



Strings

Rafael Vieira Coelho

Strings

- ? Criação de strings
- ? Concatenação de strings
- ? Conversão de string
- ? Extração de caracteres
- ? Comparação
- ? igualdade
- ? Modificações do texto de strings

Tratamento de Strings

- ❓ Uma string é uma seqüência ou cadeia de caracteres.
- ❓ Java tem uma classe embutida no pacote `java.lang` a qual encapsula a estrutura de dados de uma string.
- ❓ A classe é chamada de **String** e é a representação de objeto de um array de caractere imutável.
- ❓ Há diversos métodos para manipular strings na classe `String`.

Formas de Criar um Texto em Java

```
char chs[ ] = {'a', 'b', 'c'};
```

```
String t = "abc";
```

Tamanho de uma String

? `System.out.println(s.length());`

? Pode-se chamar os métodos diretamente em uma string como se ela fosse uma referência ao objeto:

? `System.out.println("abc".length());`

Concatenação de strings

```
int idade = 24;  
String s = "Ele tem " + idade + " anos de idade."  
  
System.out.println (s);
```

Ele tem 24 anos de idade.

```
double media = (3.1 + 5.0 + 4.5)/4.0;  
  
System.out.println ("O valor da média é " + media + "!");
```

O valor da média é 3.15!

Conversão de Tipos (CAST)

```
int i = 24;  
String s = String.valueOf(i);
```

i = 24
s = "24"

```
String s = "12";  
int i = Integer.parseInt(s);
```

s = "12"
i = 12

Conversão de Tipos (CAST)

```
double d = 8.56;  
String s = String.valueOf(d);
```

d = 8.56
s = "8.56"

```
String s = "4.3";  
double d = Double.parseDouble(s);
```

s = "4.3"
d = 4.3

Extração de caracteres

? Para extrair um único caractere de uma string, use o método **charAt**.

? “abc”.charAt(1) retornará ‘b’

? Para extrair mais de um caractere use o método **getChars**.

Comparação

? Use o método **equals** para comparar duas strings.

? O valor de retorno será true ou false.

```
String s1 = "eu";  
String s2 = "nós";  
if (s1.equals(s2))  
    System.out.println("São iguais");  
else  
    System.out.println("São diferentes");
```

? O método **equalsIgnoreCase** compara ignorando maiúsculas e minúsculas.

Exemplo

```
class Igualstr{  
  public static void main (String args[ ]){  
    String s1 = "Hello";  
    String s2 = "Hello";  
    String s3 = "Alo";  
    String s4 = "HELLO";  
  
    System.out.println(s1.equals(s2)); -> true  
    System.out.println(s1.equals(s3)); -> false  
    System.out.println(s1.equals(s4)); -> false  
    System.out.println(s1.equalsIgnoreCase(s4)); -> true  
  }  
}
```

Igualdade

- ❓ O método **equals** e o operador **==** fazem dois testes de igualdade totalmente diferentes.
- ❓ Método **equals** compara caracteres dentro de uma string
- ❓ Operador **==** compara duas referências de objetos para ver se elas se referem a mesma instância.

indexOf e lastIndexOf

? Estes métodos retornam o índice do caractere que se está procurando ou o índice do início da substring.

? indexOf - primeira ocorrência

? lastIndexOf - última ocorrência.

? `int indexOf(char ch);`

? `int lastIndexOf(char ch);`

substring

? "Hello World".substring(6)

? "World"

? "Hello World".substring(3, 8)

? "lo Wo"

replace

- ? Toma dois caracteres como parâmetros.
- ? Todas as ocorrências do primeiro caractere são substituídas pela segunda.
- ? `"Hello".replace('l', 'w');`
 - ? **"Hewwo"**

toLowerCase e toUpperCase

☐ "Hello".toLowerCase();

☐ "hello"

☐ "Hello".toUpperCase();

☐ "HELLO"

☐ " Hello World ".trim();

☐ "Hello World"

E quando for char?

```
char letra = 'A';
```

```
if (Character.isLowerCase(letra)) {  
    System.out.println("É minúscula");  
} else if (Character.isUpperCase(letra)) {  
    System.out.println("É maiúscula");  
}
```

Método toCharArray

? Podemos transformar uma String em um vetor de caracteres:

? "Java".toCharArray() -> {'J','a','v','a'}

```
String texto = "Este é o meu texto.";
```

```
char[] vetor = texto.toCharArray();
```

index	0	1	2	3	4	5	6	7	8	9
elements	'E'	's'	't'	'e'	' '	'é'	' '	'o'	' '

Exercícios URI



URI1234.java



URI2242.java



URI2866.java