

String Introduction

Strings:-

Strings can be defined as Array of characters or group of characters enclosed within " "

For example:

"Deep"

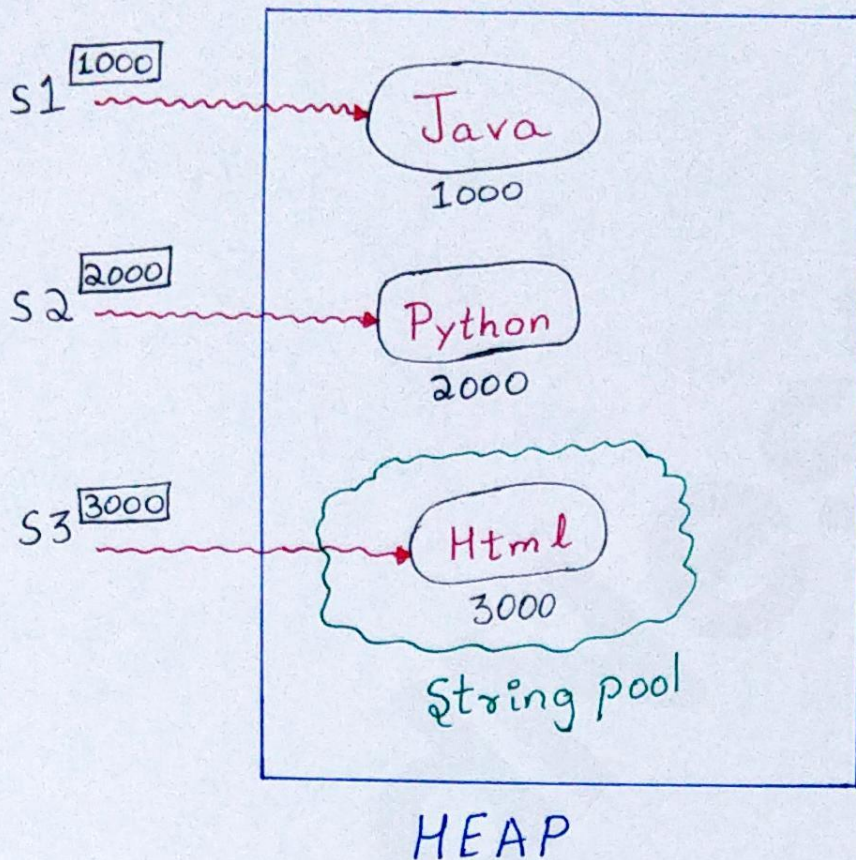
"Deep123"



Different ways to create Strings

```
class StringDemo
{
    public static void main (String args[])
    {
        String s1 = new String("Java");
        System.out.println ("s1:" + s1);
        char ch[] = {'p', 'y', 't', 'h', 'o', 'n'};
        String s2 = new String(s2);
        System.out.println ("s2:" + s2);
        String s3 = "HTML";
        System.out.println ("s3:" + s3);
    }
}
```





Output :-

s1 : Java

s2 : Python

s3 : Html



Specification of String Pool

Major difference between String Pool area and outside of String Pool area

| String Pool area | Outside string pool area |
|---|---|
| <ul style="list-style-type: none">i. Duplicate values are not allowedii. String literals created "without using new keyword" are stored in String Pooliii. Before storing any string value inside String Pool whether that value already present or not will be checked | <ul style="list-style-type: none">i. Duplicate values are allowedii. String literals created using 'new' keyword are stored outside String Pool.iii. If the string literal already present outside string pool then also same string literal is allowed |

Note: If the String literal already present in String Pool then the copy of the address will be returned.



Eg:

```
class StringDemo
{
    public static void main(String [] args)
    {
        String s1 = new String("Java"); //creating
        // a string object using 'new' keyword.
        String s2 = new String("Java");
        String s3 = "Python"; //creating a string object
        // without using new keyword.
        String s4 = "Python";
        // For comparing 2 string address (reference)
        // "==" operator is used and it returns boolean
        // result
        System.out.println(s1==s2); //checking is
        // address of s1 = s2
        System.out.println(s3==s4); //checking is
        // address of s3 = s4
    }
}
```



Output:

false
true

Memory Mapping:

