

Nachdenkzettel: Interfaces und Software-Architektur

Firaz Ilhan, Jens Schlegel, Maximilian Dolbaum

1. Spezifizieren Sie das Interface „Stecker“ für diese Implementation.



copyright Aunkrig, CC-BY-SA-4.0

- Wo sich die Stecker befinden.
- Die Form/Größe
- Schutzkontakt
- Spannung

2. Ist das

a) eine korrekte Ableitung von der obigen Implementation?

- Form und Verhalten sollten gleich sein.
- Ähnliches Interface, nur ein wenig anders.

b) eine korrekte Implementation Ihres Interfaces

- Stecker passen nicht ineinander!!



copyright hic et nunc, Cc-by-sa-3.0-migrated

3. Und das? Autor: somnusde, wikimedia-commons, PD

Keine korrekte Ableitung des Schukostecker



4. Wie sieht es mit 220 V aus? Interface oder Implementation? Und das Material des Schukosteckers?

- 220 V = Interface
- Material = Implementation

5. Wieviel Spass hätten wir ohne die DIN Norm für Schukostecker oder Eurostecker?

- Ohne DIN Normen hätten wir eine sehr schlechte Übersicht

6. Was gehört alles zum „Interface einer Klasse“ in Java? (Anders formuliert für UX-Leute: wenn ich von jemandem eine Klasse in meinem Code benutze: was ärgert mich, wenn es geändert wird?)

- Name der Methode
- Rückgabotyp
- Parameter der Methode

7. „Class B implements X“. Jetzt fügen Sie eine neue Methode in Interface X ein. Was passiert?

- Wir erhalten einen Fehler, da die Methode in der Klasse B implementiert werden muss

8. Zwei Interfaces sind nicht voneinander abgeleitet, haben aber zufällig die gleiche Methode. Können Sie Implementationen dieser Interfaces polymorph behandeln?

```
Interface X {                                Interface Y {                                class B implements Y
{ ...}
    public void foo();                        public void foo();
}                                              }
```

```
X x = new B(); ??
x.foo();        ??
```

Nein, es geht nicht, da es sich um verschiedene Interfaces handelt

9. Ihr code enthält folgendes statement: `X xvar = new X();`
Was ist daran problematisch, wenn Sie eine Applikation für verschiedene Branchen/Kunden/Fälle bauen?

- Für Verschiedene Anwendungen ist es sinnvoller abgeleitete Klassen zu benutzen