



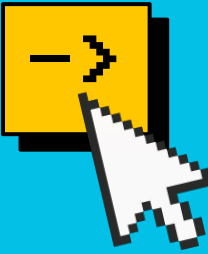
COMPETEC





Aula nº05

Strings



Assuntos da aula

- >O que são Strings
- >Como declarar uma String
- >Métodos e funções das Strings

Ok

Cancel

Options<<

Strings são objetos que contém uma sequência de caracteres e são criados de forma muito semelhante a uma variável de tipo primitivo.

```
String competec = "COMPETEC";  
String nome = "Bruno";
```

Assim como as variáveis que vimos antes, um Scanner pode ler uma String do teclado. Basta usar o método `nextLine()`, que lê uma linha inteira.

```
Scanner scan = new Scanner(System.in);
```

```
String entrada = scan.nextLine();
```

```
//digite: A B C D
```

```
System.out.println(entrada);
```

```
//imprime: A B C D
```

O método *length* retorna o tamanho de uma String, ou seja, a quantidade de caracteres.

```
String competec = "COMPETEC";
```

```
int tamanho = competec.length( );
```

```
System.out.println(tamanho);
```

```
//imprime: 8
```

O método *charAt* retorna o char, ou seja, o caractere em um índice específico. Lembrando que a primeira letra da String tem índice 0.

```
String competec = "COMPETEC!";
```

```
System.out.println(competec.charAt(0));
```

```
//imprime: C
```

```
char ultimo = competec.charAt(competec.length() - 1);
```

```
System.out.println(ultimo);
```

```
//imprime: !
```

Os métodos *toLowerCase* e *toUpperCase* são utilizados para mudar a capitalização de Strings.

```
String competec = "ComPeTeC";
```

```
System.out.println(competec.toLowerCase());
```

```
//imprime: competec
```

```
System.out.println(competec.toUpperCase());
```

```
//imprime: COMPETEC
```


Apesar de poderem ser criadas como um tipo primitivo, as strings não podem ser comparadas da mesma forma (`==`).

Para compará-las, utilizamos o método *equals* e ele retorna um boolean.

```
String competec = scan.nextLine();
```

```
//digite: COMPETEC
```

```
System.out.println(competec == "COMPETEC");
```

```
//imprime: false
```

```
System.out.println(competec.equals("COMPETEC"));
```

```
//imprime: true
```

Existe também o método *equalsIgnoreCase*, que ignora a diferença entre maiúsculas e minúsculas.

```
String competec = "ComPeTeC";
```

```
System.out.println(competec.equals("COMPETEC"));
```

```
//imprime: false
```

```
System.out.println(competec.equalsIgnoreCase("COMPETEC"));
```

```
//imprime: true
```

Split é um método que separa uma String a partir de uma referência.

O retorno do método é um array, onde em cada campo dele guarda uma das partes da String separada.

Por exemplo, se quisermos separar as palavras de uma frase, basta usar o *split* e dizer para o método separar a String pelos espaços que contêm nela.

```
String frase = "A persistência é o caminho do êxito";  
String[] separado = frase.split(" "); //separando pelo espaço  
System.out.println(separado[1]); //imprime: persistência
```

0	1	2	3	4	5	6
A	persistência	é	o	caminho	do	êxito

- **Concat():** *concat* retorna duas strings concatenadas, ou seja, juntas.
- **Contains():** é um método que verifica se uma sequência de caracteres está contida em uma String. Contains é case-sensitive.
- **IndexOf():** *indexOf* também irá procurar se uma sequência de caracteres está contida numa String. A diferença é que, ao invés de retornar true ou false, ele retornará a posição do primeiro caractere do início da sequência.
- **compareTo():** O método `compareTo()` compara duas strings em ordem alfabética. Retorna um número negativo se a primeira string for antes, zero se forem iguais e um número positivo se a primeira string for depois da segunda.
- **Substring():** Substring é um método para “cortar” uma String. Use-o quando quiser pegar uma parte específica do texto.

Dica: pesquisar no google como utilizar esse métodos e outros existentes.

J main.java X

J main.java

1. Crie um programa que conta o número de vogais em uma string recebida pelo usuário.
2. Crie um programa que inverta uma String (deixar de trás pra frente).

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

J main.java X

J main.java

3. (Opcional) Escreva um programa que conta quantas vezes uma determinada palavra ocorre em uma frase dada pelo usuário.

4. (Opcional) Escreva apenas os números de um CPF e formate ele para o formato tradicional.

Ex: 12345678922 > 123.345.678-22

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

COMPETEC



Obrigado pela
atenção! :)

Don't show this message again

Ok

