

String Comparison and Important Methods in Java



String Comparison - Overview

➡ In Java, you can compare strings
in 3 main ways:

- `.equals()`
- `==`
- `.compareTo()`

Let's explore each!

Using .equals()

- ✅ Compares the content of two strings.

java

```
String s1 = "hello";  
String s2 = "hello";  
System.out.println(s1.equals(s2)); // true
```

Note:

Always use .equals() to compare
string values!

Using ==

✅ Compares references, not content.

java

```
String s1 = "hello";  
String s2 = new String("hello");  
System.out.println(s1 == s2); // false
```

Note:

== checks if both variables point to the same object.

Using .compareTo()

Example :

```
java
```

```
String s1 = "apple";  
String s2 = "banana";  
System.out.println(s1.compareTo(s2)); // Negative value
```

Returns :

- 0 if equal
- Negative if $s1 < s2$
- Positive if $s1 > s2$

Important String Methods (Part 1)

Method	Example	Result
<code>.length()</code>	<code>"hello".length()</code>	5
<code>.charAt(1)</code>	<code>"hello".charAt(1)</code>	'e'
<code>.substring(1, 4)</code>	<code>"hello".substring(1, 4)</code>	"ell"

Important String Methods (Part 2)

Method	Example	Result
<code>.contains("ell")</code>	<code>"hello".contains("ell")</code>	true
<code>.equalsIgnoreCase("HELLO")</code>	<code>"hello".equalsIgnoreCase("HELLO")</code>	true
<code>.toUpperCase()</code>	<code>"hello".toUpperCase()</code>	HELLO

Important String Methods (Part 3)

Method	Example	Result
<code>.trim()</code>	<code>" hello ".trim()</code>	hello
<code>.replace("l", "p")</code>	<code>"hello".replace("l", "p")</code>	heppo
<code>.split(",")</code>	<code>"a,b,c".split(",")</code>	<code>["a", "b", "c"]</code>

Thank You