



# Java String Methods

Java is a programming language that provides a set of built-in methods that can be used to manipulate strings. Here are some commonly used string methods in Java:

1. `length()` method: This method returns the length of a string.

```
String str = "Hello, world!";  
int len = str.length();  
System.out.println("Length of the string is: " + len);
```



Length of the string is: 13

Example:

1. `substring()` method: This method returns a substring of a string based on the specified starting and ending index.

```
String str = "Hello, world!";  
String substr = str.substring(0, 5);  
System.out.println("Substring is: " + substr);
```



Substring is: Hello

1. `equals()` method: This method compares two strings and returns true if they are equal, otherwise it returns false.

Example:

```
String str1 = "Hello";
String str2 = "hello";
if (str1.equals(str2)) {
    System.out.println("Strings are equal.");
} else {
    System.out.println("Strings are not equal.");
}
```



Strings are not equal.

1. `toUpperCase()` method: This method converts all characters in a string to uppercase.

Example:

```
String str = "Hello, world!";
String strUpper = str.toUpperCase();
System.out.println("Uppercase string is: " + strUpper);
```



Uppercase string is: HELLO, WORLD!

1. `replace()` method: This method replaces all occurrences of a specified character or substring with another character or substring.

Example:

```
String str = "Hello, world!";
String strReplace = str.replace("o", "x");
System.out.println("Replaced string is: " + strReplace);
```



Replaced string is: Hellx, wxrld!

1. `indexOf()` method: This method returns the index of the first occurrence of a specified character or substring in a string. If the specified character or substring is not found, it returns -1.

Example:

```
javaCopy code
String str = "Hello, world!";
int index = str.indexOf("o");
System.out.println("Index of 'o' is: " + index);
```

1. `trim()` method: This method returns a copy of the string with leading and trailing whitespace removed.

```
String str = " Hello, world! ";
String trimmedStr = str.trim();
System.out.println("Trimmed string is: " + trimmedStr);
```

1. `startsWith()` method: This method checks if a string starts with a specified prefix and returns true if it does, otherwise it returns false.

```
String str = "Hello, world!";
boolean startsWithHello = str.startsWith("Hello");
System.out.println("Starts with 'Hello': " + startsWithHello);
```

1. `endsWith()` method: This method checks if a string ends with a specified suffix and returns true if it does, otherwise it returns false.

```
String str = "Hello, world!";  
boolean endsWithWorld = str.endsWith("world!");  
System.out.println("Ends with 'world!': " + endsWithWorld);
```

1. `valueOf()` method: This method converts a value of any type to a string.

```
int num = 42;  
String str = String.valueOf(num);  
System.out.println("String value of the number is: " + str);
```