**What are strings?**

Strings are nothing but the sequence of characters enclosed within double quotes. For example, “ABC”, “xyz”, “123” etc.

**How strings are represented in Java?**

In some other languages, strings are represented as array of characters. But in Java, strings are represented as objects of java.lang.String class.

**How do you create string objects in Java?**

There are two ways to create string objects in Java.

**1) Using String Literals**

String s1 = “ABC”;

String s2 = “123”;

**2) Using new Operator**

String s1 = new String(“ABC”);

String s2 = new String(“123”);

**How string objects are stored in the memory?**

Whenever you create string objects using string literals, those objects will be stored in string constant pool and whenever you create string objects using new operator, such objects will be stored in normal heap memory.

String constant pool is a part of heap memory which is specially dedicated to store string objects. JVM allocates pool space to an object depending upon it’s content. There will be no two objects in the string constant pool with same content.

Whenever you create a string object using string literal, JVM first checks content of an object to be created. If there exist an object in the pool with same content, then it returns reference of that object. It doesn’t create new object. If the content is different from the existing objects then only it creates new object.

**String objects are immutable :**

String objects, either created using string literals or using new operator, are immutable in nature. That means, once you create a string object, you can’t modify the contents of that object. If you try to modify the contents of a string object, a new string object is created with modified content.

**java.lang.StringBuffer Class :**

java.lang.StringBuffer class is used to create mutable and thread-safe string objects. In other terms, this class is same as java.lang.String class except it’s objects are mutable.

It is not possible to create StringBuffer objects using string literals. You have to use new operator to create StringBuffer objects.

**Important Methods :** append(), insert(), replace(), delete(), reverse(), length(), charAt() and substring().

**java.lang.StringBuilder Class :**

java.lang.StringBuilder class is used to create mutable and non thread-safe string objects. In other terms, this class is same as java.lang.StringBuffer class except it’s objects are not thread-safe.

It is also not possible to create StringBuilder objects using string literals. You have to use new operator to create StringBuilder objects.

**Important Methods :** append(), insert(), replace(), delete(), reverse(), length(), charAt() and substring().

**Difference Between String, StringBuffer and StringBuilder :**

|  |  |  |
| --- | --- | --- |
| **String** | **StringBuffer** | **StringBuilder** |
| Immutable | Mutable | Mutable |
| Thread-safe | Thread-safe | Not thread-safe |
| Objects can be created either through string literal or through new operator | Objects can be created only through new operator | Objects can be created only through new operator |
| Objects are stored in string constant pool as well as heap memory | Objects are stored in heap memory only. | Objects are stored in heap memory only. |
| Slower | Slower | Faster |

**String Intern :**

String intern refers to string object in the string constant pool.

**Interning** is the process of creating a string object in String Constant Pool which will be exact copy of string object in heap memory.

**intern()**methodof java.lang.String class is used to perform interning i.e creating an exact copy of heap string object in string constant pool.