String Handling

Strings

What is a string ?

- Java platform provides a “String” class to create and manipulate strings.
- A String is an object that represents a sequence of characters.
- There are two ways to create a string object
  - Using string literal
  - Using new operator

Ways to create a String object

- Using String literal
  - A series of characters enclosed in double quotes.

```
String str = “hello”;
```

- Using new operator
  - String object can be created using a new operator.

```
String str = new String(“hello”);
```

Features of String

- String Manipulation

```
String oldString = “java learning”;
String newString = str.replace(“learning”,”Programming”);
```

- String Concatenation
  - concat():

```
String str = “Java”;
String res = str.concat(“Programming”);
```

  - Using “+” :

```
String str = “Java”+  “+”Programming”;
```

Continues…

- String Conversion:

```
String intVal = String.valueof(10000);
String doubleVal = String.valueOf(10.2/3.4);
```

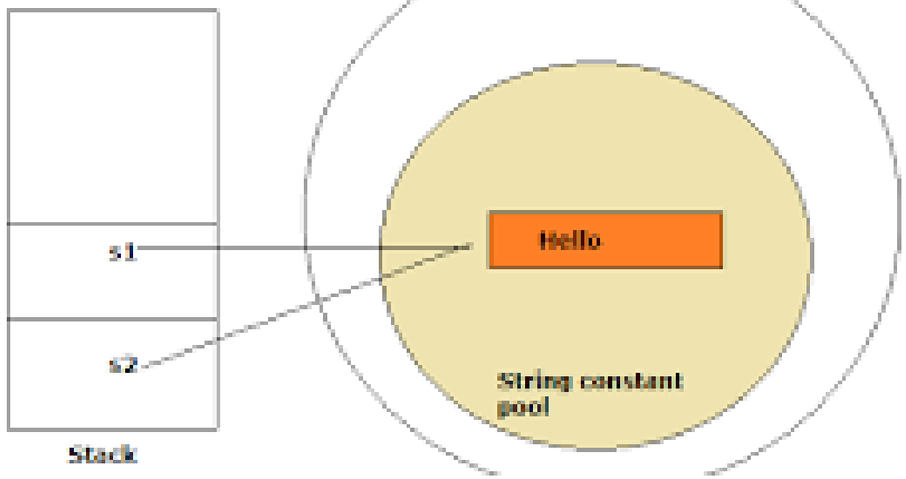
String Class Methods

- **charAt(int i) :** Returns the character located at the specified index
- **equalsIgnoreCase(String str) :** Determines the equality of two Strings, ignoring case
- **length() :**Returns the number of characters in a String
- **substring(int startIndex):** Returns a part of a String
- **toLowerCase() :** Returns a String with uppercase characters converted
- **toString() :** Returns the value of a String
  - **Refer API for rest of the method parameters and return types**

Strings – Memory Management

- String Constant Pool
  - JVM sets aside a special area of memory for Strings
- Compiler checks the pool for existence of identical String
- If found, the reference is assigned to the existing String and new literal object is not created
- The String class is final, so no behavior of the String methods can be overridden

String s1 = “Hello”;
String s2 =“Hello”;



StringBuffer

StringBuffer class

- Java StringBuffer class is used to create mutable (modifiable) string objects.
- Methods of a StringBuffer class are thread-safe(synchronized)
- Syntax:

```
StringBuffer buffer = new
StringBuffer(“JavaProgramming”);
```

Methods of StringBuffer class

- append(): The characters of the String argument are appended.

```
StringBuffer buffer = new StringBuffer(“Java”);
buffer.append(“Programming”);
```

- insert(): The characters of the String argument are inserted, into the sequence indicated in the offset

```
Synatx: insert(int offset, String str);
StringBuffer sb = new StringBuffer(“Java");
sb.insert(3,“Programming");
```

Continues…

- delete(int start, int end)
- deleteCharAt(int index)
- replace(int start, int end, String s)
- setCharAt(int index, char c)
- reverse()
- toString()

StringBuilder

StringBuilder class

- Java StringBuilder class is used to create mutable (modifiable) string
- Instances of StringBuilder are not safe to use by multiple threads(non-synchronized).

```
StringBuilder builder = new
StringBuilder(“JavaProgramming”);
```

StringTokenizer

StringTokenizer

- StringTokenizer class is available in the java.util package
- Can be used to break a string into separate tokens.
- A token is a portion of a string that is separated from another portion of that string by one or more chosen characters
- Chosen characters are called as delimeters.

```
String msg = "http://10.123.43.67:/";
StringTokenizer st = new StringTokenizer(msg,"://.“false);
```