

Nachdenkzettel: Interfaces und Software-Architektur

.....
Vorname, Name, Matrikelnummer

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



```
public interface Plug {  
    void onPlugIn(Voltage voltage, Gound ground);  
    void onPlugOut();  
}
```

copyright Aunkrig, [CC-BY-SA-4.0](https://creativecommons.org/licenses/by-sa/4.0/)

2. Ist das a) eine korrekte Ableitung von der obigen Implementation?
b) eine korrekte Implementation Ihres Interfaces



Yes, it is.

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



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

No, because the Ground Pin is missing.

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

The interface would work perfectly, nevertheless we would recommend to implement it.

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

Hell of a lot! :)

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?)

-
- All of this has little to do with how I utilize a class as an external.
- These will mostly be methods, presuming that the programmer of the class I wish to utilize follows basic java principles like encapsulation.
-

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

The new method must be implemented by Class B. Otherwise, a compiler error will occur.

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(); ??

No

9. Ihr code enthält folgendes statement: X xvar = new X();

Was ist daran problematisch, wenn Sie eine Applikation für verschiedene Branchen/Kunden/Fälle bauen?

If X is a class that implements an interface, use the interface type for variables so that implementations may be changed out easily.

10. Von ArrayList ableiten oder eigene Klasse „Catalog“ oder ähnlich bauen und ArrayList<> verwenden? Sprich: soll man von Java Basisklassen ableiten? Beispiele: Vegetable, VegetableCatalog Task, TaskList, GameObject, GameObjectList etc.

In order to get what you want, rather of inheriting from base classes, you should implement the same corresponding interfaces.