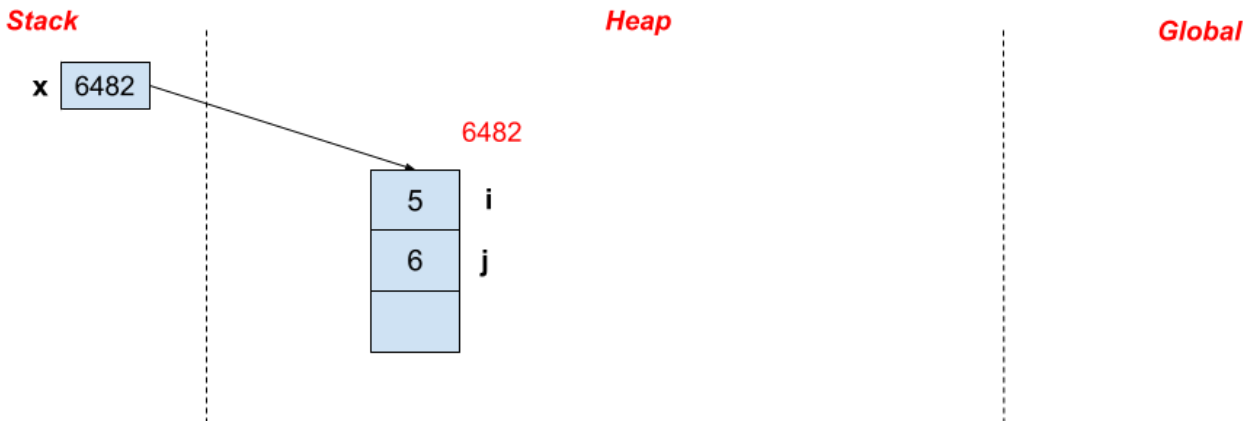
Dmitri Stanchevici
Unit 6
Module 3

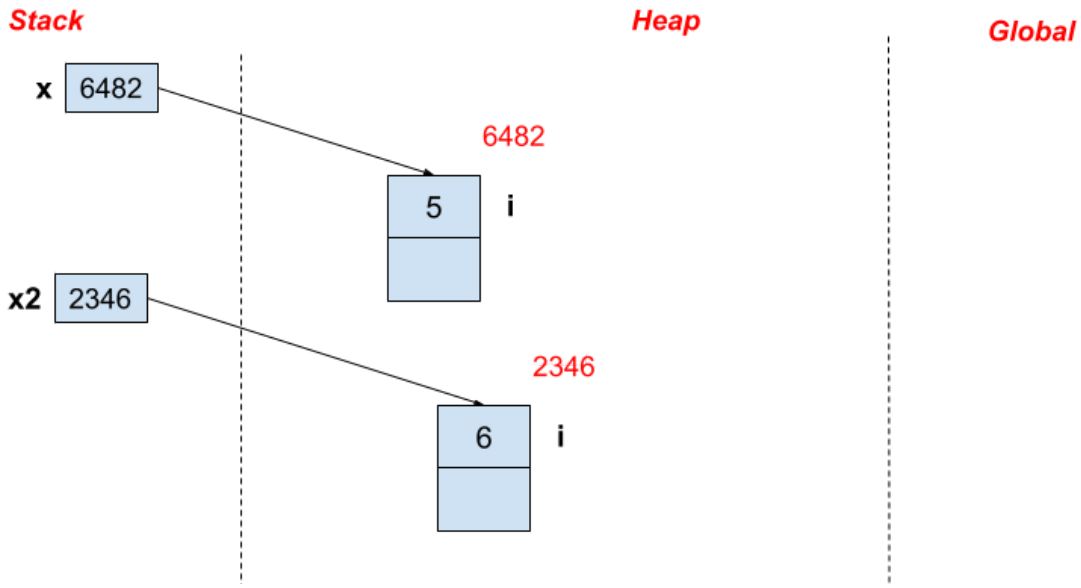
Ex. 3.3

Memory picture of at the end of **main()** in **MyDynamicExample**:



Ex. 3.4

Memory picture after the execution of **x2.i = 6** in **DynamicExample4**:

