

27 May 2023

// Static Variable

static
count = 0 X 2

SVE

a
incr(cnt)

o1

o2

a = 10
incr(cnt)

a = 20
incr(cnt)

```
J Main.java 3 x
J Main.java > Main > main(String[])
1 // Static Variable in Java
2 class StaticVariableExample
3 {
4     static int count = 0;
5     int a;
6     void incrementCounter()
7     {
8         count++;
9     }
10 }
11 public class Main {
12     Run | Debug
13     public static void main(String[] args)
14     {
15         StaticVariableExample o1 = new StaticVariableExample();
16         o1.a = 10;
17         System.out.println("Counter value: "+o1.count);
18         System.out.println("a: "+o1.a);
19         o1.incrementCounter();
20         StaticVariableExample o2 = new StaticVariableExample();
21         o2.a = 20;
22         System.out.println("Counter value: "+o2.count);
23         System.out.println("a: "+o2.a);
24         o2.incrementCounter();
25         System.out.println("Counter value: "+o1.count);
26     }
27 }
```

PROBLEMS 3 OUTPUT DEBUG CONSOLE TERMINAL

Counter value: 0 ✓
a: 10 ✓
Counter value: 1 ✓
a: 20 ✓
Counter value: 2 ✓

(base) ingledarshan@192 NS 27 May 2023 %

0 3 Live Share

Ln 19, Col 63 Spaces: 4 UTF-8 LF {} Java

Method Overloading

In a same class, defining 2 or more methods with the same name is called Method Overloading.

A class may have as many methods with the same name based on the following three rules. Atleast one of the following rules must be true for successful overloading of method.

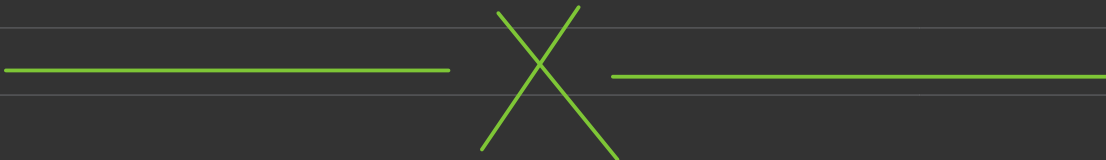
1. It should have different number of parameters.
2. If the number of parameters are same, the datatype of parameters should be different.
3. If the datatypes are also the same, at least the sequence of parameters should be different, otherwise it is a compilation error.

Note:

- a. If only the return type of a method is different, it is not the case of overloading. it is a compilation error.
- b. Overloading works on parameters. Return type is not the criteria to overload the function.

Constructor Overloading

same as Method Overloading, but here we work with constructors.



Static Method:

If a method is declare as static, i can be accessed without creating the object of the class, just by calling:

`ClassName.methodName()` syntax

but if we try to call static method of the same class, then even the classname is not required.

Also, the static method can access only static instance variable. It cannot access non-static instance variable which are local to the function.