Nachdenkzettel: Interfaces und Software-Architektur

.....

Bianca Knülle, 43373/ Despoina Sfantou, 43368/ Julia Fellmeth, 43391/ Kethella Oliveira, 43385

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

One could observe the differences between certain sockets in order to define interfaces. For example the ones in the photo. The first socket has a white interface and the main difference between the two sockets is the fact that the one provides, whereas the other forwards the power.
Furthermore, voltage, material and the shape of a socket are also important components of an interface.

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

In the depicted socket, there is a different implementation as the first one. Different material is being used and the socket in this example is located on the wall.
All in all, this implementation is not preferable, since not all chargers are suitable with this kind of sockets due to the third output / hole.

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

The shape is different, it can, however, be easily be put in the same socket.

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

By different Voltage, there also come different interfaces.
Concerning the material, the implementation differs; though the implementation does not matter as long as the interface is the same.

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

Common interfaces are of paramount importance. It would be very ineffective, if there were several different sockets for each device and each brand. (More costs, more material, more plastic, heavier to carry around).

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?)

- One would be upset if the main method of a code was altered. This would mean that the other code or classes would be changed as well. In conclusion, changing the main method could affect a whole project and thus cause errors.
-
-
-
-

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

By adding a new method in the Interface X class, a compilation error is caused. One possible solution would be to use a default method,

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();            ??
```

—> Class B implements Y, X{…}
The interfaces both implement the same method, which causes an error.
Class B can Override the method.

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

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

Technically there is nothing wrong.
The interface variables, however, are a better option, since they are flexible and easier to use. The latter can be, hence, easier to alter.

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.

The best option would be to create a list and an abstract list.
For example, we have Task and TaskList. This would look like this:
TaskList extends AbstractLIst<Task>
Through this option, fewer methods are implemented as e.g. List<Task>