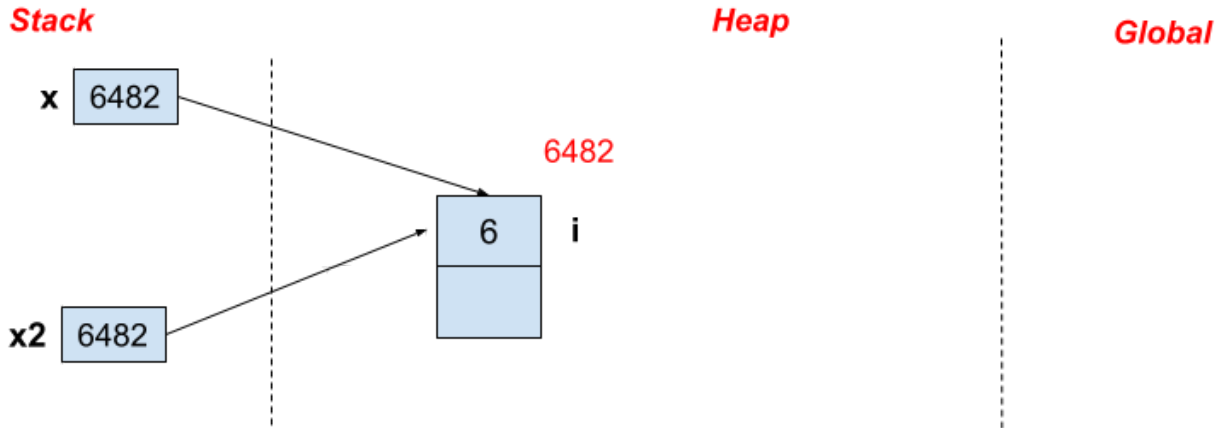


Ex. 3.5

The 3rd and 4th iterations of the loop print:

i=0
i=0

Ex. 3.6



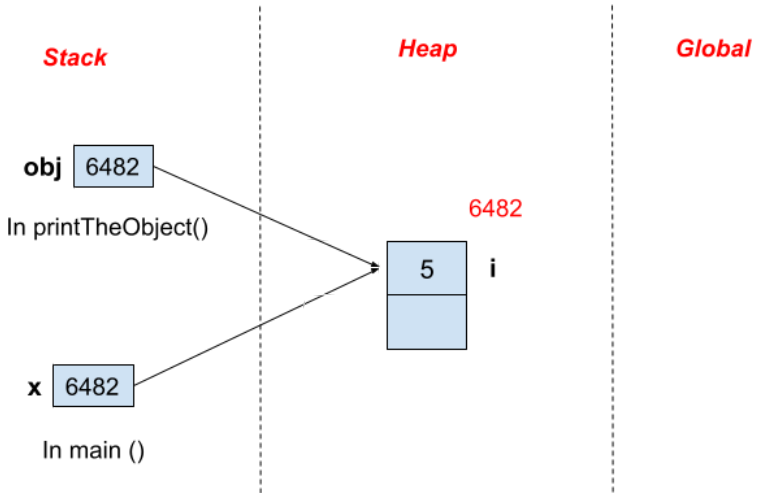
Printed out is

i=6
i=6

When **ObjX6 x2** is declared, it is assigned the value of **x,** and this value is the address of the object instance to which x is pointing. In other words, x2 is made to point to the same object instance as the one pointed to by x. Because x and x2 point to the same object instance on the heap, changes to this instance's variables can be made through either of these two variables. Thus stating x2.i = 6 means that x.i has also changed to 6 now.

