

Java

(String)

Entrada / Saída

```
public static void main(String[] args) {  
    Scanner ler = new Scanner(System.in);  
    String nome;  
    System.out.println("Olá! Sou Computalino. E você?");  
    nome = ler.nextLine();  
    System.out.println("Bonito nome, " + nome + "!");  
}
```

char charAt(int index)

- Retorna o caracter da posição index.

```
public static void main(String[] args) {  
    String palavra1 = "Exemplo";  
  
    System.out.println(palavra1.charAt(0));  
    System.out.println(palavra1.charAt(2));  
    System.out.println(palavra1.charAt(5));  
}
```

Saída:

E
e
l

int length()

- Retorna a quantidade de caracteres na string.

```
public static void main(String[] args) {  
    String palavra1 = "Exemplo";  
  
    System.out.println(palavra1.length());  
}
```

Saída:

7

Exercício 1

- Faça uma função que receba por parâmetro uma palavra e mostre na tela a palavra na vertical.

Exercício 2

- Faça uma função que receba por parâmetro uma palavra e mostre na tela a palavra como mostrado abaixo.

- Palavra = “Exemplo”

- Saída:

E

Ex

Exe

Exem

Exemp

Exempl

Exemplo

Exercício 3

- Faça uma função que receba por parâmetro uma palavra e devolva a quantidade de letras ‘a’ que aparecem na palavra.

String toLowerCase()

Devolve a string com todas as letras minúsculas.

```
public static void main(String[] args) {  
    String palavra1 = "Exemplo";  
  
    System.out.println(palavra1);  
    System.out.println(palavra1.toLowerCase());  
}
```

Saída:

Exemplo
exemplo

String toUpperCase()

Devolve a string com todas as letras maiúsculas.

```
public static void main(String[] args) {  
    String palavra1 = "Exemplo";  
  
    System.out.println(palavra1);  
    System.out.println(palavra1.toUpperCase());  
}
```

Saída:

Exemplo

EXEMPLO

String substring(int beginIndex, int endIndex)

Devolve uma substring que começa na posição beginIndex até a posição endIndex-1.

```
public static void main(String[] args) {  
    String palavra1 = "Exemplo";  
    String palavra2;  
  
    palavra2 = palavra1.substring(2, 5);  
    System.out.println(palavra1);  
    System.out.println(palavra2);  
}
```

Saída:

Exemplo
emp

int indexOf(int ch)

Devolve o índice da primeira ocorrência do carácter ch ou -1 caso não exista.

```
public static void main(String[] args) {  
    String palavra1 = "Exemplo";  
  
    System.out.println(palavra1.indexOf('e'));  
}
```

Saída:

2

int indexOf(String substring)

Devolve o índice da primeira ocorrência da substring ou -1 caso não exista.

```
public static void main(String[] args) {  
    String palavra1 = "Paralelepipedo";  
  
    System.out.println(palavra1.indexOf('p', 9));  
}
```

Saída:

10

int indexOf(String substring, int fromIndex)

Devolve o índice da primeira ocorrência da substring ou -1 caso não exista a partir da posição fromIndex.

Exercício 4

- Faça uma função que receba por parâmetro uma palavra e uma substring e retorne a palavra sem a substring.
- Exemplo:
 - Palavra = “Paralelepípedo”
 - Substring = “ara”
 - Retorno da função: “Plelepípedo”

int lastIndexOf(char ch)

Devolve o índice da última ocorrência do caracter ch ou -1 caso não exista.

```
public static void main(String[] args) {  
    String palavra1 = "Paralelepipedo";  
  
    System.out.println(palavra1.lastIndexOf('a'))  
};  
}
```

Saída:

3

int lastIndexOf(String substring)

Devolve o índice da última ocorrência da substring ou -1 caso não exista.

```
public static void main(String[] args) {  
    String palavra1 = "Paralelepipedo";  
  
    System.out.println(palavra1.lastIndexOf("le"));  
}
```

Saída:

6

Exercício 5

- Faça uma função que receba por parâmetro uma frase e retorne uma subfrase sem a primeira e a última palavra.

String replace(char old, char new)

Devolve a string com os caracteres iguais a old substituídos pelo caracter new.

```
public static void main(String[] args) {  
    String palavra1 = "Paralelepipedo";  
  
    System.out.println(palavra1.replace('a', 'e'));  
}
```

Saída:

Perelelepipedo

String replace(CharSequence old, CharSequence new)

Devolve a string com a substring old substituída pela substring new.

```
public static void main(String[] args) {  
    String palavra1 = "Paralelepipedo";  
  
    System.out.println(palavra1.replace("pipe", "lele"));  
}
```

Saída:

Paraleleleledo

String concat(String str)

Devolve a string com string str concatenada ao final.

```
public static void main(String[] args) {  
    String p1 = "Hoje";  
    String p2 = "tem";  
    String p3 = "aula.";  
    String p4;  
    p4 = p1.concat(" ").concat(p2).concat(" ").concat(p3);  
    System.out.println(p4);  
}
```

Saída:

Hoje tem aula.

Outras

- **boolean contains(CharSequence s)**
 - Retorna true se a string contém a substring s e false caso contrário
- **boolean equals(Object another)**
 - Retorna true se a string another é igual à string e false caso contrário.
- **boolean isEmpty()**
 - Retorna true se a string está vazia e false caso contrário.