SWE 1 - **Konstruktoren** (1)

```java
public class Person {
  String name;
  int age;

  public Person() {
    // assign default values to all member variables
    name = "";
    age = 0;
  } // end default constructor

  public Person(String name, int age) {
    this.name = name;     // init name w/ given value
    this.age = age;
  } // end constructor
    ...                      // further methods
} // end Person
```

SWE 1 - **Konstruktoren** (2)

```java
  ...
  public Person(Person p) {
    this();
    if (p != null) {
      // copy all values from p to this object
      name = p.name;
      age = p.age;
    } else {
      this();
    } // end if
  } // end constructor
  ...
```

SWE 1 - **Strings** (1)

```
String name = null;     // var w/o ref. to string-object

name = "Otto";
String name2 = name;
writeLn(name2 == name); // => true, it's the same object
                        // here we compare objects!
name2 = new String("Otto");
writeLn(name2 == name); // => false, not the same object
                        //           only same string
writeLn(name2.equals(name)); // => true, same string
                               // here we compare strings!

writeLn(name2.length());// => 4, "Otto" has 4 characters
```
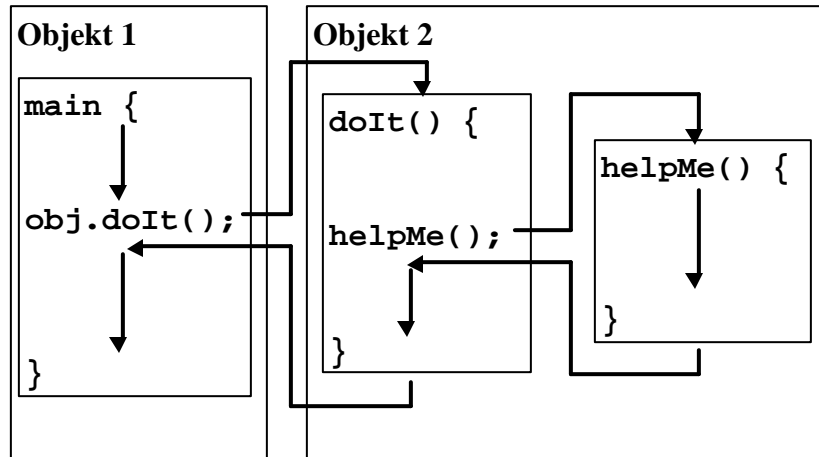
SWE 1 - **Strings** (2)

```
String a = "strange", b = "strange,
      c = new String("strange"),
      d = new String("strange");
writeLn((a==b) + " | " + (c==d));
```

➔ Ausgabe: true | false

SWE 1 - **Kontrollfluß bei Methodenaufruf**


```
Objekt 1              Objekt 2

main {                doIt() {              helpMe() {

obj.doIt();           helpMe();

}                     }                     }
```

SWE 1 - **Konrollfluß bei Methodenaufrufen** (2)

```
class Person {         class Car {            class Engine {
  ...                    Engine eng;           ...
  void start(Car c) {    ...                   void start() {
    c.start();           void start() {          ...
  }                        eng.start();        }
  ...                    }                     ...
}                        ...                 }
                       }
```

```
public static void main (String[] args) {
  Person p = new Person();
  Car c = new Car();
  p.start(c);
 }
```