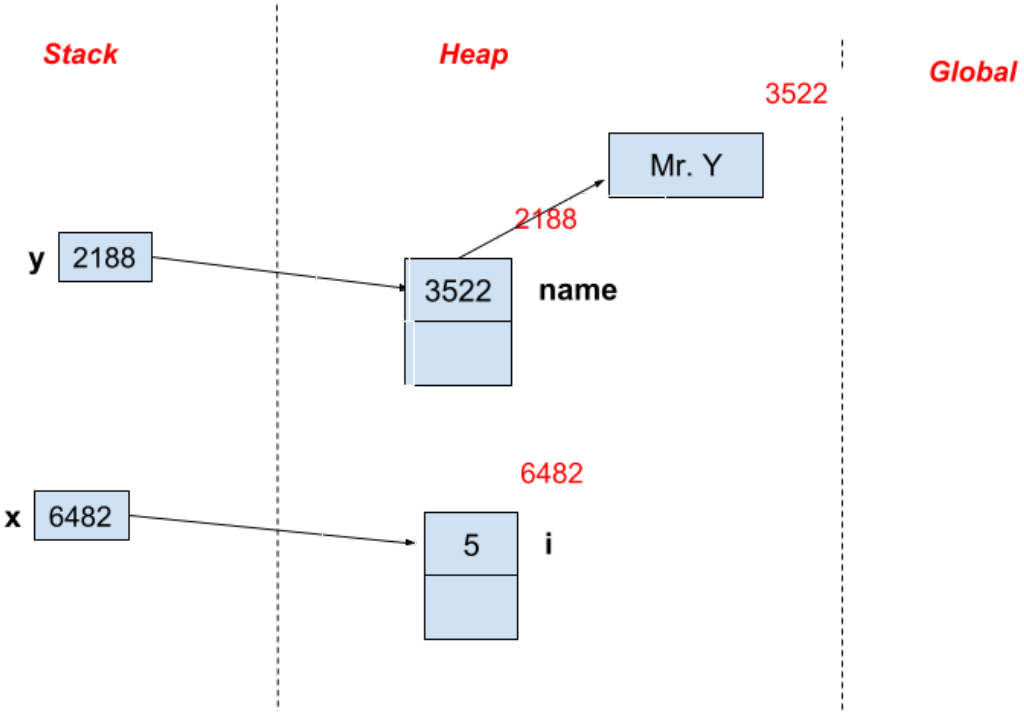
Ex. 3.8

Memory picture just after executing the one line in **printTheObject(),** but before the method returns:

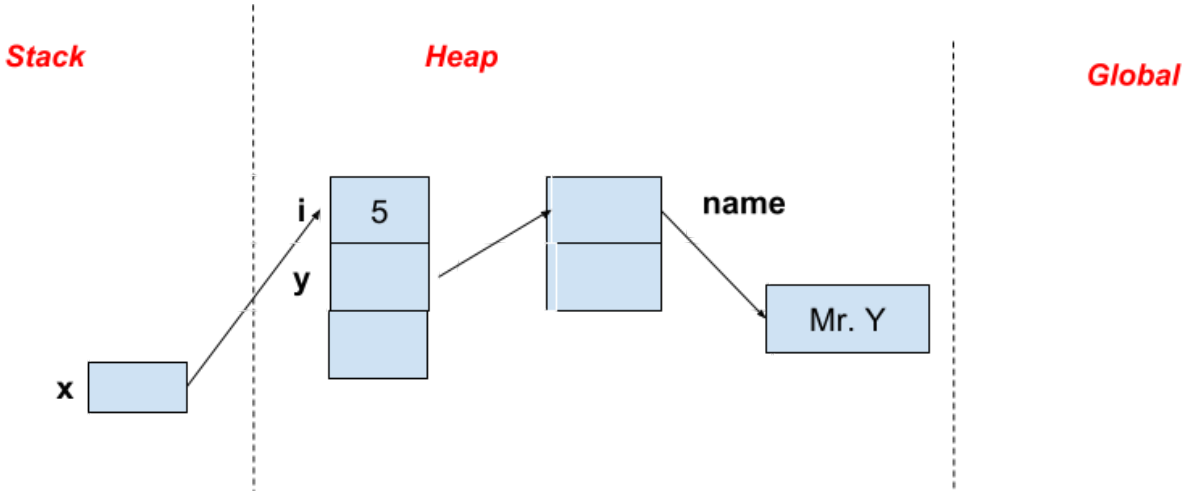


Ex. 3.9

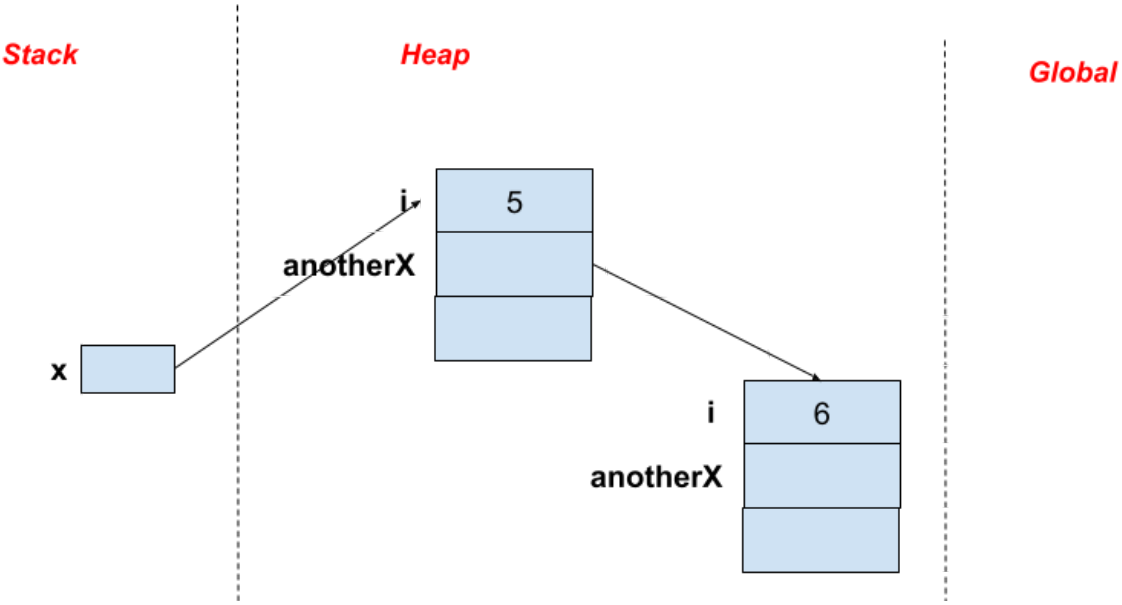
Memory picture of **DynamicExample8** just after the last statement in **main()**:



Ex. 3.11


Stack
Heap
Global
i
5
y
name
Mr. Y
x

Ex. 3.12


Stack
Heap
Global
i
5
anotherX
x
i
6
anotherX

Ex. 3.14

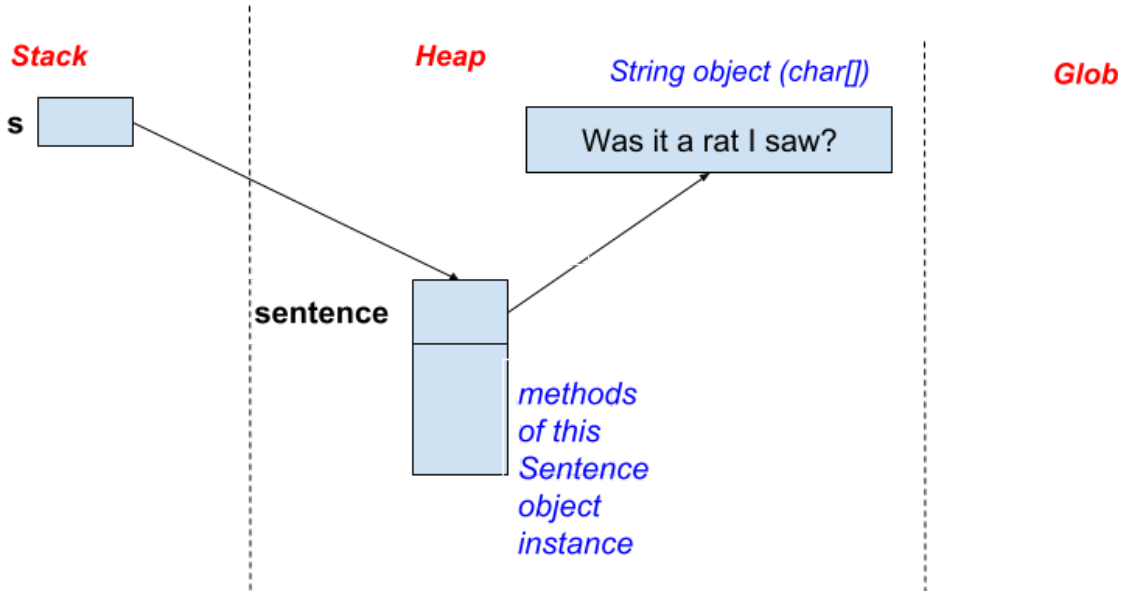
With toString() commented out, printed is the value of x

ObjX11@7ad041f3
Objtc x: ObjX11@7ad041f3

This value is the address of the object instance pointed to by x.

