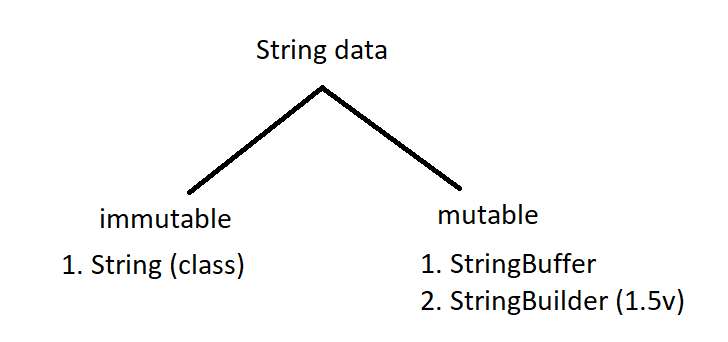
String

String is basically a inbuilt class present in java. lang. String package for which object can be created (user defined collection of characters enclosed in double quotes).

Eg: String\_Object\_Eg1



Since String is a class it may consists of instance variables and methods .

String :

class String{

// instance variables

// methods

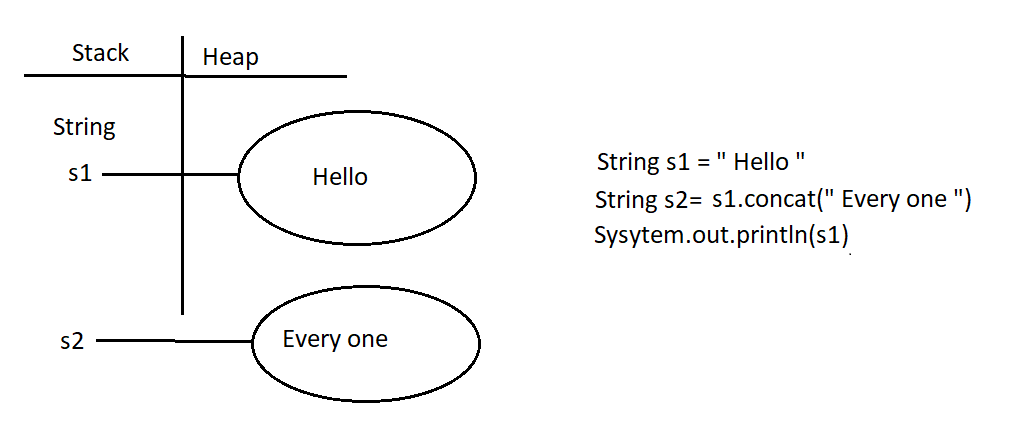
}

To access the string methods we should create an object and call them .

The object is the user defined string , with that reference variable of object we can call String methods

Eg: String\_Method\_Concat

Since string is immutable you cannot add anything to it once it is declared , but if we use concat method jvm creates a new object to concat . if that object is not collected by user , garbage collector will take that object .



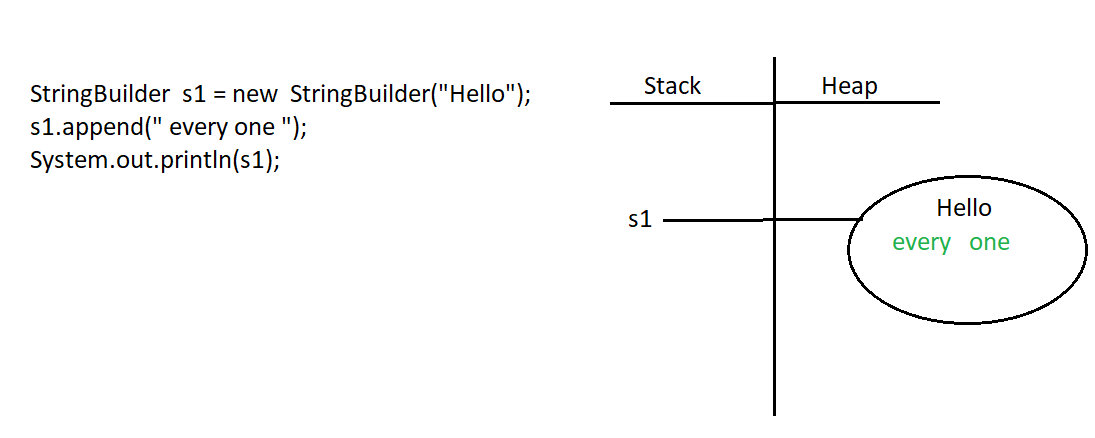
If concat method is used in the print statement there is no need of creating reference variable to collect that concat object.

If not used in the print statement or created reference variable for it , garbage collector will clear that.

StringBuilder

String builder is mutable , when it is declared an object is created . if we want to add something to that object we can use append() method of StringBuilder class . unlike String class jvm will not create another object to add new one . instead it adds to old object.

