String Methods in Java

String Methods

1. int length(): Returns the number of characters in the String.

```
"GeeksforGeeks".length();  // returns 13
```

2. Char charAt(int i): Returns the character at ith index.

```
"GeeksforGeeks".charAt(3); // returns  'k'
```

3. String substring (int i): Return the substring from the ith index character to end.

```
"GeeksforGeeks".substring(3); // returns "ksforGeeks"
```

4. String substring (int i, int j): Returns the substring from i to j-1 index.

```
"GeeksforGeeks".substring(2, 5); // returns "eks"
```

5. String concat(String str): Concatenates specified string to the end of this string.

```
String s1 = "Geeks";
 String s2 = "forGeeks";
 String output = s1.concat(s2); // returns "GeeksforGeeks"
```

6. int indexOf (String s): Returns the index within the string of the first occurrence of the specified string.

```
String s = "Learn Share Learn";
 int output = s.indexOf("Share"); // returns 6
```

7. int indexOf (String s, int i): Returns the index within the string of the first occurrence of the specified string, starting at the specified index.

```
String s = "Learn Share Learn";
 int output = s.indexOf("ea",3);// returns 13
```

8. Int lastIndexOf(String s): Returns the index within the string of the last occurrence of the specified string.

```
String s = "Learn Share Learn";
```

```
int output = s.lastIndexOf("a"); // returns 14
```

9. boolean equals(Object otherObj): Compares this string to the specified object.

```
Boolean out = "Geeks".equals("Geeks"); // returns true
Boolean out = "Geeks".equals("geeks"); // returns false
```

10. boolean equalsIgnoreCase (String anotherString): Compares string to another string, ignoring case considerations.

```
Boolean out= "Geeks".equalsIgnoreCase("Geeks"); // returns true
Boolean out = "Geeks".equalsIgnoreCase("geeks"); // returns true
```

11. int compareTo(String anotherString): Compares two string lexicographically.

```
int out = s1.compareTo(s2);  // where s1 ans s2 are
                             // strings to be compared

This returns difference s1-s2. If :
out < 0  // s1 comes before s2
out = 0  // s1 and s2 are equal.
out > 0   // s1 comes after s2.
```

12. int compareToIgnoreCase(String anotherString): Compares two string lexicographically, ignoring case considerations.

```
int out = s1.compareToIgnoreCase(s2);
// where s1 ans s2 are
// strings to be compared

This returns difference s1-s2. If :
out < 0  // s1 comes before s2
out = 0   // s1 and s2 are equal.
out > 0   // s1 comes after s2.
```

- *Note- In this case, it will not consider case of a letter (it will ignore whether it is uppercase or lowercase).*

13. String toLowerCase(): Converts all the characters in the String to lower case.

```
String word1 = "HeLLo";
String word3 = word1.toLowerCase(); // returns "hello"
```

14. String toUpperCase(): Converts all the characters in the String to upper case.

```
String word1 = "HeLLo";
String word2 = word1.toUpperCase(); // returns "HELLO"
```

15. String trim(): Returns the copy of the String, by removing whitespaces at both ends. It does not affect whitespaces in the middle.

```
String word1 = " Learn Share Learn ";
String word2 = word1.trim(); // returns "Learn Share Learn"
```

16. String replace (char oldChar, char newChar): Returns new string by replacing all occurrences of *oldChar* with *newChar.*

```
String s1 = "feeksforfeeks";
String s2 = "feeksforfeeks".replace('f' ,'g'); // returns "geeksgorgeeks"
```

- *Note:- s1 is still feeksforfeeks and s2 is geeksgorgeeks*