

# Nachdenkzettel: Interfaces und Software-Architektur

.....  
Vorname, Name, Matrikelnummer

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



Interface Plug:  
Material  
PinLayout  
CurrentLimit  
VoltageLimit

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



a) No, because the plug from above doesn't fit into the Outlet because of the different pin layout. Also the plug will not be connected to ground.

b) Yes, as it does contain all the attributes, just not compatible ones to the above mentioned plug.

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



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

It does fit into the outlet, but ground won't be connected, so it doesn't fulfill all the criteria.

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

Implementation, as the material defines the voltage limit.

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

Zero, as there would be countless variations of different plugs and outlets and a ton of compatibility issues.

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?)

- Methods with methods return type, name, parameter, without method body
- declared exceptions
- package
- class name
- 

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

New method in interface X has to be implemented in Class B

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();      ??
```

Yes, as they share the same name and return type.

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

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

You have to change the code for `X()` for each customer, that wants a different object

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.

That would interfere with the encapsulation and dependencies of the classes.