Core Java 8

Lesson 08 : String Handling

Lesson Objectives

After completing this lesson, participants will be able to:

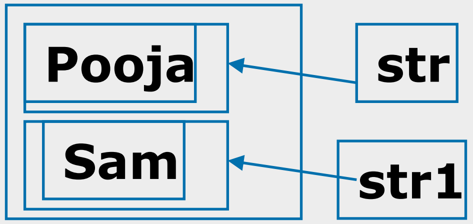
- Work with String Handling
- Understand new Date and Time API
- Best Practices

String Handling

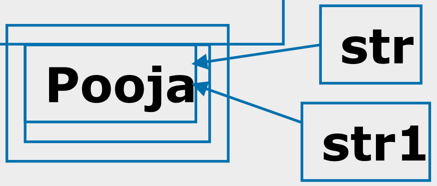
String is handled as an object of class String and not as an array of characters

- String class is a better & convenient way to handle any operation
- String objects are immutable

```
String str = new String("Pooja");
String str1 = new String("Sam");
```



```
String str = new String("Pooja");
String str1 = str;
```



String - Important Methods

length(): length of string

indexOf(): searches an occurrence of a char, or string within other string

substring(): Retrieves substring from the object

trim(): Removes spaces

valueOf(): Converts data to string

isEmpty(): Added in Java 6 to check whether string is empty or not

concat(String s) : Used to concatenate a string to an existing string. Eg

```
String string = "Core ";
System.out.println( string=string.concat(" Java") );
Output -> "Core Java”
```

String Concatenation

Use a “+” sign to concatenate two strings Examples:

- Example: String string = "Core " + "Java"; -> Core Java

```
String a = "String"; int b = 3; int c=7
System.out.println(a + b + c);  -> String37
```

```
System.out.println(a + (b + c)); -> String10
```

String Comparison

```
Output :  Hello equals Hello -> true
     Hello == Hello -> false
```

```
class EqualsNotEqualTo {
   public static void main(String args[]) {
         String str1 = "Hello";
         String str2 = new String(str1);
         System.out.println(str1 + " equals " + str2 + " -> " +
                      str1.equals(str2));
         System.out.println(str1 + " == " + str2 + " -> " + (str1 ==str2));
   }
}
```

StringBuffer Class

Following classes allow modifications to strings:

- java.lang.StringBuffer
- java.lang.StringBuilder

Many string object manipulations end up with a many abandoned string objects in the String pool, since String objects are immutable

```
StringBuffer sb = new StringBuffer("abc");
sb.append("def");
System.out.println("sb = " + sb); // output is "sb = abcdef"
```

StringBuilder Class

Added in Java 5

Exactly the same API as the *StringBuffer* class, except:

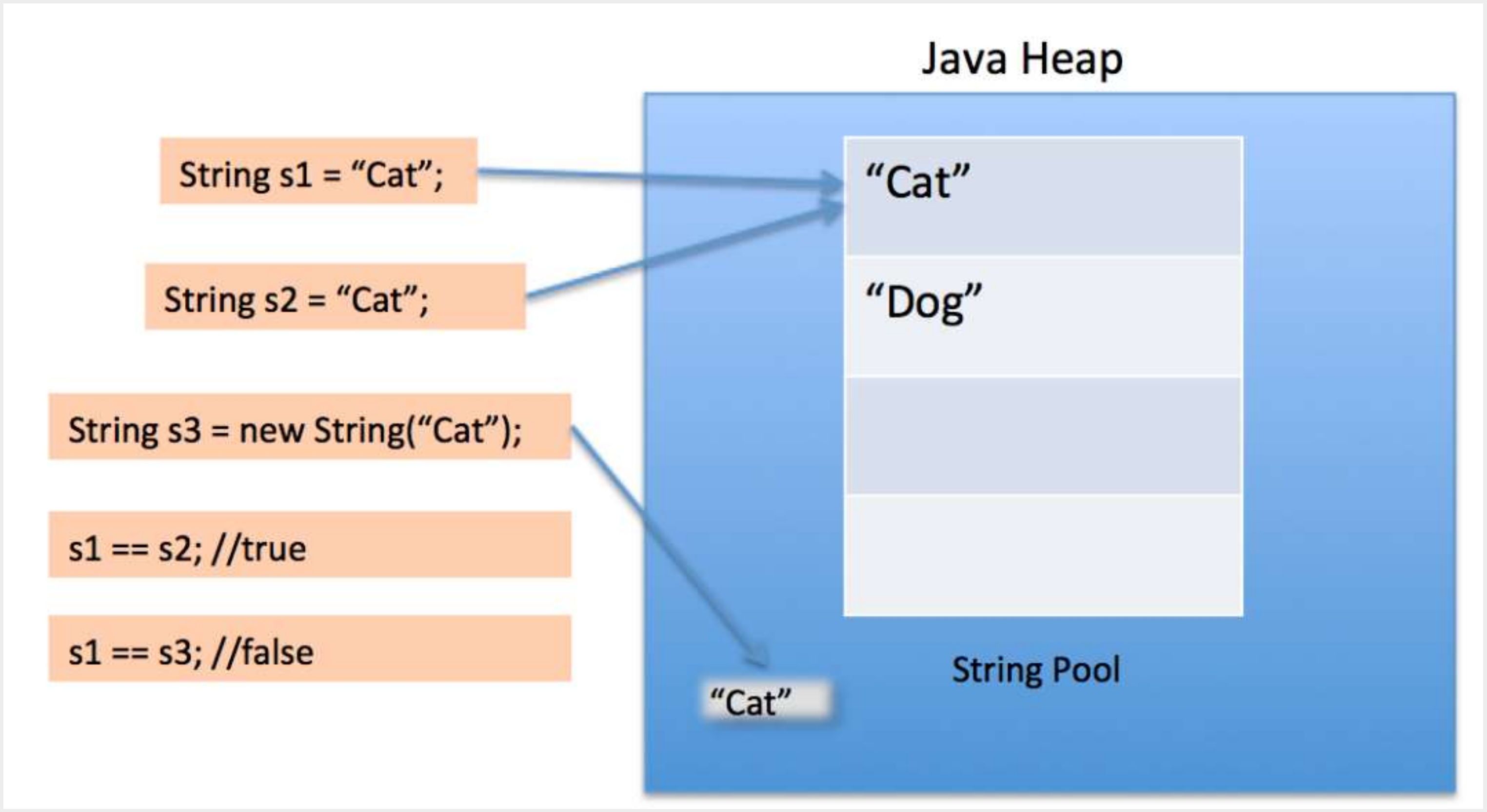
- It is not thread safe
- It runs faster than StringBuffer

```
StringBuilder sb = new StringBuilder("abc");
sb.append("def").reverse().insert(3, "---");
System.out.println( sb ); // output is "fed---cba"
```

Important Facts about Strings and Memory

String is immutable :

String Pool in java is a pool of Strings stored in Heap Memory .This is possible only because String is immutable in java.



Demo

Execute the following programs:

- SimpleString.java
- ToStringDemo.java
- StringBufferDemo.java
- CharDemo.java

Summary

In this lesson you have learnt:

- String Handling
- Best Practices

Review Questions

Question 1: String objects are mutable and thus suitable to use if you need to append or insert characters into them.

- True/False

Question 2: Which of the following static fields on wrapper class indicates range of values for its class:

- Option 1:MIN_VALUE
- Option 2: MAX_VALUE
- Option 3: SMALL_VALUE
- Option 4: LARGE_VALUE