Ex. 3.15

Memory picture at the end of **main()** in **SentenceExample**:

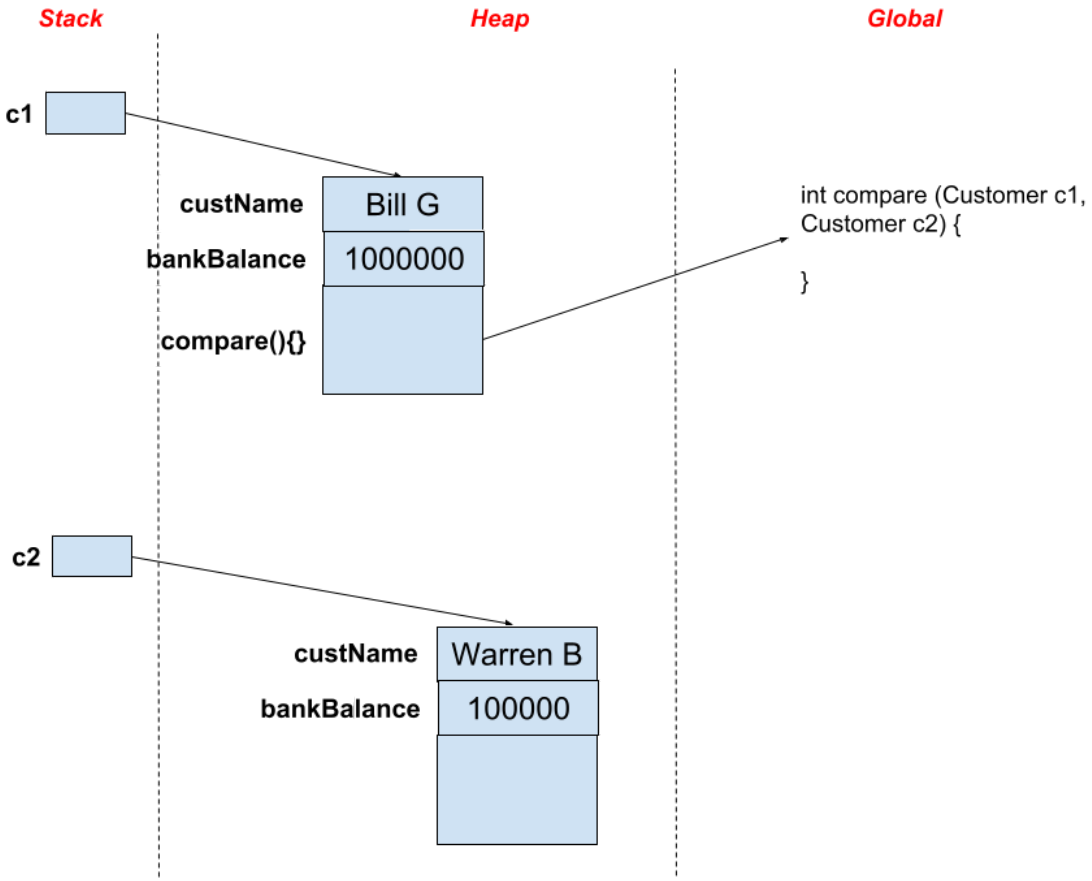


Ex. 3.16

The attempt to compile with a constructor without parameters generates an error:

```
DynamicExample12.java:31: error: no suitable constructor found for ObjX12(no
arguments)
        ObjX12 x = new ObjX12 ();
                   ^
    constructor ObjX12.ObjX12(String) is not applicable
      (actual and formal argument lists differ in length)
    constructor ObjX12.ObjX12(String,int) is not applicable
      (actual and formal argument lists differ in length)
1 error
```

Ex. 3.18



The answer is: c1

Because in main(), **compare()** is accessed through **c1** (**c1.compare (c1, c2)**), it is inside this object (**c1**) that compare() is executed.