

# Métodos de string

## length() Método de string em Java

Em Java, o `length()` método string retorna o número total de caracteres – o comprimento – de um arquivo

`String` .

```
String str = "Codecademy";
```

```
System.out.println(str.length());  
// prints 10
```

## concat() método de string em Java

Em Java, o `concat()` método string é usado para anexar um `String` ao final de outro `String` . Este método retorna uma `String` representação do texto das strings combinadas.

```
String s1 = "Hello";  
String s2 = " World!";
```

```
String s3 = s1.concat(s2);  
// concatenates strings s1 and s2
```

```
System.out.println(s3);  
// prints "Hello World!"
```

## Método String equals() em Java

Em Java, o `equals()` método string testa a igualdade entre dois `String` s.

`equals()` compara o conteúdo de cada um `String` . Se todos os caracteres entre os dois corresponderem, o método retornará `true` . Se algum dos caracteres não corresponder, ele retornará `false` .

Além disso, se você quiser comparar duas strings sem considerar maiúsculas/minúsculas, você pode usar

`.equalsIgnoreCase()` .

```
String s1 = "Hello";  
String s2 = "World";
```

```
System.out.println(s1.equals("Hello"));  
// prints true
```

```
System.out.println(s2.equals("Hello"));  
// prints false
```

```
System.out.println(s2.equalsIgnoreCase("w  
orld"));  
// prints true
```

## método de string indexOf() em Java

In Java, the `indexOf()` string method returns the first occurrence of a character or a substring in a `String`. The character/substring that you want to find the index of goes inside of the `()`. If `indexOf()` cannot find the character or substring, it will return `-1`.

```
String str = "Hello World!";

System.out.println(str.indexOf("l"));
// prints 2

System.out.println(str.indexOf("Wor"));
// prints 6

System.out.println(str.indexOf("z"));
// prints -1
```

## charAt() String Method in Java

In Java, the `charAt()` string method returns the character of a `String` at a specified index. The index value is passed inside of the `()`, and should lie between `0` and `length()-1`.

```
String str = "This is a string";

System.out.println(str.charAt(0));
// prints 'T'

System.out.println(str.charAt(15));
// prints 'g'
```

## toUpperCase() and toLowerCase() String Methods

In Java, we can easily convert a `String` to upper and lower case with the help of a few string methods:

`toUpperCase()` returns the string value converted to uppercase.

`toLowerCase()` returns the string value converted to lowercase.

```
String str = "Hello World!";

String uppercase = str.toUpperCase();
// uppercase = "HELLO WORLD!"

String lowercase = str.toLowerCase();
// lowercase = "hello world!"
```