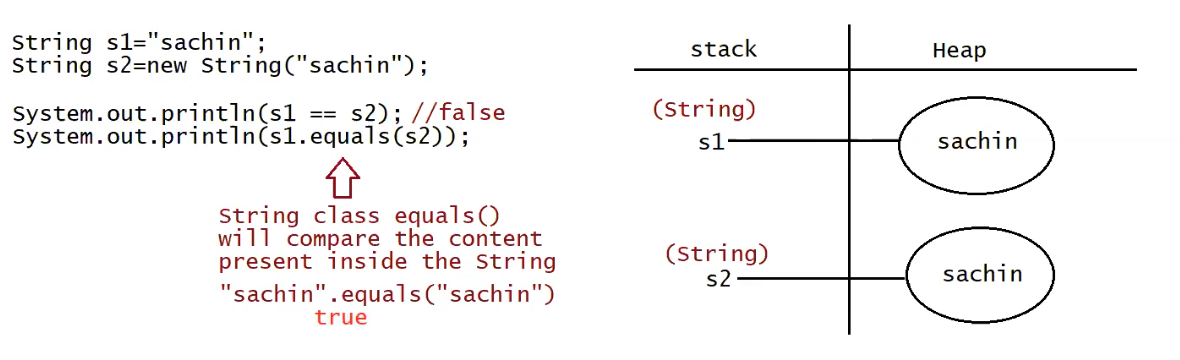
Eg: String\_Builder\_Eg3



String class equals method

Eg: String\_Equals\_Method\_Eg4

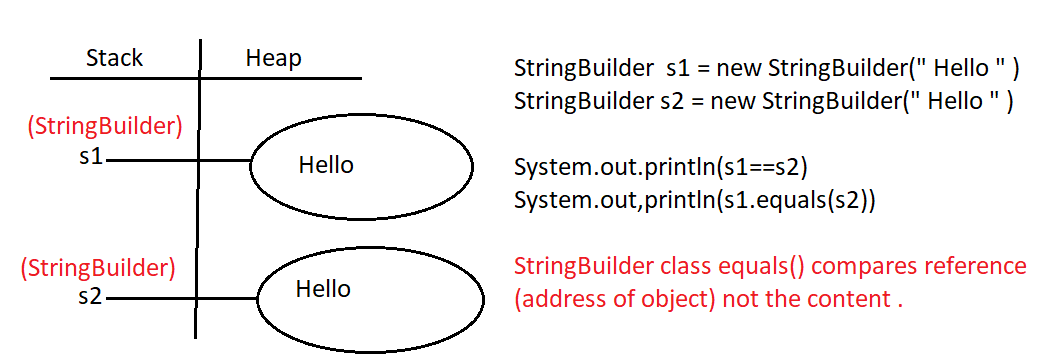
// go through the program



String \_Builder\_Equals \_Method

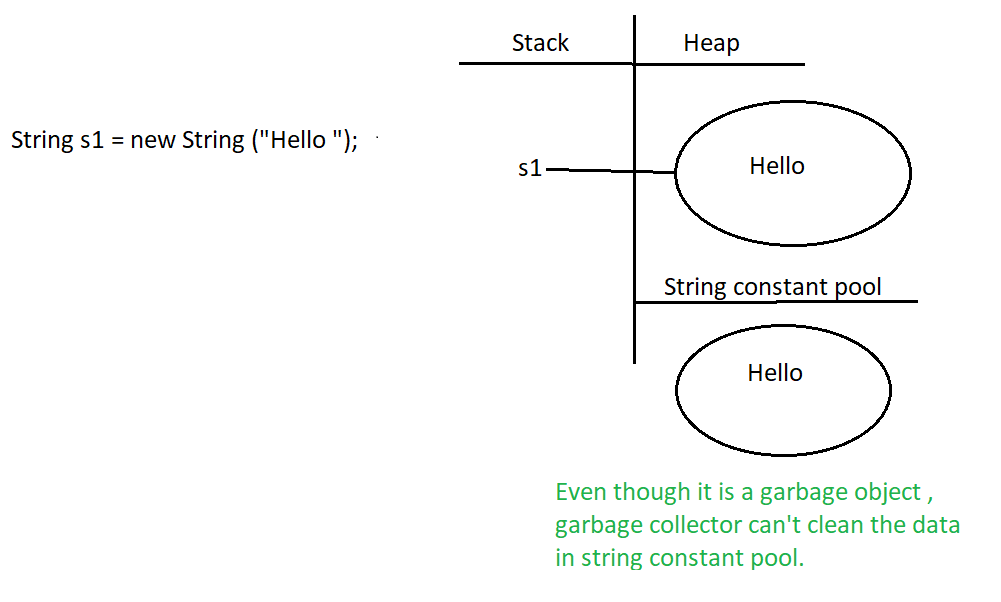
equals() method in String builder class compares the referece objects not the content in the objects.

Eg: String\_Builder\_Equals\_Method\_Eg5



String s1 = new String (“Hello”);

In this case two objects will created one in the heap , and the other in the String constant pool ( s . c . p ) . The reference always points out to the heap.



String s2 = “ Hello”;

In this case one object will be created in the string constant pool and it points out to the refernce

