

POR CRISTIANE PAGINE



JAVA

INTRODUÇÃO À CLASSE STRING

2025



Introdução

Em Java, a String é uma classe imutável, o que significa que seu valor não pode ser alterado depois de criado. No entanto, existem diversos métodos para manipular e trabalhar com strings, que geram novas instâncias de String com os resultados.

Obter Informações sobre a String

- `length()`: Retorna o tamanho da string (número de caracteres).
- `charAt(int index)`: Retorna o caractere em uma posição.
- `isEmpty()`: Verifica se a string está vazia.

```
String texto = "Java";  
System.out.println(texto.length());  
// Saída: 4  
System.out.println(texto.charAt(1));  
// Saída: a  
System.out.println("").isEmpty());  
// Saída: true
```

Manipulação de Conteúdo:

- `toUpperCase()` e `toLowerCase()`: Convertem para maiúsculas ou minúsculas.
- `trim()`: Remove espaços no início e no fim da string.
- `replace(char oldChar, char newChar)`: Substitui todas as ocorrências de um caractere por outro.
- `substring(int beginIndex, int endIndex)`: Extrai uma parte da string.

```
String texto = "Java";
System.out.println(texto.toUpperCase());
// Saída: JAVA

System.out.println(texto.toLowerCase());
// Saída: java

String textoComEspacos = "  Java  ";
System.out.println(textoComEspacos.trim());
// Saída: Java

System.out.println(texto.replace('a', 'o'));
// Saída: Jovo

System.out.println(texto.substring(1, 3));
// Saída: av
```

Comparação de Strings:

- `equals(Object obj)`: Verifica se duas strings são iguais (considera maiúsculas e minúsculas).
- `equalsIgnoreCase(String anotherString)`: Verifica igualdade ignorando maiúsculas e minúsculas.
- `compareTo(String anotherString)`: Compara duas strings lexicograficamente.
- Retorna 0 se forem iguais.
- Retorna um valor negativo se a string atual for menor.
- Retorna um valor positivo se a string atual for maior.

```
String texto = "Java";
```

```
System.out.println(texto.equals("Java"));  
// Saída: true
```

```
System.out.println(texto.equals("java"));  
// Saída: false
```

```
System.out.println(texto.equalsIgnoreCase("java")); //  
Saída: true
```

```
System.out.println("Java".compareTo("Jaca"));  
// Saída: 21
```

Busca em Strings:

- `contains(CharSequence seq)`: Verifica se a string contém uma sequência de caracteres.
- `indexOf(String str)`: Retorna a posição da primeira ocorrência de um caractere ou sequência.
- `lastIndexOf(String str)`: Retorna a posição da última ocorrência de um caractere ou sequência.
- `startsWith(String prefix)` e `endsWith(String suffix)`: Verifica se a string começa ou termina com um prefixo/sufixo.

```
String texto = "Java";
```

```
System.out.println(texto.contains("va"));  
// Saída: true
```

```
System.out.println(texto.indexOf('a'));  
// Saída: 1
```

```
System.out.println("Java Java".lastIndexOf("Java"));  
// Saída: 5
```

```
System.out.println(texto.startsWith("Ja"));  
// Saída: true
```

```
System.out.println(texto.endsWith("va"));  
// Saída: true
```

Divisão e Junção de Strings:

- `split(String regex)`: Divide a string em um array, usando um delimitador.
- `join(CharSequence delimiter, CharSequence... elements)`: Junta elementos de um array com um delimitador.

```
String frase = "Java é incrível";
String[] palavras = frase.split(" ");
System.out.println(palavras[0]);
// Saída: Java

String[] nomes = {"Java", "Python", "C++"};
System.out.println(String.join(", ", nomes));
// Saída: Java, Python, C++
```

Exemplo prático:

```
public class StringExample {
    public static void main(String[] args) {
        String texto = " Java é Incrível! ";

        // Remover espaços e converter para maiúsculas
        String textoFormatado =
texto.trim().toUpperCase();
        System.out.println(textoFormatado); // Saída:
JAVA É INCRÍVEL!

        // Substituir palavras
        String novoTexto =
textoFormatado.replace("INCRÍVEL", "PODEROSO");
        System.out.println(novoTexto); // Saída: JAVA
É PODEROSO!

        // Verificar conteúdo
        System.out.println(novoTexto.contains("JAVA")); //
Saída: true
    }
}
```