verlassen hat, ändert sich alles

Zu Fall 8

Die Umstellung von mutable auf immutable ist genau dann ein reines Refactoring, wenn **keine** Aufrufstelle die Nebenwirkung genutzt hat. Prüfen lässt sich das nur, indem man alle Aufrufstellen anschaut - oder indem Tests existieren, die die Nebenwirkung abdecken.

```
# vorher: der Aufrufer sieht die Änderung, ohne den Rückgabewert zu
verwenden
def preise_erhoehen(produkte, prozent):
    for p in produkte:
        p["preis"] = round(p["preis"] * (1 + prozent / 100), 2)
    return produkte

# nachher: wer den Rückgabewert ignoriert, bekommt jetzt gar nichts mehr
def preise_erhoehen(produkte, prozent):
    return [{**p, "preis": round(p["preis"] * (1 + prozent / 100), 2)} for p
in produkte]
```

Der Umbau ist trotzdem richtig - er muss nur als Verhaltensänderung behandelt werden: Aufrufstellen suchen, anpassen, testen, in einem eigenen Commit.

Sicherheitsnetz je Fall

- 1, 4, 6: bestehende Tests genügen, IDE-Werkzeug verwenden
- 2, 5: neuer Test für das neue Verhalten, alter Test wird angepasst
- 3: assert auf gleiches Ergebnis plus Messung vorher/nachher
- 7, 8: Characterization Tests vor der Änderung ([LU07e](#))

M323-LU07, M323-D1B

 © Kevin Maurizi

From:

<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:

<https://wiki.bzz.ch/modul/m323/learningunits/lu07/loesungen/einordnen>

Last update: **2026/09/09 11:13**

