Nachdenkzettel: Interfaces und Software-Architektur

Aufgabe 1

The interface “plug” is defined by its length and width (shape) and the depth of its “bolts”. Also, the grounding is very important. (DIN - Norm)

Aufgabe 2

a) Not a correct extension of the above implementation, because the grounding doesnt work as expected

b) Obviously not, because the grounding isn’t correct implemented

Aufgabe 3

a) Matches the interface but the implementation is different

b) We can willful break the interface if its safe to use a plug without grounding (e.g. socket and plug made out of plastic ⇒ no grounding problem)

Aufgabe 4

The voltage is an interface, because the 220V define the used material for the plug.

Aufgabe 5

We wouldn't have fun, because every manufacturer would have different solutions for their devices.

Aufgabe 6

- Methods
- Attributs
- name of interface

Aufgabe 7

You would get an error in Class B till you implement the new method.

Aufgabe 8

No. If we have two objects, a car and a house and they both have the method closeDoor(), we can’t treat them the same way, because they are completely different objects.

Aufgabe 9

Poor naming → Later hard to identify what's the variable even for, It´s not extendable etc. You should use it as an interface.

Aufgabe 10

The normal objections to extending a class is the favor composition over inheritance discussion. Extension isn't always the preferred mechanism, but it depends on what you're actually doing.