

The Interface contains:

- a) static methods
- b) method signatures with an implementation
- c) default methods
- d) constants

The correct method example in a Interface looks like:

- a) public static void name();
- b) public void name() {methods body};
- c) void name() {methods body};
- d) public static void name();
- e) static void name();

Why do we use Interfaces?

- a) to get better performance of a system
- b) to get to originally inkompatibel system compatibel
- c) to reduce error-prone

Which sentence is false:

- a) Every class has one directsuperclass
- b) Not every class implement an Object class
- c) An other name of subclass is extended class or derived class.
- d) Multiple inheritance is not allowed.

How we can create an Object in a subclass

- a) we can initialize an Instance and give them the parameter manually
- b) We have to use an Constructor von the superclass
- c) we can use a Constructor von the superclass
- d) It doesn't matter how the Instance of subclass become the attributes.

What is a correct implementation of instanceof method?

a)

```
if (obj instanceof MountainBike) {  
    MountainBike myBike = obj;  
}
```

b)

```
if (obj instanceof MountainBike) {  
    MountainBike myBike = (MountainBike)obj;  
}
```

c)

```
if (obj instanceof MountainBike) {  
    MountainBike myBike = (MountainBike)obj;  
}
```